MS&E 235: Lecture 10
Prof. Ashish Goel
May 5, 2009
Notes by Rio Goodman

Lecture 10: PageRank

In the last lecture, we developed an expression for naïve PageRank. We called the set of web pages (or nodes) V, and the set of links (or directed edges) E. If page u has a link to page v, then $(u, v) \in E$. We denote the number of edges *coming out of* a page, v, as d_v, and the PageRank of v as π_v. As we derived last time, we get the following expression for naïve PageRank.

$$\pi_v = \sum_{(w,v)\in E} \frac{\pi_w}{d_w}$$

such that

$$\sum_v \pi_v = 1$$

$$\pi_v \geq 0 \text{ for all } v$$

Consider the following example network, where page A has a link to B, B has links to A and C, and C has a link to B:

$$\begin{tikzcd}
 & B \arrow[dl, bend right] \arrow[dr] & \\
A \arrow[ur] & & C \arrow[ul, bend right]
\end{tikzcd}$$

There is only one link from A, so $d_A = 1$. Similarly, $d_C = 1$. We can plug these values into the PageRank expression above, to obtain an equation for π_B:

$$\pi_B = \pi_A + \pi_C$$

We can find expressions for π_A and π_C in a similar manner:

$$\pi_A = \frac{\pi_B}{2}$$

$$\pi_C = \frac{\pi_C}{2}$$

The constraints from the expression above become:

$$\pi_A + \pi_B + \pi_C = 1$$

$$\pi_i \geq 0$$

Thus, the unique solution to this set of expressions is: $\pi_A = \pi_C = 1/4$, $\pi_B = 1/2$.

Flaws in naïve PageRank

For the previous network, we were able to efficiently obtain a unique solution. Now consider a different example network:

$$\begin{tikzcd}
& B \arrow[dr] & \\
A \arrow[ur] & & C \arrow[ll]
\end{tikzcd}$$

The naïve PageRank will yield the solution: $\pi_A = \pi_B = \pi_C = 1/3$

A web page cannot change the number of links that are coming into it, but it can construct new links that are going out of it. Is there any unilateral change a web page can make to increase its own PageRank? Lets see what happens if C changes its link from A to B.

The network will now look like this:

$$\begin{tikzcd}
& B \arrow[dr, bend right] & \\
A \arrow[ur] & & C \arrow[ul, bend right]
\end{tikzcd}$$

Since A has no links to it, $\pi_A = 0$. Then $\pi_B = \pi_C = 1/2$. So C has increased its PageRank from 1/3 to 1/2. This problem is amplified by a ring network:

$$\begin{tikzcd}
a_1 \arrow[d, bend right] & a_N \arrow[l, bend right] \\
a_2 \arrow[r, bend right] & a_3 \arrow[u, dotted]
\end{tikzcd}$$

Where the PageRank for each page is $\pi_i = 1/N$ for $i = 1, ..., N$. Now suppose a_2 changes its link from a_3 to point to a_1. We will get this network:

$$\begin{tikzcd}
a_1 \arrow[d, bend right] & a_N \arrow[l, bend right] \\
a_2 \arrow[u, bend right] & a_3 \arrow[u, dotted]
\end{tikzcd}$$

Now the PageRank of a_3 will be zero, because there are no links to it. If $\pi_{a_3} = 0$, then $\pi_{a_4} = 0$, and the zero will propagate through the network so that all pages are left with PageRank zero except for a_1 and a_2. Thus $\pi_{a_1} = \pi_{a_2} = 1/2$. So by changing one link, a_2 increased its PageRank from $1/N$ to $1/2$.

This example demonstrates naïve PageRank's vulnerability to collusion, or "link spam." Yet there is a more serious problem with naïve PageRank: the solution is not always well defined. Consider the following network:

$$\begin{tikzcd}
& b \arrow[dr] & & & q \arrow[dr] & \\
a \arrow[ur] & & c \arrow[ll] & p \arrow[ur] & & r \arrow[ll]
\end{tikzcd}$$

Where a, b and c are in a different community than p, q and r, with no links between the two communities. From the structure of this network, it is easy to see that we must satisfy the conditions

$$\pi_a = \pi_b = \pi_c$$

$$\pi_p = \pi_q = \pi_r$$

We also must satisfy the standard conditions

$$\pi_a + \pi_b + \pi_c + \pi_p + \pi_q + \pi_r = 1$$

$$\pi_i \geq 0 \text{ for all } i$$

But there are infinitely many solutions to this system. One possible solution is $\pi_a = \pi_b = \pi_c = 1/3$, $\pi_p = \pi_q = \pi_r = 0$. Another is $\pi_a = \pi_b = \pi_c = 0$, $\pi_p = \pi_q = \pi_r = 1/3$. And any average of these two solutions is also a solution.

So for this example, there are infinitely many solutions. For other networks, there may not be any solution.

PageRank

To solve this problem of ill defined networks, we turn to (not naïve) PageRank. This will be analogous to the internet monkey we discussed last lecture randomly jumping to a new page every so often. Now instead of the monkey following a link on the current page every time period, the monkey either follows a link, or jumps to a new page (with a certain probability).

The modified expression for this new PageRank will be:

$$\pi_v = (1-\epsilon)\left(\sum_{(w,v)\in E} \frac{\pi_w}{d_w}\right) + \frac{\epsilon}{N}$$

Where N is the total number of pages, and ϵ is the "reset probability" (the probability that the monkey randomly jumps to another page). It is widely believed that Google initially used $\epsilon \approx 1/7$ to calculate PageRank. This formulation of PageRank makes sense, in terms of the way people generally surf on the internet. A normal user will click on links most of the time, but sometimes will type in a new url, and randomly jump to a new page.

