**DM Project 3: Link Analysis**

1. **Environment:** anaconda spyder 3.2.8, python 3.6, cpu: i7-4770
2. **Implementation Detail**

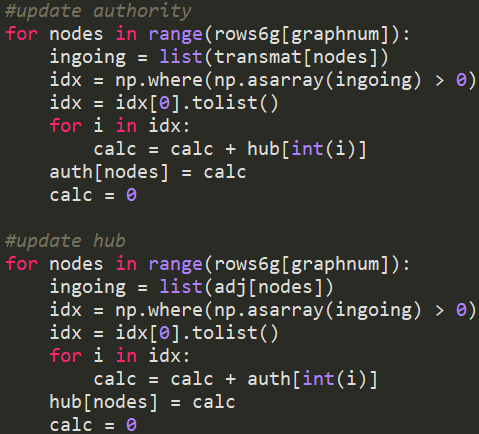
Hits and PageRank are in hw3\_HITS.py, simrank is in hw3\_simrank.py

hw3\_HITS.py: output csv file as result (nodes, auth, hub, pr)

hw3\_simrank.py: output csv file (point 1, point 2, similarity)

**Hits**

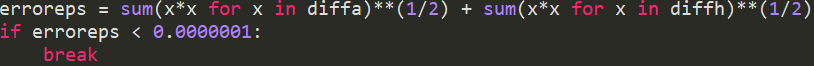
Update authorities first, then hubs.



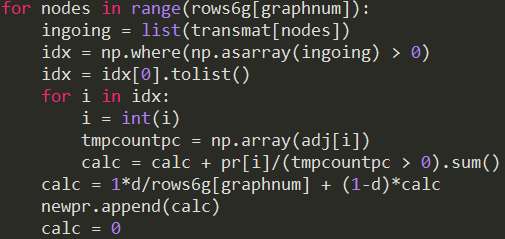
Normalization (here I use one-norm distance, you can change to two-norm by modifying the comment)



Stop condition: epsilon = 10e-7, add two-norm distance of difference vectors, authority and hub.



**PageRank** (one output score for each node, calculate by the formula, d=0.15, stop condition and normalization similar as above)



**SimRank**

Do the following recursion:

S(a,b) stands for the similarity between point a and point b

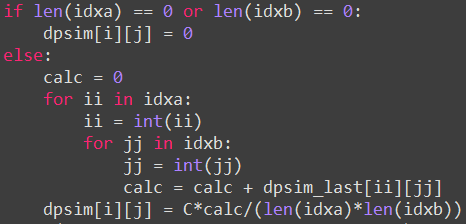
S0: if a =b, S(a,b)=1, else S(a,b)=0

Sk: following by the formula and the result of Sk-1

If one of a or b’s parent set is empty, returns 0.

C is decay factor, set to 0.8 in this case.

Stop condition similar as above.

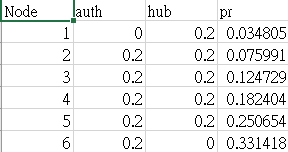


1. **Result Analysis and Discussion**

You can change input graph by modifying “graphnum” parameter

For project1 dataset, change the “project1” and “bidir” parameters, and set “graphnum” to 7.

**Graph 1:**

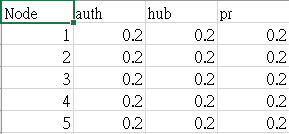


Since there is no link goes into node 1, the authority is 0, same situation as the hub of node 6.

Notice the PageRank is highest in node 6 since every node can get to node 6 after some iterations, however no one can be arrived to node 1, thus the PageRank is relatively low.

SimRank is zero for every pair of points, which is reasonable because no nodes sharing the same parent.

