Ranking web pages

David F. Gleich and Paul G. Constantine

Purdue University and Colorado School of Mines

Google's™ search engine enables Internet users around the world to find web pages that are relevant to their queries. Early on, Google distinguished its search algorithm from competing methods by combining a measure of a page's textual relevance with a measure of the page's global importance; this latter measure was dubbed *PageRank*.™ Google's PageRank scores help distinguish important pages like the homepage `www.purdue.edu` from its array of subpages.

If we view the web as a huge directed graph, then PageRank scores are the stationary distribution of a particular MARKOV CHAIN [II.xy] on the graph. In the *web graph*, each web page is a node and there is a directed edge from node i to node j if web page i has a hypertext reference, or link, to web page j. A small sample of the web graph from Wikipedia™ pages is shown in Figure 1. The ADJACENCY MATRIX [II.xy] for the graph from the figure is:

$$A = \begin{bmatrix} 0&1&1&1&1&0&1&0&0&0 \\ 1&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&1&1&1&0&0&0 \\ 1&1&0&0&1&1&0&1&0&0 \\ 1&1&1&1&0&0&0&0&1&0 \\ 0&0&1&0&0&0&1&0&1&1 \\ 0&0&0&0&0&1&0&0&0&1 \\ 0&0&0&0&0&0&0&0&1&0 \\ 0&0&0&0&0&0&0&1&0&0 \\ 0&0&0&0&0&0&0&0&0&0 \end{bmatrix} \begin{matrix} \text{PageRank} \\ \text{Google} \\ \text{Adjacency matrix} \\ \text{Markov chain} \\ \text{Eigenvector} \\ \text{Directed graph} \\ \text{Graph} \\ \text{Linear system} \\ \text{Vector space} \\ \text{Multiset} \end{matrix}$$

Google's founders Brin and Page imagined an idealized web surfer with the following behavior. At a given page, the surfer flips a coin with probability α of heads and probability $1-\alpha$ of tails. On heads, the surfer clicks a link chosen uniformly at random from all the links on the page. If the page has no links, then we call it a *dangling node*. On tails, and at dangling nodes as well, the surfer jumps to a page chosen uniformly at random from whole graph. (There are alternative models for handling dangling nodes.) This simple model of a *random surfer* creates a Markov chain on the web graph; transitions depend only on the current page and not the web browsing history. The PageRank vector is the stationary distribution of this Markov chain, which depends on the value of α. For the graph in Figure 1, the transition matrix P for the PageRank Markov chain with $\alpha = 0.85$ is (to two decimal places)

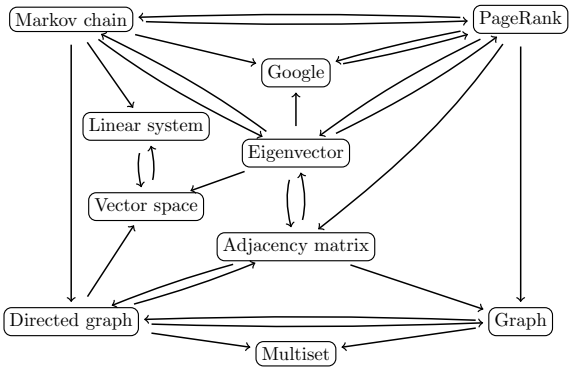

Figure 1: A subgraph of the web from Wikipedia.

$$P = \begin{bmatrix} .02&.19&.19&.19&.19&.02&.19&.02&.02&.02 \\ .86&.02&.02&.02&.02&.02&.02&.02&.02&.02 \\ .02&.02&.02&.02&.30&.30&.30&.02&.02&.02 \\ .19&.19&.02&.02&.19&.19&.02&.19&.02&.02 \\ .19&.19&.19&.19&.02&.02&.02&.02&.19&.02 \\ .02&.02&.23&.02&.02&.02&.23&.02&.23&.23 \\ .02&.02&.02&.02&.02&.44&.02&.02&.02&.44 \\ .02&.02&.02&.02&.02&.02&.02&.02&.86&.02 \\ .02&.02&.02&.02&.02&.02&.02&.86&.02&.02 \\ .10&.10&.10&.10&.10&.10&.10&.10&.10&.10 \end{bmatrix} \begin{matrix} \text{PR} \\ \text{Go} \\ \text{AM} \\ \text{MC} \\ \text{EV} \\ \text{DG} \\ \text{Gr} \\ \text{LS} \\ \text{VS} \\ \text{MS} \end{matrix}$$

The PageRank vector x is

$$x = \begin{bmatrix} 0.08 \\ 0.05 \\ 0.06 \\ 0.04 \\ 0.06 \\ 0.07 \\ 0.07 \\ 0.24 \\ 0.25 \\ 0.06 \end{bmatrix} \begin{matrix} \text{PageRank} \\ \text{Google} \\ \text{Adjacency matrix} \\ \text{Markov chain} \\ \text{Eigenvector} \\ \text{Directed graph} \\ \text{Graph} \\ \text{Linear system} \\ \text{Vector space} \\ \text{Multiset} \end{matrix}$$

To define the PageRank Markov chain generally:

- let $G = (V, E)$ be the web graph;
- let A be the adjacency matrix;
- let $n = |V|$ be the number of nodes;
- let D be a diagonal matrix where D_{ii} is 1 divided by the number of outlinks for page i or 0 if page i has no outlinks;
- let d be an n-vector where $d_i = 1$ if page i is dangling with no outlinks and 0 otherwise; and

- let e be the n-vector of all ones.

