

## Problem Set 0

Due 11:59pm PDT October 6, 2016

No late days are allowed for this problem set. This problem set should be completed individually.

### General Instructions

This exercise is easy and will get you started on tools for network analysis. If you find these questions difficult, then you might not be ready for this course.

**Submission instructions:** You should submit your answers via GradeScope and your code via the SNAP submission site. Register for GradeScope at <http://gradescope.com> using your Stanford e-mail (if not SCPD) and include your student ID number with sign-up. Use the entry code **MJ2ZDM** to sign up for CS224W.

*Submitting answers:* Prepare answers to your homework in a single PDF file and submit it via GradeScope. Make sure that the answer to each sub-question is on a *separate, single page*. The number of the question should be at the top of each page. Fill out the information sheet and sign it in order to acknowledge the Honor Code (if typing the homework, you may type your name instead of signing). This should be the last page of your submission. Failure to fill out the information sheet will result in a reduction of 2 points from your homework score.

Please use the solution template file to prepare your submission.

*Submitting code:* Upload your code at <http://snap.stanford.edu/submit>. Put all the code for a single question into a single file and upload it.

### Questions

The purpose of these exercises is to get you started with network analysis and the SNAP software. For this homework, you need to install and try out the SNAP network analysis tools. We strongly encourage you to use Snap.py for Python (available from <http://snap.stanford.edu/snappy/>). However, you can also use SNAP for C++ (available from <http://snap.stanford.edu/snap/download.html>)

#### 1 Analyzing the Wikipedia voters network [9 points]

Download the Wikipedia voting network `wiki-Vote.txt.gz`: <http://snap.stanford.edu/data/wiki-Vote.html>.

Using one of the network analysis tools above, load the Wikipedia voting network. Note that Wikipedia is a directed network. Formally, we consider the Wikipedia network as a directed graph  $G = (V, E)$ , with node set  $V$  and edge set  $E \subset V \times V$  where (edges are ordered pairs of nodes). An edge  $(a, b) \in E$  means that user  $a$  voted on user  $b$ .

To make our questions clearer, we will use the following small graph as a running example:  $G_{\text{small}} = (V_{\text{small}}, E_{\text{small}})$ , where  $V_{\text{small}} = \{1, 2, 3\}$  and  $E_{\text{small}} = \{(1, 2), (2, 1), (1, 3), (1, 1)\}$ .

Compute and print out the following statistics for the `wiki-Vote` network:

1. The number of nodes in the network. ( $G_{\text{small}}$  has 3 nodes.)
2. The number of nodes with a self-edge (self-loop), i.e., the number of nodes  $a \in V$  where  $(a, a) \in E$ . ( $G_{\text{small}}$  has 1 self-edge.)
3. The number of directed edges in the network, i.e., the number of ordered pairs  $(a, b) \in E$  for which  $a \neq b$ . ( $G_{\text{small}}$  has 3 directed edges.)
4. The number of undirected edges in the network, i.e., the number of unique unordered pairs  $(a, b)$ ,  $a \neq b$ , for which  $(a, b) \in E$  or  $(b, a) \in E$  (or both). If both  $(a, b)$  and  $(b, a)$  are edges, this counts a single undirected edge. ( $G_{\text{small}}$  has 2 undirected edges.)
5. The number of reciprocated edges in the network, i.e., the number of unique unordered pairs of nodes  $(a, b)$ ,  $a \neq b$ , for which  $(a, b) \in E$  and  $(b, a) \in E$ . ( $G_{\text{small}}$  has 1 reciprocated edge.)
6. The number of nodes of zero out-degree. ( $G_{\text{small}}$  has 1 node with zero out-degree.)
7. The number of nodes of zero in-degree. ( $G_{\text{small}}$  has 0 nodes with zero in-degree.)
8. The number of nodes with more than 10 outgoing edges (out-degree  $> 10$ ).
9. The number of nodes with fewer than 10 incoming edges (in-degree  $< 10$ ).

Each sub-question is worth 1 point.

## 2 Finding Experts on the Java Programming Language on StackOverflow [4 points]

Download the StackOverflow network `stackoverflow-Java.txt.gz`: <http://snap.stanford.edu/class/cs224w-data/hw0/stackoverflow-Java.txt.gz>. An edge  $(a, b)$  in the network means that person  $a$  endorsed an answer from person  $b$  on a Java-related question.

Using one of the network analysis tools above, load the StackOverflow network. Note that StackOverflow is a directed network.

Compute and print out the following statistics for the `stackoverflow-Java` network:

1. The number of weakly connected components in the network. This value can be calculated in Snap.py via function `GetWccs`.
2. The number of edges and the number of nodes in the largest weakly connected component. The largest weakly connected component is calculated in Snap.py with function `GetMxWcc`.
3. The top 3 most central nodes in the network by PageRank scores. PageRank scores are calculated in Snap.py with function `GetPageRank`.
4. The top 3 hubs and top 3 authorities in the network by HITS scores. HITS scores are calculated in Snap.py with function `GetHits`.

Each sub-question is worth 1 point.

You can find more details about this exercise on the Snap.py tutorial page: <http://snap.stanford.edu/proj/snap-icwsm/>. As an extra exercise, extend the tutorial to find experts in other programming languages or topics.