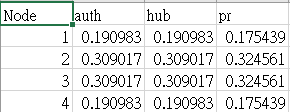
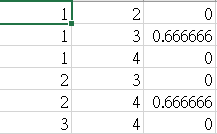
**Graph 2:**



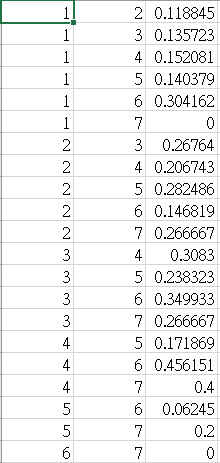
Graph 2 is a cycle, thus each node has the same probability to get to another node, leads to same authorities and hubs.

The SimRank is zero for all pairs in this case too.

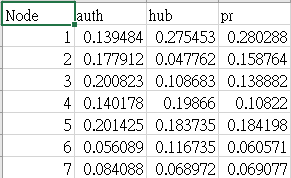
**Graph 3:**



Node 1 and node 4 are the ends of the graph, different from node 2 and node 3. These two nodes only has one ingoing edge and one outgoing edge, while the others has two for both edges. We can observe that the former case has less authority, hub and PageRank values, which is highly correlated with authority.

SimRank is not zero in pair (1,3) and (2,4) since the pair shared the same parent, 2 and 3.

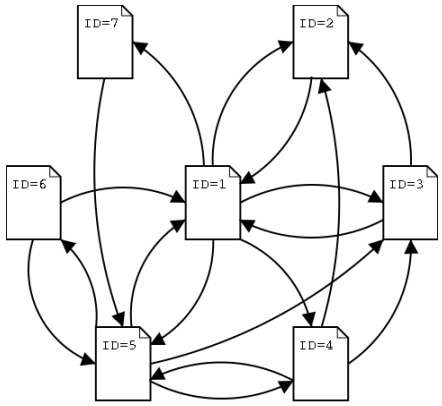
**Graph 4:**



Graph 4 is a more general case of

a small scale graph.

(7 nodes 18 edges)

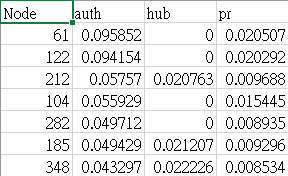
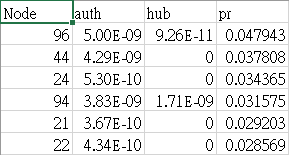


Consider node3 and node5, both have similar authority values, but their amount of ingoing links is different, one is four links and the other is three. This shows a pattern that nodes connected with high hubs will get higher authority value, even though the links weren`t enough.

The SimRank is highest in pair (4,6). I assumed that the reason causing this situation can be described by the path selected by red boxes. They shared the parent node 5, which has cycle connected between node 1, 2, and 3. This may lead to high similarity between the children, node 4 and 6.

**Graph 5 (sorted and picked the highest values)**

Sorted by auth sorted by PageRank



Notice that the node 96 with the highest PageRank doesn`t intuitively has the highest authority and hub. This special case is result that node 96 is a sink, where no outgoing links to other nodes, thus the “rank sink problem” occur and cause unusual PageRank.

Ex: Rank sink Problem

Iter1 1 1 1

Iter2 0.3 2.3 0.3

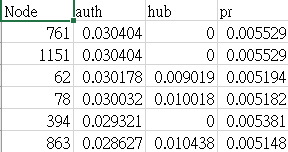
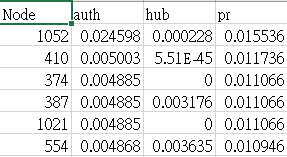
Iter3 0.1 1.37 0.1

Node 3 is like stealing every PageRank score from node 1 and node 2, and also the total PageRank decays fast.

SimRank each pair is written in simrank.csv, not showing result here.

**Graph 6** **(sorted and picked the highest values)**

Sorted by auth sorted by PageRank



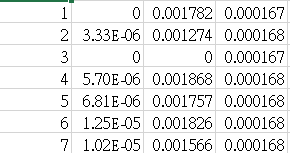
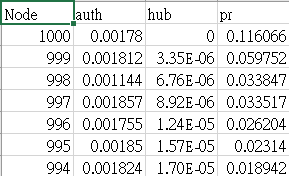
**Graph from project 1 (directed)**

1000 transactions with list length 1000, average item amount in each list is 100.

