Lecture 2, Ranking 1

Tuesday, September 12, 2006
1:14 PM

I. How search engines work:
   a. Crawl the web, creating a database
   b. Answer query somehow, e.g. grep. (ex. Funk search)
   c. Revolution came with two ideas for exploiting link structure (beyond indegree)
      i. PageRank of Brin-Page, Google
      ii. Hits, Kleinberg's algorithm, hubs and authorities
      iii. Downside of being at IBM.

II. We will start with PageRank, which Google supposedly uses
   a. Disclaimer: what Google actually does is not public knowledge.
   b. Main idea: give a score to every page on the web
   c. When get a query, use old technology to get a list of pages (e.g. grep), and then display them in the order given by their scores.
   d. PageRank is the algorithm for assigning a score/rank.

I. First approximation of PageRank.
   a. This first approximation is not quite what do, but is a good approximation. Will fill fix some details later in the lecture, once we know enough for it to make sense.
   b. Notation: for a node v, let $d^+(v)$ denote the number of edges leaving v, the out-degree of v, and $d^-(v)$ be the number of edges entering v, the in-degree of v.
   c. We would like PageRank to be a function from vertices to the non-negative reals, r such that

$$r(v) = \sum_{(u,v) \in E} r(u) \frac{1}{d^+(u)}$$

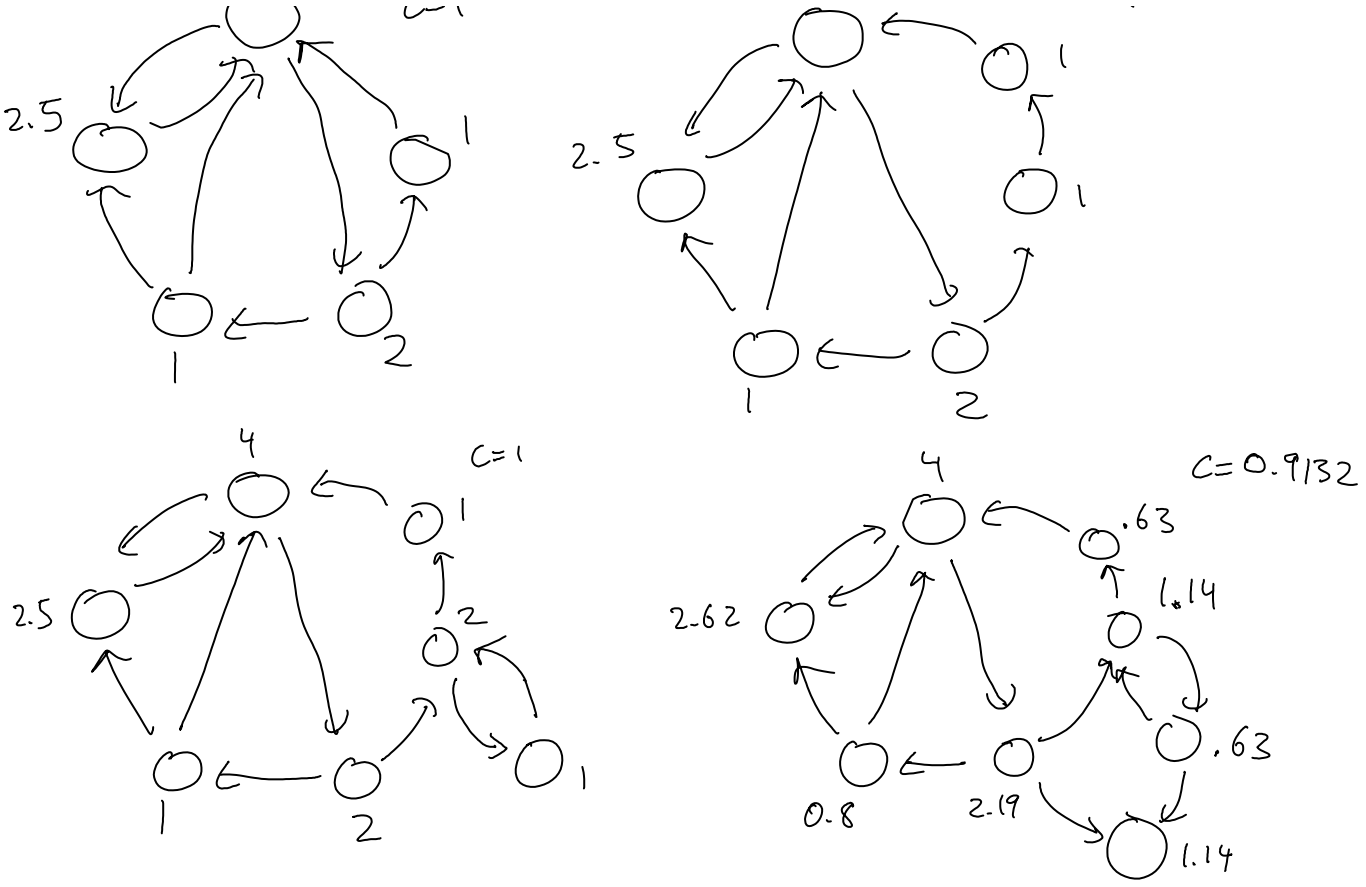
   d. As it is not necessarily possible to satisfy this equation, we just ask for a solution to the more general equation

