Assignment : Graph algorithms and Mining

Instructions :

- You must describe your approach in sufficient detail. This would be helpful for partial credit in case you are not able to complete your implementation in time.
- You may use software libraries that implement the required functions. There are many software libraries for graphs, a few examples are SNAP (a C++ library `http://snap.stanford.edu/` written in C++, python interface also provided), igraph (`https://igraph.org/`, available with R, python and C), NetworkX and Neo4j.
- While you may discuss with others, the submission must be your own work.
- There may be no single right answer. Hence, do not just report the final answers, conduct some analysis of the results and clearly explain any insights obtained from these analyses.

1 Problem description

Use the LastFM-Asia graph dataset `http://snap.stanford.edu/data/feather-lastfm-social.html` for this assignment.

Various models have been proposed for the spread of information in a network (graph). Such models have been used in various applications ranging from modeling spread of information in a social network to viruses in a computer network to spread of infectious diseases in a population. A relatively, simple model for information diffusion in a network is the **Linear Threshold model** which is described as below.

- Each node v has a threshold θ_v chosen uniformly at random in the range 0-1], this represents the weighted fraction of v's neighbors that must become active in order for v to become active. Select the thresholds initially and keep these fixed throughout the rest of the simulation.
- For each node v set the weights of edges connecting each neighbor to $b_{u,v} = \frac{1}{n_v}$ where n_v = Number of neighbors of v. Node v becomes active if $\sum_{w \in N(v)} b_{w,v} \geq \theta_v$, where $N(v)$ is the set of neighbors of v i.e. node

v becomes active if the sum of weights of active neighbors exceeds the activation threshold θ_v

- Given an initial set of thresholds (randomly chosen) and active nodes A_0 (with all other nodes inactive), the diffusion process **unfolds deterministically in discrete steps** as follows :

- In step t, all nodes that were active in step $t-1$ remain active, and we activate any node v for which the total weight of its **active neighbors** is at least θ_v
- A node that becomes active once remains active throughout. In the context of modeling spread of diseases this is analogous to the SI (Susceptible Infected) model i.e. a person infected remains infected, at least during the time scale covered by the simulation In contrast in the Susceptible Infected Recovery i.e. SIR model the person has a chance of recovery.

Selection of initial nodes
A crucial aspect in the simulation is the choice of initial nodes. For an application such as targeted marketing run on a social network graph, the aim is to identify a small set of "influential" nodes to be targeted such that the information spreads to as many other nodes as quickly as possible. On the other hand, in applications such as epidemiological modeling of disease spread, the motivation is to identify "influential nodes" to target for interventional treatment such as vaccinations.

2 Experiment description

- Select $n = 10$ initial nodes to target initially. This is the initial set of active nodes A_0. For instance if the nodes selected are 1,6,7,10,21,22,32,34,40 and 80, then $A_0 = \{1, 6, 7, 10, 21, 22, 32, 34, 40, 80\}$. Choose $n = 10$ nodes according to the following **two strategies** and run your experimental simulation.
  - Choose the top n highest degree nodes as the initial set A_0 (degree centrality)
  - Select the n top most influential nodes chosen according to another node centrality measure, possible choices include betweenness centrality, closeness centrality etc. Many of these are implemented in libraries that support graph based analysis.
- Starting with the initial set of starting nodes A_0, run the simulation according to the linear threshold model described above, until no more nodes can be activated. This needs to be done for the two sets of initial nodes chosen according to the above two strategies. For each simulation, plot a graph showing the number of active nodes with respect to time step t. Note that the simulation is run until no more nodes can be activated.