Full connect each node, from the smaller number to bigger one, eg. 0->10, 30->100, in other words no ingoing link to node 0.

Set graphnum to 7 and project1 = True, bidir to 0 if just one-way.

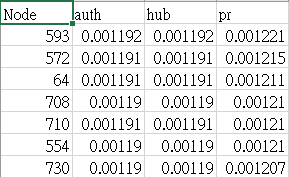
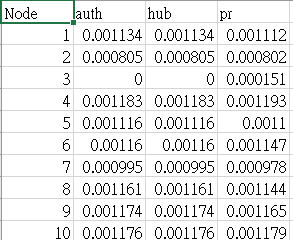
Sorted by node num sorted by PageRank



High Authority and lower hub value may get higher PageRank in this case, notice that the sink has unusual high value of PageRank.

**Graph from project 1 (bidirected)**

Sorted by authority sorted by PageRank

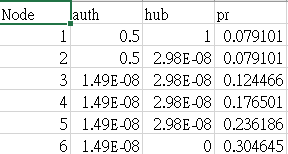
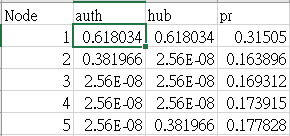
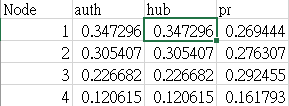


Bidirected graph has more similar values for each node since they can more easily reached to each other, except nodes that don`t connect with any of the others.

**Q: How to improve node 1’s authority, hub, and PageRank at the same time for graph 1~3?**

The answer is quite simple and cheated, that is, add self link to node.

Graph 1 Graph 2 Graph 3



Notice that in graph 1 and graph 2, we observed huge effect on node 1 and its closest neighbor node 2. However in graph 3, since the link is bi-directed, apply a self link on node 1 may not cast a great effect since it originally can get to itself after few iterations.

1. **Computation Performance analysis**

Time complexity for both HITS and PageRank is O((node+link)\*iteration)

The difference is that HITS is faster than running PageRank. The reason is that HITS only consider the node we observed and the link of the node, however PageRank consider the node and its parent nodes’ links, which takes more time for computation and analysis.

Furthermore, HITS is a query-based algorithm (but in the homework implementation I take the whole page set to a single query), which takes even less time than PageRank since PageRank is an algorithm going through the whole graph.

SimRank is O(node^3\*iteration), since it has a huge cost calculating every pair similarity in each iteration

1. **Discussion**

* **Limitations of link analysis algorithm**

Self link, sink are basically not allowed.

* **Can find important pages?**

Mostly can’t. First some people create web pages citing lots of other important pages like Wikipedia and force their web page to have large authority value, or even use lots of pages only linked to its page like advertisements. Second a page without outgoing links (sink) might affect the score in PageRank case. Third, new pages always have less score since the link and cites aren’t enough as old pages for ten or twenty years, causing a situation that the page is important and highly relevant to our query, however, it’s not old enough to receive high importance score.

* **Practical issues?**

PageRank I implemented on graph 6, with only about 1000 nodes and 5000 edges, cost me more than ten minutes to finish calculation. When users surf the net, it means that they have to wait this long time to get query result, which is not very practical.

Practical way is that first we have to define a small closed subset, similar to the query title, then compute the subset with a more refined algorithm based on these original ones.

* **C in simrank**

If you set less C value, you will get less similarity score for each pair of nodes. I guess the effect is that if you set less C, the similarity between two pairs will have less difference (eg. For graph 4, S(1,2)=0.118 and S(1,3)=0.135 with C=0.8, however S(1,2)=0.059 and S(1,3)=0.064 with C=0.5, the difference has diminished), hard for us to observe the relative similarity.

* **New idea?**

We may not just want to analyze the ingoing links and outgoing links only, instead we can take more concern into time factor or content related analysis. This may increase the precision of getting the webpage we want.