Now let us re-examine the example that was not well defined under naïve PageRank:

$$\begin{tikzcd}
 & b \arrow[dr] & & & q \arrow[dr] & \\
a \arrow[ur] & & c \arrow[ll] & p \arrow[ur] & & r \arrow[ll]
\end{tikzcd}$$

Under PageRank, we will now be able to compute a unique solution. We can no longer have a solution where $\pi_p = \pi_q = \pi_r = 0$, because of the $\frac{\epsilon}{N}$ term. Under this new PageRank, we have the condition:

$$\pi_i \geq \frac{\epsilon}{N} \text{ for all } i$$

Lets say that $\epsilon = 0.12$, and thus $\frac{\epsilon}{N} = 0.02$. Is $\pi_a = \pi_b = \pi_c = 0.02$, $\pi_p = \pi_q = \pi_r = \frac{0.94}{3}$ a solution? No, because $\pi_a = 0.88\pi_c + 0.02$. If $\pi_c = 0.02$, then π_a must be greater than 0.02.

In fact, there is only one unique solution to this system: $\pi_i = 1/6$ for all i.

In the ring example, a_2 was able to increase its PageRank by changing its link. Now let us revisit this example.

$$\begin{tikzcd}
a_1 \arrow[d, bend right] & a_N \arrow[l, bend right] \\
a_2 \arrow[u, bend right] & a_3 \arrow[u, dotted]
\end{tikzcd}$$

Under our new formulation of PageRank, $\pi_{a_3} = \frac{\epsilon}{N}$ (not zero). This means that π_{a_4} will be larger than $\frac{\epsilon}{N}$, and π_{a_5} will be larger than π_{a_4}. Thus, we will not have the same problem that we had with naïve PageRank where a_2 was able to increase its PageRank by an unbounded amount, and cause the PageRank of most of the other pages to go to zero. Although PageRank is less vulnerable to collusion than naïve PageRank, it is not completely immune to it.

Special values of ϵ

If $\epsilon = 1$, the PageRank of every page is $\frac{1}{N}$. And if $\epsilon = 0$, then we have naïve PageRank.

Computing PageRank

Since there are billions of web pages, it is not obvious how to go about computing the PageRank for every page. One possibility is matrix inversion since we can rewrite the PageRank equations as a matrix equation. To solve such an equation, the matrix must be inverted, and for billion by billion matrix, this would take a very long time.

Consider the earlier example with two web communities:

$$\begin{tikzcd}
 & b \arrow[dr] & & & q \arrow[dr] & \\
a \arrow[ur] & & c \arrow[ll] & p \arrow[ur] & & r \arrow[ll]
\end{tikzcd}$$

We saw that using PageRank, if $\epsilon = 0.12$, we cannot have a solution of $\pi_a = \pi_b = \pi_c = 0.02$ because $\pi_a = 0.88\pi_c + 0.02$. But if we use these values as a starting point, and then solve equations like $\pi_a = 0.88\pi_c + 0.02$ to obtain new values, we can get closer to the true solution, and in fact this algorithm will converge quickly as we will see.

We will denote the iterative PageRank of page v after t iterations as $\pi_v^{(t)}$. Initially, we set $\pi_v^{(0)} = \frac{1}{N}$. Then, at every iteration of the algorithm, we will use the following formula to compute the new PageRank.

$$\pi_v^{(t)} = (1-\epsilon)\left(\sum_{(w,v)\in E} \frac{\pi_w^{(t-1)}}{d_w}\right) + \frac{\epsilon}{N}$$

After a small number of iterations, the solution will converge to the true PageRank solution.

To prove that the convergence time is small, we will define π_v^* as the true PageRank of v. Then we can define the total error at step t to be:

$$Err(t) = \sum_v |\pi_v^{(t)} - \pi_v^*|$$

Since π_v^* is the true solution, we know that it must satisfy the PageRank equations exactly:

$$\pi_v^* = (1-\epsilon)\left(\sum_{(w,v)\in E} \frac{\pi_w^*}{d_w}\right) + \frac{\epsilon}{N}$$

To find the error, we subtract this from the iterative method equation, and obtain:

$$\pi_v^{(t)} - \pi_v^* = (1-\epsilon)\left(\sum_{(w,v)\in E} \frac{\pi_w^{(t-1)} - \pi_w^*}{d_w}\right)$$

Using the Triangle Inequality, we can get this expression for the error in PageRank v at step t:

$$|\pi_v^{(t)} - \pi_v^*| \leq (1-\epsilon)\left(\sum_{(w,v)\in E} \frac{|\pi_w^{(t-1)} - \pi_w^*|}{d_w}\right)$$

Now we can sum over all v to get the total error. Notice that the page w will occur d_w times on the right hand side, and since there is a d_w on the denominator, these will cancel.

$$Err(t) = \sum_v |\pi_v^{(t)} - \pi_v^*| \leq (1-\epsilon)\left(\sum_w |\pi_w^{(t-1)} - \pi_w^*|\right)$$

We are left with $(1-\epsilon)$ times the total error at time $t-1$ on the right hand side.

$$Err(t) \leq (1-\epsilon)Err(t-1)$$

This shows fast convergence, because the decrease in total error is compounding.