AI607 HW4 Report

Fanchen BU [20194185]

1.1, degree/random, q = 1.0, n = N * 0.1%/0.25%/0.5%/1%/2.5%/5%, N = 75879

n/N	0.1%	0.25%	0.5%	1%	2.5%	5%
degree	9230.9 ± 524.13	12639.6 ± 624.55	15709.3 ± 468.78	20029.1 ± 361.21	25160.9 ± 333.99	29601.8 ± 134.11
random #0	282.3 ± 257.99	760.3 ± 214.01	1815.5 ± 618.65	3418.2 ± 867.92	7720.0 ± 814.72	12304.8 ± 813.43
random #1	254.8 ± 128.05	1033.6 ± 483.45	1599.9 ± 649.78	2774.6 ± 696.26	7362.4 ± 599.34	11693.8 ± 1093.38
random #2	344.3 ± 181.15	852.1 ± 531.08	1524.7 ± 668.95	3430.5 ± 762.65	7586.4 ± 936.55	12459.9 ± 928.63
random #3	326.6 ±149.8	1910.9 ± 433.59	1574.4 ± 632.45	3461.0 ± 336.72	6233.9 ± 1065.54	13070.3 ± 1093.0
random #4	386.3 ± 286.78	1167.7 ± 682.14	2451.9 ± 644.48	3274.4 ± 637.85	7768.2 ± 796.43	12665.3 ± 660.59

1.2, degree/random, q = 0.2/0.4/0.6/0.8/1.0, n = N * 1%, N = 75879

q	0.2	0.4	0.6	0.8	1.0
degree	3065.9 ± 50.33	5940.5 ± 82.61	9605.3 ± 114.47	14196.0 ± 197.4	19750.9 ± 308.89
random #0	855.8 ± 15.37	1013.7 ± 34.97	1255.9 ± 93.58	1690.0 ± 147.43	2973.8 ± 864.17
random #1	934.8 ± 17.5	1217.1 ± 69.63	1663.7 ± 188.53	2238.9 ± 356.82	3706.8 ± 614.48
random #2	874.2 ± 11.65	1030.0 ± 39.01	1352.0 ± 148.65	1789.6 ± 146.17	2882.9 ± 545.0
random #3	858.8 ± 11.03	1011.9 ± 48.83	1307.5 ± 114.59	2016.8 ± 566.94	2940.7 ± 428.23
random #4	895.7 ± 17.2	1151.3 ± 89.82	1431.3 ± 67.24	2061.2 ± 175.58	3880.7 ± 707.18

* Here, data in the same row use the same set of initial active nodes

2.1, custom, q = 1.0, n = N * 1%, N = 75879

I use a quite simple policy. For each node v, let D(v) be its in-degree and N(v) be the set of its out-neighbors.

I only consider nodes that have out neighbors, for each of them, say node k, the score is

$$\sum_{u \in N(k)} 1/D(u),$$

which is just the expectation of the number of nodes (k itself excluded, q=1.0) after 1 timestep, given k is the only active node at the beginning. Then I pick n nodes with highest scores.

The result is 24314.3 ± 114.02, which is better than the ‘degree’ policy with the same parameters.