$$r(v) = c \sum_{(u,v) \in E} r(u) \frac{1}{d^+(u)}$$

      For some $c > 0$.
   e. Finally, r is normalized so that

      $\sum_v r(v) = 1$.

      Here are some example, but without the normalization so the numbers are easier to read.



1. Does a PageRank vector exist? Yes.
   a. Here's one way to compute a PageRank vector. It converges relatively quickly, as can be seen from the example. In fact, the actual vector converges even faster.

   b. Is the PageRank vector unique?
      i. Not necessarily. If the graph has disconnected components, there will be many solutions to the equations. Correct pagerank algorithm will fix this.

   c. If the graph is strongly connected: for every u and v, is a directed path from u to v, then r exists, is unique, and has c = 1. We will now prove this fact. But first, we must introduce some matrix notation for dealing with this.

   d. The Adjacency matrix of a graph is a matrix A such that

$$A_{u,v} = \begin{cases} 1 & \text{if } (u,v) \in E \text{ from } u \text{ to } v \\ 0 & \text{o.w.} \end{cases}$$

   Recall that $A_{u,v}$ is the entry in the uth row and vth column.

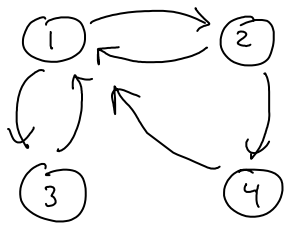
   Now, define the matrix D by

$$D_{u,v} = \begin{cases} d^+(u) & \text{if } u = v \\ 0 & \text{o.w.} \end{cases}$$

   And, the matrix M by

$$M = D^{-1}A$$

Here's an example:



$$D^{-1} = \begin{pmatrix} \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

$$M = D^{-1}A = \begin{pmatrix} 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

Note that the sum of the entries in every row of M is 1. For M to be well-defined, we require that $d^+(v) > 0$ for all v.

Now, the PageRank equations become r = c r M. That is, r is a left-eigenvector of M of eigenvalue 1/c.

Now, let's look carefully at this example, and see why the following vector satisfies the PageRank equations:

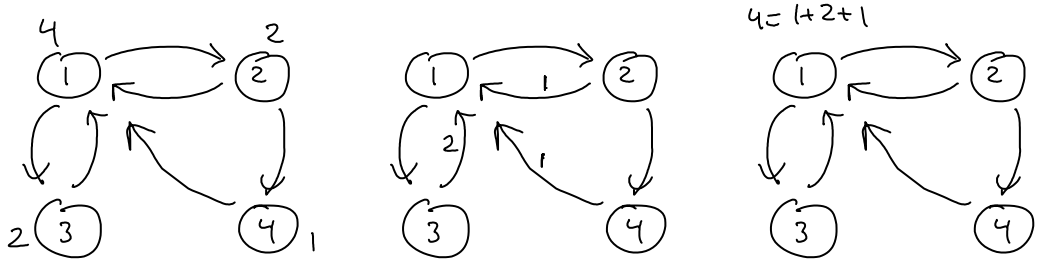
We have $(4\ 2\ 2\ 1)\ M = (4\ 2\ 2\ 1)$

because $(4\ 2\ 2\ 1)\ D^{-1} = (2\ 1\ 2\ 1)$

this is how much authority will be transported from each node, and

$$(2\ 1\ 2\ 1)\begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} = (4\ 2\ 2\ 1)$$

here's a picture for the first vertex:



a. Again, let's consider the case in which the graph is strongly connected, and show that 1 is an eigenvalue. This is necessary to show that c=1 is achievable. To show this, we show that the all-1's vector is a right-eigenvector of eigenvalue 1. To see this, note that A***1** is a column-vector giving the out-degree of every vertex. So, D^{-1} (A***1**) is again the all-1's vector. Now, as the right and left-eigenvalues are the same, we know that M has a left-eigenvector of eigenvalue 1.

