Google's PageRank algorithm powered by linear algebra

Andrew Dynneson

Fall 2010

Abstract

Google's PageRank algorithm ranks the importance of internet pages using a number of factors to be discused, such as backlinking, which can be computed using eigenvectors and stochastic matrices. However, due to the overwhelmingly large number of web-pages available on the internet, another method must be employed which will be a modified power method, which accurately approximates the ranking.

1. Introduction: the basic algorithm

Up until this semester in Linear Algebra, I had mistakenly assumed that Google ranked pages in order of importance based on how much money the host URL was willing to pay for that ranking. To take a line from Schindler's List, "Call it gratuity." Upon professing my ignorance, I discovered that the actual proceedure for determining which listing shows up on the first result pages is actually an incredibly in-depth proceedure using Semantic Analysis and Linear-Algebra. This paper gives an explanation of one aspect of Google's ranking, known as the "Page-Rank Algorithm."

The complete nature of how PageRank works is not entirely known, nor is PageRank in the public domain. Most of the articles that discuss the algorithm indicate that it works by Markov chains. However, the algorithm runs into trouble when there are dangling nodes [2] (pages that do not link to other pages). In other words, there is considerable mystery surrounding the workings of Google's algorithm. For information on the workings of the original algorithm, see [5]. The authors in [5] also provide us with some colorful verbiage, which accentuates the neccessity of PagRank quite nicely:

> The average web page quality experienced by a user is higher than the quality of the average web page. This is because the simplicity of creating and publishing web pages results in a large fraction of low quality pages that users are unlikely to read.

PageRank works by forming a de facto democracy of links, where each link to another page acts as a vote. These votes are weighted according to importance of the website placing the vote, and are scaled according to how many votes a website casts [1,2,3]. Beginning with a crude example, and it will be shown how the majorant with respect to raw votes is usurped once weighting is applied. Consider this elementary model of an internet consisting of only four

websites. Their linking schemes are indicated by arrows. The following examples and algorithm are taken from [3], and the figures are drawn using Geometer's Sketchpad.

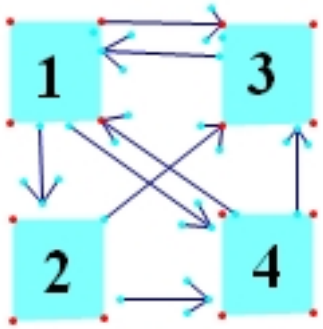

Figure 1

Allow x_k to equal the number of times page k is voted for by the other pages. Then, as can be seen by figure 1, $x_1 = 2,\ x_2 = 1,\ x_3 = 3$ and $x_4 = 2$. In this elementary ranking system, x_3 holds the majority vote. However, this elementary voting system fails to take into account that a link from a website of the highly frequented caliber of msn.com should have more weight than that of an obscure URL that has little actual clout in the way that the Internet is structured.

For example, form the equation $x_1 = x_3 + x_4$, the sum of the values of the two sites voting for page 1. This type of recursive approach is already begining to become incredibly ubiquitious, since by the same token, we would have $x_3 = x_1 + x_2 + x_4$, and so on.

Furthermore, in an ideal model, it is not desirable to have an obscure website artificially gain voting power by having a googol, or other large number, of links to other websites. The way this is rectified is by allowing each website to have exactly one vote, and then dividing that one vote into equal pieces among all of the websites that it links to.

Stepping outside of the four-website model for a moment in order to form a general theory, suppose that the web has n pages, and fixing k, let page j contain a total of n_j links, where one of these links belong to k. Then, add to k's score a value of x_j/n_j. Using the convenient notation of [3], let $L_k \subset \{1, 2, \ldots, n\}$, where L_k is the set of all pages linking to k. Then:

$$x_k = \sum_{j \in L_k} \frac{x_j}{n_j} \tag{1}$$

Returning to the model proposed in Figure 1, $x_1 = x_3 + \frac{x_4}{2}$, $x_2 = \frac{x_1}{3}$ $x_3 = \frac{x_1}{3} + \frac{x_2}{2} + \frac{x_4}{2}$ and $x_4 = \frac{x_1}{3} + \frac{x_2}{2}$. Placing these results into a matrix:

$$\begin{bmatrix} 0 & 0 & 1 & 1/2 \\ 1/3 & 0 & 0 & 0 \\ 1/3 & 1/2 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{2}$$

Next, this matrix is a square, with all of its entries nonnegative, and the sum of the entries in a column add up to 1. Therefore, the matrix is column-stochastic, and so by a proposition in [3], 1 is an eigenvalue. In other words, equation (2) is fully justified solution-wise. By normalizing the solution, deriving our desired importance vector: $[x_1, x_2, x_3, x_4] = \left[\frac{12}{31}, \frac{4}{31}, \frac{9}{31}, \frac{6}{31}\right]$. Observe that page 3, which previously ruled by a simple majority, is now usurped by page 1 once the weighting scheme is applied.