Then

$$P = \alpha DA + \alpha/n \cdot de^T + (1-\alpha)/n \cdot ee^T$$

is the PageRank Markov chain's transition matrix. The PageRank vector x is the eigenvector of P^T with eigenvalue 1:

$$P^T x = x.$$

This eigenvector always exists, is nonnegative, and is unique up to scaling because the matrix P is irreducible. By convention, the vector x is normalized to be a probability distribution, $e^T x = 1$. Therefore, x is unique for a given web graph and $0 < \alpha < 1$. The vector x also satisfies the nonsingular linear system

$$(I - \alpha(A^T D + 1/n \cdot ed^T))x = (1-\alpha)/ne.$$

Thus, PageRank is both an eigenvector of the Markov chain transition matrix and the solution of a nonsingular linear system. This duality gives rise to a variety of efficient algorithms to compute x for a graph as large as the web. As the value of α tends to 1, the matrix $I - \alpha(A^T D + 1/n \cdot ed^T)$ becomes more ill-conditioned, and computing PageRank becomes more difficult. However, when α is too close to 1, the quality of the ranking degrades. For the graph in Figure 1, as α tends to 1 the PageRank vector concentrates all its mass on the pair "Linear system" and "Vector space." This happens because this pair is a terminal strong component of the graph. The same behavior occurs in the web graph; consequently, the PageRank scores of important pages such as `www.purdue.edu` are extinguished as $\alpha \to 1$. We recommend $0.5 \leq \alpha \leq 0.99$ and note that $\alpha = 0.85$ is a standard choice.

The canonical algorithm to compute PageRank scores is the POWER METHOD [IV.NLA] applied to the eigenvector equation $P^T x = x$ with the normalization $e^T x = 1$. If we start with $x^{(0)} = e/n$ and iterate:

$$x^{(k+1)} = \alpha A^T D x^{(k)} + \alpha(d^T x^{(k)})/n \cdot e + (1-\alpha)/n \cdot e,$$

then after k steps of this method, $\|x^{(k)} - x\|_1 \leq 2\alpha^k$. This iteration converges quickly when $\alpha \leq 0.99$. With $\alpha = 0.85$, it gives a useful approximation with 10-15 iterations.

PageRank was not the first web ranking to use the structure of the web graph. Shortly before Brin and Page proposed PageRank, Jon Kleinberg proposed hypertext-induced topic search (HITS) scores to estimate the importance of pages in a query-dependent subset of the web. HITS scores are only computed on a subgraph of the web graph with the top 1000 textually relevant pages and all inlink and outlink neighbors within distance 2. The left and right dominant SINGULAR VECTORS [II.SVD] of the adjacency matrix are *hub* and *authority* scores for each page, respectively. An authority is a page with many hubs pointing to it, and a hub is a page that points to many authorities. The former search engine Teoma™ used scores related to HITS.

Modern search engines use complex algorithms to produce a ranked list of web pages in response to a query; scores such as PageRank and HITS may be one component of much larger systems. To judge the quality of a ranking, search engine architects must compare algorithmic results to human judgments of pages' relevance to a particular query. There are a few common measures for such comparison. *Precision* is the percentage of the search engine's results that are relevant to the human. *Recall* is the percentage of all relevant results identified by the search engine. Recall is not often used for web search as there are often many more relevant results than would fit on a top 10 or even top 1000 list. *Normalized discounted cumulative gain* is a weighted score that rewards a search engine for placing more highly relevant documents before less relevant documents. Architects study these measures over a variety of queries to optimize a search engine and choose components of their final ranking procedure. Two active research areas on ranking algorithms include new types of regression problems to automatically optimize ranked lists of results and multi-armed bandit problems to generate personalized rankings for both web search engines and content recommendation services like Netflix.™

Further reading

Langville and Meyer's book contains a complete treatment of mathematical web ranking metrics around 2005. Manning *et al.* give a modern treatment of search algorithms including web search.

While PageRank's influence in Google's ranking may have decreased over time, its importance as a tool to find important nodes in a graph has grown tremendously. PageRank vectors for graphs from biology (Singh *et al.*), chemistry (Mooney *et al.*), ecology (Allesina and Pascual) have given domain scientists important new insights.

Further Reading

1. Allesina, S. and Pascual, M. 2009. Googling food webs: Can an eigenvector measure species' importance for coextinctions? *PLoS Comput. Biol.* **5**, e1000494.
2. Langville, A. N. and Meyer, C. D. 2006. *Google's PageRank and Beyond*. Princeton, NJ: Princeton University Press.
3. Manning, C. D., Raghavan, P., and Schütze, H. 2008. *An Introduction to Information Retrieval*. Cambridge, UK: Cambridge University Press.
4. Mooney, B. L., Corrales, L. R., and Clark, A. E. 2012. MoleculaRnetworks: An integrated graph theoretic and data mining tool to explore solvent organization in molecular simulation. *J. Comput. Chem.* **33**, 853–860.
5. Singh, R., Xu, J., and Berger, B. 2008. Global alignment of multiple protein interaction networks with application to functional orthology detection. *P. Natl. Acad. Sci. USA* **105**, 12763–12768.