COMP 4321 Search Engine for Web and Enterprise Data **Score:**

1)	**4)**
2)	**5)**
3)	**6)**

Mid-Term Examination, Fall 2012
October 30, 2012 Time Allowed: 1 hour

Name: ______________________________ Student ID: ________________________
Note: Answer all questions in the space provided. Answers must be precise and to the point.

1. **[15]** Circle True or False in the following questions:

T [✓ F] When you choose a search engine, you should always choose the one with highest average precision.

[✓ T] F When stemming has been applied to document terms, stemming must be applied to the query terms.

[✓ T] F A large damping factor d in the PageRank formula will result in a larger number of iterations before convergence is reached

[✓ T] F In the vector space model, terms are assumed to be independent in the document collection.

[✓ T] F Similarity between two queries can be defined in the same way as the similarity between a query and a document

T [✓ F] Cosine similarity measures the cosine of the angle between the document vector and the origin of the vector space

[✓ T] F Search Engine Optimization (SEO) is to optimize the ranking of a site in search engines.

T [✓ F] A high Page Rank means a page is more relevant to the query

T [✓ F] A phrase must be broken down into individual words and represented as individual words in the document vector

T [✓ F] Precision and recall must add up to 100%

2. **[5]** Briefly explain why search engine (e.g. Google, Bing) can response (return the relevant results) so fast for a query. (List 3 reasons)

Ans: (1) Crawler will crawl the web pages from time to time and do comprehensive indexing in advance.

(2) Some smart pattern matching algorithm.
(3) Web pages are stored and algorithms are run in distributed system.
(4) the search engine may have cached the results of the queries.
(5) PageRank values of the pages can be pre-computed, etc.

Note: The first is essential. Other coherent answers will also be accepted. Students who can answer not less than two points can get the full mark.

3. **(a) [15]** The table below shows the ***term frequencies*** of the terms, T1, T2, T3 and T4, in three documents, D1, D2 and D3.

	T1	**T2**	**T3**	**T4**	**tf$_{max}$**
D1	2	1	1	0	2
D2	1	2	0	0	2
D3	0	2	0	4	4

Furthermore, there are a total of 1000 documents in the collection, and the document frequencies for T1 to T4 are:
df$_{T1}$ = 20, df$_{T2}$ = 30, df$_{T3}$ = 10, df$_{T4}$ = 20.

Using the **tf/tf$_{max}$** × **idf** weighting strategy, obtain the term weights of each term in each document.

D1:
W(T1) = 2/2 * log $_2$(1000/20) = 5.64;
W(T2) = 1/2 * log $_2$(1000/30) = 2.53;
W(T3) = 1/2 * log $_2$(1000/10) = 3.32;
W(T4) = 0/2 * log $_2$(1000/20) = 0;
D1 = <5.64, 2.53, 3.32, 0>.

D2:
W(T1) = 1/2 * log $_2$(1000/20) = 2.82;
W(T2) = 2/2 * log $_2$(1000/30) = 5.06;
W(T3) = 0/2 * log $_2$(1000/10) = 0;
W(T4) = 0/2 * log $_2$(1000/20) = 0;
D2 = <2.82, 5.06, 0, 0>;

D3:
W(T1) = 0/4 * log $_2$(1000/20) = 0;
W(T2) = 2/4 * log $_2$(1000/30) = 2.53;
W(T3) = 0/4 * log $_2$(1000/10) = 0;
W(T1) = 4/4 * log $_2$(1000/20) = 5.64;
D3 = <0, 2.53, 0, 5.64>

	Wt$_{T1}$	**Wt$_{T2}$**	**Wt$_{T3}$**	**Wt$_{T4}$**
D1	5.64	2.53	3.32	0
D2	2.82	5.06	0	0
D3	0	2.53	0	5.64

(b) [10] Compute the cosine similarity between Q = < 0, 1, 0, 1 > and each of the three documents.

Sim(D1, Q) = $\frac{2.53}{\sqrt{5.64^2+2.53^2+3.32^2}*\sqrt{1^2+1^2}}$ = 0.25;

Sim(D2, Q) = $\frac{5.06}{\sqrt{2.82^2+5.06^2}*\sqrt{1^2+1^2}}$ = 0.62;

Sim(D3, Q) = $\frac{2.53+5.64}{\sqrt{2.53^2+5.64^2}*\sqrt{1^2+1^2}}$ = 0.93;

4. **(a) [5]** State ONE main difference between Google's link-based ranking method and the link-based ranking methods employed in HyPursuit and WISE.

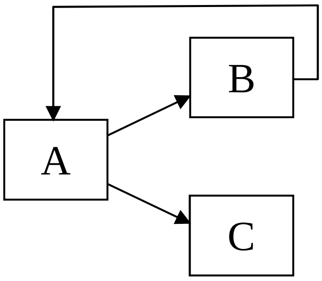
HyPursuit and WISE use links to infer the content similarity between pages that are linked together. Google uses links to infer the authority or quality of a page that are pointed at by the links.

(b) [5] When we say PageRank is "query independent", what does it mean? State one advantage and one disadvantage of the query independence of PageRank.

