MS&E 235, Internet Commerce

Stanford University, Winter 2007-08

Instructor: Prof. Ashish Goel, Notes Scribed by Shrikrishna Shrin

Lecture 8: Reputation Systems, Page Rank

Google's PageRank

The PageRank algorithm was the original algorithm used by the Google search engine to rank web pages.

Main notion of PageRank

The primary notion of PageRank is that you are highly reputed if other highly reputed entities think highly of you.

Consider an adjacency matrix A such that any element $A_{i,j}$ is

$$= 1 \text{ if web page } i(w_i) \text{ links to web page } j(w_j)$$

$$= 0 \text{ otherwise}$$

Naive PageRank

$$\pi_j = \sum_i A_{i,j} \frac{\pi_i}{D_i}$$

$$\sum_j \pi_j = 1 \text{ (normalization constraint)}$$

Where π_j is the PageRank of web page w_j, $A_{i,j}$ is the adjacency matrix which represents which pages a particular page links to and D_i is the dilution factor or the number of outgoing links from web page i.

The PageRank algorithm is an eigen value method as it can be represented in matrix form as $\pi = P\pi$ where π is an eigen vector of matrix P.

Intuitive explanation of PageRank

Imagine a monkey that is surfing the web. The monkey does not care about the content of a page but randomly clicks on one of the outgoing links on a particular page to go to another page.

Now, suppose there are two links on a page, the monkey clicks on either link with a probability of 0.5. Suppose, there were D links, then the monkey follows each link with a probability of $1/D$. Therefore, the probability that the monkey arrives at a page j from a page i is equal to the probability that the monkey was at page i (ie. π_i) multiplied by the probability that the monkey clicks on the link that leads it to page j (ie. $1/D_i$). Now, the monkey can arrive at page j from all pages that link to it. Therefore, the probability that the monkey is

at page j at any given time (π_j) is equal to $\sum_i A_{i,j} \frac{\pi_i}{D_i}$ which is nothing but the PageRank formula.

Also, $\sum_j \pi_j = 1$ as the monkey never leaves the system.

PageRank of any web page can therefore be thought of as the fraction of time that the monkey spends at that web page.

Problems associated with the naive PageRank algorithm

1. It is easy to game the algorithm. Spamming is relatively easy because you can create many web pages that point to your web page in order to increase the PageRank of your web page.

2. There can be pages with in-links but no out-links in which case the "monkey" will get stuck. This can be solved by allowing the monkey to jump to a random web page in such situations.

3. There can be islands of web pages (ie. web pages such that all the web pages in the island are connected but there are no out links from the island as a whole to any other web page). In such cases, the "monkey" is trapped in the web pages that form an island. What is worse is when there are multiple such islands in which case, the question of where the monkey will get trapped is indeterminate.

PageRank algorithm

In order to solve the problems described in the above section, the "monkey" is allowed at every page to jump to a random web page with a small probability.

Let ϵ be the reset probability. The PageRank equation can then be written as:

$$\pi_j = \frac{\epsilon}{N} + \sum_i \frac{(1-\epsilon)(A_{i,j})}{D_i} \pi_i$$

It is widely believed that ϵ was initially set $\approx 1/7$.

Boosting of PageRank

Consider two web pages 1 and 2 such that web page 1 has only one out link pointing to web page 2 and web page 2 only has one out link pointing to web page 1. In such a case, once the "monkey" arrives at web page 1, it is trapped in a cycle.

Let the monkey be at web page 1. The probability that the monkey visits web page 1 once is 1 as it is already at web page 1. The probability that the monkey visits web page 1 twice is $(1-\epsilon) * \epsilon$, thrice is $(1-\epsilon)^2\epsilon$ and so on.

The expected time the monkey spends on website 1 is therefore obtained as:

$$\epsilon + 2(1-\epsilon)\epsilon + 3(1-\epsilon)^2\epsilon + ...$$

$$= 1/\epsilon$$

We can arrive at this value in another way. The monkey spends the first time unit at web page 1 with probability 1. The probability that it also spends the second time unit at web page 1 is $(1 - \epsilon)$ and so on. We therefore obtain the expected time it spends at web page 1 as:

$$1 + (1 - \epsilon) + (1 - \epsilon)^2 + ...$$
$$= 1/\epsilon$$

The PageRank of web page 1 can therefore be increased by a factor of $1/\epsilon$ by creating such a cycle. Since search engines can detect these cycles easily and penalize websites for resorting to such tactics to boost their PageRank, website owners often resort to cleverer techniques such as increasing the complexity and degree of these cycles and inserting links to other web pages infrequently to make the structure of their web pages closely resemble structures that are found frequently on the web.

Detecting such colluding web pages has been shown to be a computationally hard problem and whether an algorithmic solution exists or not is yet to be determined.

Since it might be possible to observe structures of web pages that have a high PageRank and duplicate the structure to boost the PageRank of your websites, there is also a notion of trust rank which is similar to PageRank except that trust rank is allowed to propagate from a set of trusted pages that have been manually selected as trustworthy websites. Therefore only web pages that are reachable by following out links from the trusted set of websites can obtain a trust rank.