b. In a few moments, we will show that the left-eigenvector of eigenvalue 1 is non-negative, and unique. But, for now let's assume that a non-negative vector r exists such that r = r M, and show this implies it is unique.

c. First, we establish that if the graph is strongly connected, r = r M, and r is non-negative and has at least one non-zero entry, then r is strictly positive. Assume $r(u_0) > 0$. Then, for every v, there exists a directed path from u_0 to u_1 to … to u_k to v. From the PageRank equations, we have

$$r(u_1) = \sum_{(v,u_1)\in E} r(v)\frac{1}{d^+(v)} \geq r(u_0)\frac{1}{d^+(u_0)} > 0$$

So, by induction, $r(u_k) > 0$ and $r(v) > 0$.

In particular, $$r(v) \geq r(u_0)\frac{1}{d^+(u_0)d^+(u_1)\cdots d^+(u_k)}$$

d. Now, let's observe that if r and s are two vectors that satisfy r = r M and s = s M, then for every beta, (r + beta s) = (r + beta s) M. This follows simply from linearity: $(r+\beta s)M = rM + \beta sM = r + \beta s$

e. Finally, note that if r is strictly positive, and r and s satisfy r = r M and s = s M, then there is a beta such that the vector (r - beta s) is non-negative, but has at least one zero entry. As we will also have (r - beta s) = (r - beta s) M, this will contradict the point established above. So, any such r must be unique.

f. I was going to show that if the graph is strongly connected and r = r M then r must be non-negative. But, I ran out of time. So, we will move this fact to a problem set.

There are two more things I should say about how PageRank works. The first is how they handle nodes with out-degree zero. Brin and Page say that they ignore these nodes at first, solve the remaining problem, and then put those nodes back in. We could think of many reasonable ways of making this formal. I'm not sure which one they actually do.

To explain the second point, let me first observe that in the algorithm as we specified so far, each node make equal contributions to the ranks of the nodes it points to. It would be easy to imagine that a node could make different

contributions to each of the nodes it points to. In fact, Brin and Page make each node make a small contribution to **every other node.**

Formally, the matrix they consider is given by:

$$M_{u,v} = \begin{cases} \frac{1-\alpha}{d^+(u)} + \frac{\alpha}{n} & \text{if } (u,v) \in E \\ \frac{\alpha}{n} & \text{o.w.} \end{cases}$$

Which can also be written:

$$M = (1-\alpha) D^{-1} A + \frac{\alpha}{n} J,$$

where J is the all-1's matrix

They say that they use $\alpha = 0.15$

Now, let me explain the probabilistic approach to understanding pagerank. They imagine a monkey surfing the web. At each time step, the monkey chooses to follow a random link on a page (with probability 1- alpha), or to go to a completely random place on the web (with probability alpha). Imagine that the monkey surfs for a very long time. Eventually, there is one probability distribution giving the chance that the monkey is at each page. The probability of being at page v becomes r(v), where r is the pagerank vector.

To see this, consider the probability of being at a page v. It is the sum over all pages u that point to v of the probability of being at u, times the probability of jumping from v to u. This is exactly what is captured by the equation

$$\tau(v) = \sum_u \tau(u) M_{u,v}$$

This is why we chose the normalization

$$\sum_v \tau(v) = 1$$

the sum of probabilities should be 1.

This is also why we want $\angle_{V} M_{u,v} = 1$

Finally, I also mentioned Kleinberg's algorithm, called Hits.
Kleinberg had the idea that some nodes are authorities on certain topics, and others are hubs, which list authorities. For example, my web page points to authorities on many topics. So, he assigned each node a hub-weight, h(v), and an authority-weight, a(v). A page that is pointed to by many good hubs becomes an good authority, and a page that points to many good authorities should be a good hub. So, he asked that the vectors a and h satisfy:

$$a(v) = \sum_{(u,v) \in E} h(u)$$

$$h(u) = \sum_{(u,v) \in E} a(v)$$

In matrix form, these give

$$a = hA \quad \text{and}$$

$$h = aA^T,$$

which when combined give

$$a = aA^TA \quad \text{and}$$

$$h = hAA^T.$$