2. Troubleshooting:

There are several problems that arise when using equation (1). One issue is that a web may have dangling nodes. Another possibility is that there may not be a unique solution. In the previous example, where V_1 is the eigenspace for $\lambda = 1$, the $\dim V_1 = 1$, which is desirable. However, it can only be assumed that this is universally true when we are able to travel from one page to any other page in finitely many steps. Take, for example, the web modeled as follows:

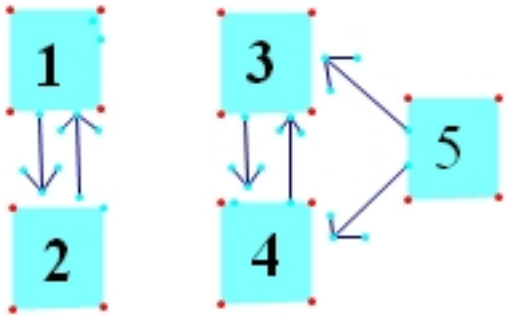

Figure 2

With the corresponding matrix:

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1/2 \\ 0 & 0 & 1 & 0 & 1/2 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

In this case, the $\dim V_1 = 2$, and there are two distinct solutions: $\left[\frac{1}{2}, \frac{1}{2}, 0, 0, 0\right]$ and $[0, 0, \frac{1}{2}, \frac{1}{2}, 0]$. Also, V_1 is a subspace, and there are many additional solutions by taking linear combinations of these two solutions, and so the algorithm would fail at this stage.

Assuming that our web has no dangling nodes, we then may eliminate the ambiguity by letting $\mathbf{S}$ be the $n \times n$ matrix where each entry in the matrix is equal to $1/n$. Then $\mathbf{S}$ is column-stochastic, with respect to $\mathbf{S}$, the $\dim V_1 = 1$, and $\mathbf{M} := (1 - m)\mathbf{A} + m\mathbf{S}$, $0 \leq m \leq 1$ (Originally, Google used $m = 0.15$ [3]). For any $0 \leq m \leq 1$, $\mathbf{M}$ is column-stochastic, and for $0 \neq m$, with respect to $\mathbf{M}$, the $\dim V_1 = 1$, as desired.

Also, in the case of dangling nodes, using this weighting gives such a webpage a score of m/n, however [3] does not describe why this is, nor why this fails to resolve the issue of dangling nodes.

3. The Power Method and Adaptive PageRank: computing the imposible.

Since in effect, n is in the billions of websites, and growing every day, computing the unique eigenvector can be computationally impossible. To bypass this computation, a modified power method is used.

First, since the vector that arises from this method corresponds to the dominant, or largest eigenvalue, the propositions which are presented in [3] are necessary in order to guarantee that the eigenvalues are all less than or equal to 1 with respect to the matrix $\mathbf{M}$.

Also, normalization occurs at each step, starting at a vector $\mathbf{x}_0$ and letting $\mathbf{x}_k = \dfrac{\mathbf{M}\mathbf{x}_{k-1}}{||\mathbf{M}\mathbf{x}_{k-1}||}$. Taking iterations, as k becomes arbitrarily large, then $\mathbf{x}_k$ will begin to give an excellent approximation of the solution vector, thereby ranking websites on the internet.

This process takes approximately 50 iterations for a data-set containing 80 million webpages. The authors in [4] devise a way to speed up the computation further. Since pages with low Page Rank converge sooner in general, by not recalculating for further iterations for pages with low Page-Rank, their algorithm appears to improve the computation speed by as much as 30 percent. With a calculation that typically takes days to compute, and must be computed often, this is a significant amount. The article calls this the Adaptive PageRank method.

4. Conclusion

We now have some idea how Google decides which site you receive when you click the "I'm Feeling Lucky" button, although whether anyone actually uses that option is open to debate. We have seen that PageRank is more than a simple majority. Rather, the ranking of a webpage in the democratic sense is actually a rather delicate procedure involving Stochastic matrices, a weighted average, and successive iterations of our modified power method. This method is also greatly improved by further adaptation, and indeed it is likely that the actual algorithm currently in use, although kept secret, is a marvel of modern mathematics.

Now that we have a working knowledge of how Google ranks pages according to a democratic system of hyper-text links, we gain deeper insight into the workings of Linear Algebra in the esoteric sense, also it will help us to better place our own self-designed websites within the Google search system.

For years, my website did not show up on Google. However, upon beginning my background research in this area, I placed a single link on my profile on Facebook, and within a few days, my site was picked up by one of Google's crawlers. Now my homepage is third from the top of the search results upon typing in my name. I am below Facebook, and my site has finally managed to usurp the news-paper column I wrote for the Free-lance back in 2004.

Bibliography

[1] M.W. Berry and M. Browne, *Understanding Search Engines: Mathematical Modeling and Text Retrieval*, Second Edition, SIAM, Philidelphia, 2005.

[2] M. Bianchini, M. Gori, and F. Scarselli, "Inside PageRank," *ACM Trans. Internet Tech.,* 5 (2005).

[3] K. Bryan and T. Leise, "The $25,000,000,000 Eigenvector: The Linear Algebra Behind Google," *SIAM Review*, 48 (2006).

[4] S. Kamvar, T. Haveliwala, and G. D. Meyer, "Adaptive methods for the computation of PageRank," *Linear Algebra Appl.,* 386 (2004).

[5] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank citation ranking: Bringing order to the Web," Technical Report. Stanford InfoLab, (1999).