PageRank is computed purely based on the link structure of the pages. That is, the PageRank of a page is the same no matter what query is submitted.

Pros: efficient; PageRank does not have to be computed for each query. Compute offline.
Cons: a page which is authoritative in one topic may not be authoritative in another topic.

5. **[20]** Given the web graph on the right, (i) would the PageRank values converge? If they do, what values do they converge to? (ii) Does the convergence depend on the damping factor, d? (iii) Which part of the web graph lead to the convergence behavior you observed in (i) and (ii)?



(i) Yes, they will converge.
From the iterations in the table below, the converged values are roughly $1-0.5d^2$ or $1-0.5d$ (after dropping higher order terms). However, if you answer something like $1-d$ or $1-d^2$ (forget about the 0.5 coefficient), that is fine.

(ii) Either Yes or No, it depends on students' explanation. If Yes, the reason is it will affect the values that the PageRank values converge to. If "No", the reason is, the iteration always converges which is independent on the value of d, as long as the graph topology is not dangling.

(iii) This is because the PR of A will be divided into two halves, and B gets only one half of A's PR, and in the next iteration, A's PR is half of its value in the previous iteration. The division leads to the convergent behavior.

The page rank is actually computed by solving the functions:

PR(A)=1-d + d*PR(B)*
PR(B)=1-d + 0.5d*PR(A)
PR(C)= 1-d + 0.5d*PR(A)

There are two methods (basically, they are the same):

1. Iteratively computing the values until they converge

2. Solve the functions directly

Using method 1, we have

rank(A)= $1 - 0.5d^2 - 0.25d^4 - 0.125d^6 - \ldots = (1-d^2)*2/(2-d^2)$,

rank(B)= rank(C) = $1 - 0.5d - 0.25d^3 - 0.125d^5 - \ldots = (2-d-d^2)/(2-d^2)$

Using method 2, solving the functions directly, we can still get

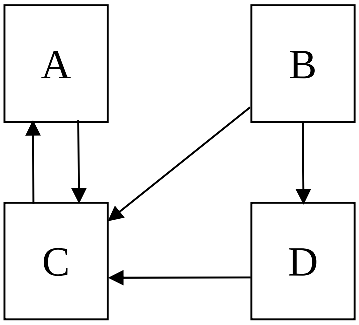
rank(A)= $(1-d^2)*2/(2-d^2)$,

rank(B)= rank(C) = $(2-d-d^2)/(2-d^2)$

[Note: I would not expect students to give my exact answer. Something close to it could be considered correct. The following actual PR calculation may help to understand the behavior, but it is not required in the answer.]

	0	1	2	3	4
PR(A)=1-d + d(PR(B))	1	1-d + d(1)=1	1-d + d(1-0.5d)=1-0.5d^2	1-d + d(1-0.5d)=1-0.5d^2	1-d + d(1-0.5d-0.25d^3)=1-0.5d^2 - 0.25d^4
PR(B)=1-d + 0.5d*PR(A)	1	1-d + 0.5d =1-0.5d	1-d + 0.5d=1-0.5d	1-d + 0.5d(1-0.5d^2)=1-0.5d-0.25d^3	1-d + 0.5d(1-0.5d^2)=1-0.5d-0.25d^3
PR(C)= 1-d + 0.5d*PR(A)	1	1-d + 0.5d =1-0.5d	1-d + 0.5d=1-0.5d	1-d + 0.5d(1-0.5d^2)=1-0.5d-0.25d^3	1-d + 0.5d(1-0.5d^2)=1-0.5d-0.25d^3

6. **[25]** Given the web graph below, compute the PageRank values, Hub and Authority weights for iteration 1 to 3. Assuming that the damping factor in PageRank is: d=0.15. For Hub and Authority weights, there is no need to normalize the weights in each iteration by the vector length.



Page Rank:

Iteration	**0**	**1**	**2**	**3**
PageRank(A)	1	0.85 + 0.15 (1) = 1	0.85 + 0.15 (1.225) = 1.03	0.85 + 0.15 (1.2) = 1.03
PageRank(B)	1	0.85	0.85	0.85
PageRank(C)	1	0.85 + 0.15 (1/1 + 1/2 + 1/1) = 1.225	0.85 + 0.15 (1/1 + 0.85/2 + 0.925) = 1.20	0.85 + 0.15 (1.03 + 0.85/2 + 0.91) = 1.21
PageRank(D)	1	0.85 + 0.15 *0.5 = 0.925	0.85 + 0.15 * 0.85/2 = 0.91	0.85 + 0.15*0.85/2 = 0.91

Authority Weights: [summation of hub weights of parents]

	0		1	2	3
A	1	=Hub(C)	1	1	1
B	1	0	0	0	0
C	1	=Hub(A)+Hub(B)+Hub(D)	3	~~4~~	10
D	1	=Hub(B)	1	2	4

Hub Weights: [summation of authority weights of children]

	0		1	2	3
A	1	=Aut(C)	1	3	4
B	1	=Aut(C)+Aut(D)	2	4	6
C	1	=Aut(A)	1	1	1
D	1	=Aut(C)	1	3	4