CS 5001 – Applied Social Network Analysis
Spring 2021
HW #4

35 points

Submit to Canvas by 11:59 p.m. on Wednesday, Mar. 10, 2021

What to Do:

For this assignment you are to write a **Python** program that will **analyze the network dynamics of the Game of Thrones characters.**

Posted on Canvas is a file[1] (**GameOfThrones.txt**). Each row in the dataset contains the names of two characters, the number of scenes that they appeared together in (hence, the "strength" of their relationship), and the season of the series in which they appeared. You may need to look up the documentation for the Networkx function **read_edgelist** to figure out how to read the data from the file and make an undirected graph, where the edges have attributes for the "strength" (i.e., "weight") and "season." The graph contains 127 nodes and 549 edges; the average degree of a node is 8.65.

You are to do all of the following for the graph:

(1) Display the graph with the nodes labelled (i.e., so that we can see the characters' names); you don't need to display the edges labels.

(2) Output the number of **maximal cliques**, the size of the largest maximal clique, and the number of maximal cliques that are of that largest size.

(3) Output the number of nodes in the **main core** and the k value that gives the main core (i.e., what is the value of k such that every node in the main core has at least k

[1] This file was created on a Windows machine. If you're using it on a Unix/Linux machine, you may want to run *dos2unix* on it before using it to remove any weird characters like '\r'

neighbors?). Also display a subgraph containing the main core (both the nodes and the edges; the nodes should be labelled, but the edges don't need to be).

(4) Output the number of nodes in the **main crust**.

(5) Output the number of nodes in the **k-corona**, where k is the value that gave you the main core. Also display a subgraph containing this k-corona (both the nodes and the edges; the nodes should be labelled, but the edges don't need to be).

(6) Output the number of nodes in the **main shell**. Also display a subgraph containing the main shell (both the nodes and the edges; the nodes should be labelled, but the edges don't need to be).

(7) Display a graph where the nodes in the **main core are red** and the nodes in the **main crust are blue**; do not include the node labels (i.e., the characters' names) or edge labels in this display as they may be difficult to read.

(8) Using the **Louvain method**, output the number of communities, the size (i.e., number of nodes) of the largest community, the size of the smallest community, and the modularity of this partitioning.

(9) Display the graph using the **Louvain partitioning**, with the **nodes in different colors** according to which partition they are in. You don't have to include the names of the nodes or edge labels if they are difficult to read.

(10) Using the **Girvan-Newman method**, output the number of communities, the size (i.e., number of nodes) of the largest community, the size of the smallest community, and the modularity of this partitioning.

(11) Display the graph using the **Girvan-Newman partitioning**, with the **nodes in different colors** according to which partition they are in. You don't have to include the names of the nodes or edge labels if they are difficult to read.

(12) Provide written answers to each of the following questions:

(a) Are the **largest maximal cliques disjoint**? If not, what can you say about actors that are in multiple largest maximal cliques? Do <u>**NOT**</u> just say something generic like "they're well connected"; tell us something <u>**RELEVANT**</u> to the fact that this dataset represents "Game of Thrones" actors and the scenes in which they appeared with each other!

(b) <u>Write code</u> to determine if the members of each of the **largest maximal cliques are together in the communities** found by Louvain partitioning. Show your results and then briefly explain what you think is the significance of this.

(c) What is the relationship between the nodes that are in the **main core** and the nodes that are in the **largest maximal cliques**? Are they the same? Different? What does this tell us?

Note that we're not doing anything with the "strength" or "season" information from this dataset. We could have, like only considering nodes that met a certain threshold of strength, but we're not. Sometimes you get more data in your dataset than you actually end up using.

Extra Credit!

You can earn **extra credit** on this assignment by doing the following, **all in Neo4j**:

- Create a **graph database** by importing the data from the given csv file. Provide a screenshot of at least part of the graph showing the APPEARED_WITH relations.
- Find the number of the **maximal cliques** in the graph.
- Find the **main core** in the graph. Your code has to figure out what the main core is; you can't hard-code a k value.
- Find the **main crust** in the graph.
- Find the **k-corona**, where k is the value that gives you the main core.
- Find the **main shell**.

Note: There are not predefined functions to compute the above things. However, some useful documentation on Neo4j's Graph Data Science Library can be found at https://neo4j.com/docs/graph-data-science/current/introduction/

What to Turn In:

Here's what you need to submit via Canvas (all as a **single** pdf file):

(1) A listing of **your source code**.

(2) A screen shot showing the **output** for all of the things you were asked to do, **in exactly the same order** you were asked to do them!

Grading:

Here's how many points each task is worth:

Task	Points Possible
Read in data, build graph, display it	2
Output # maximal cliques, largest size, # of that size	3
Output # nodes in main core and k, display main core	4
Output # nodes in main crust	1
Output # nodes in k corona, display k-corona	1
Output # nodes in main shell, display main shell	1
Display graph with colored core and crust nodes	2
Output # communities by Louvain	1
Output min and max community size by Louvain	2
Output modularity by Louvain	1
Display graph with colored Louvain communities	2
Output # communities by Girvan-Newman	2
Output min and max community size by Girvan-Newman	2
Output modularity by Girvan-Newman	1

Display graph with colored Girvan-Newman communities	2
Largest maximal cliques analysis	2
Largest maximal cliques, Louvain community analysis	3
Main core vs. largest maximal cliques	3
Total	**35**

Extra Credit Task	**Points Possible**
Build graph database in Neo4j from the datafile	1
Find # maximal cliques using Neo4j	4
Find main core using Neo4j	4
Find main crust using Neo4j	2
Find k-corona for main core using Neo4j	1
Find main shell using Neo4j	3
Total	**15**