Reasoning Behind the Google PageRank Algorithm

[redacted]

May 3, 2019

1 Introduction

With the current connectivity of the world wide web, the effectiveness of a search engine is crucial to the success of a web search engine company. For any web surfer, the expectation is to receive the most relevant and useful answers to his or her query in a timely manner after a simple click of the "search" bottom. In this aspect, Google has been very successful. It is no coincidence that users are able to find what they need solely from the first page of their search result almost every single time. The backbone of the Google search engine could be said to be the Google PageRank, which is one of the algorithms that Google utilizes to sort its search results. Compared to the other Google algorithms, the Google PageRank is by far the most popular. Surprisingly, the reasoning behind the PageRank algorithm is actually a quite simple application of basic probability and linear algebra. In theory, a Google PageRank is just a finite Markov Chain of pages that eventually converges to a steady-state vector.

2 Google PageRank

In the Google PageRank algorithm, the order in which the web pages are ordered is based on each page's importance and number of visits. The importance of a web page is determined by the page's number of incoming and outgoing links [5]. Under the assumption that an incoming link from page A to B serves as an endorsement for B, those pages with a higher number of incoming links are considered more important, and pages that are linked to highly important pages (those with good quality back links) are also considered important [6]. Using the Markov Model, Google assigns a score to each web page and list them in descending order when a search is initiated. All else equal, if two pages have the same number of links, the page with the higher number of incoming links will be assigned a higher score in the end.

2.1 Markov Chain

One of the PageRank algorithms that Google uses for its search engine is the Markov Chain Model, which is a non-deterministic finite state machine with probability-driven transitions [5]. The transitions are stochastic, having the Markov property that their future value, conditional on their present value, is independent of their past values. In other words, for a sequence X taking values in a state space S,

$$\mathbb{P}(X_{n+1} = i_{n+1} | X_0 = i_0, X_1 = i_1, ..., X_n = i_n) = \mathbb{P}(X_{n+1} = i_{n+1} | X_n = i_n)$$

for all $n \geq 0$ and $i_0, i_1, ..., i_{n+1} \in S$ [2]. Under this assumption, future events can be determined solely from information in the present state.

If X is a homogeneous Markov chain, there is a transition matrix $P = (p_{i,j} : i, j \in S)$ with transition probabilities $p_{i,j} = \mathbb{P}(X_{n+1} = j | X_n = i)$, and initial distribution $\lambda = (\lambda_i : i \in S)$, where $\lambda_i = \mathbb{P}(X_0 = i)$ [2]. The initial transition matrix P used in the Google PageRank algorithm for n pages is a column-stochastic matrix: a real non-negative square $n \times n$ matrix such that $p_{i,j} \geq 0$ and each column sums up to 1. Using this model, the probability that a surfer is at a particular page at a point in time can be found using the transition probabilities $p_{i,j}$ that the system moves from page i to j at any time t.

The Google transition matrix (G) is given by

$$G = dP_* + \frac{1-d}{n} \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & 1 & 1 & \dots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \dots & 1 \end{bmatrix},$$

where P_* is a matrix modeling the behavior of a random web surfer such that P is adjusted for dangling nodes, with entries

$$P_{*ij} = \begin{cases} c_{ij}/s_i, & \text{if } s_i \geq 1 \\ 1/n, & \text{if } s_i = 0 \end{cases}$$

for a transition matrix with n web pages [5]. The quantities are defined as follow: s_i is the number of outgoing links from page i, c_{ij} is equal to 1 if there is a hyperlink from page i to j and 0 otherwise, and $d \in [0, 1]$ is the damping factor [5]. The damping factor (usually equal to 0.85 in the Google algorithm) represents the probability that the surfer will follow the assumption of either checking on one of the outgoing links in the current page with equal probability, or jumping to a random page if there is no outgoing link (to remove dangling nodes) [6]. The resulting Google Matrix G is an irreducible stochastic matrix such that for any pair of web pages, a surfer can start at one page, eventually arrive at the other, and then get back to the original page.

2.2 Convergence of Google Matrix

The computation of a pagerank vector given G will continue until it converges or when no further changes can be made. Since the Google transition matrix G is a positive, irreducible column-stochastic matrix such that all its entries are greater than 0, we can apply the Perron-Frobenius Theorem. From the Perron-Frobenius Theorem, it follows that G will eventually converge to a steady-state vector [6].

2.2.1 Perron-Frobenius Theorem

Suppose A is a $n \times n$ positive matrix with spectral radius ρ (largest absolute value of its eigenvalues for a square matrix) [4]. Then $\rho > 0$ and

i) ρ is an eigenvalue of A

ii) ρ has algebraic multiplicity 1

iii) ρ has eigenvector v s.t. $v > 0$

iv) for all other eigenvalues λ of A, $\lambda \neq \rho$, $|\lambda| < \rho$

v) If u is an eigenvector of A (corresponding to any eigenvalue) whose entries are all positive, then u is a scalar multiple of v

Proof. (With reference to proof given in [4])

i) and iii): Suppose $x \in S$, where S is the set of $\mathbb{R}^n$ vectors with non-negative entries and a norm of 1. Then for any x, all entries of Ax are positive. Defining a function

$$L(x) = \min\left\{\frac{(Ax)_i}{x_i} : x_i \neq 0\right\},$$

where the value of $L(x)$ is the scaling factor that x is multiplied by to get Ax. In S, there is a $v \in S$ that results in the largest L value, where $L(v) = \rho$. Since $(Av)_i \geqslant \rho v_i$ for all i, $Av - \rho v \geqslant 0$ and $A(Av - \rho v) > 0$, so $\exists \epsilon > 0, c > 0$ such that $A(Av - \rho v) > \epsilon Av$ and

$$A(Av) > (\rho + \epsilon)Av, A(cAv) \geqslant (\rho + \epsilon)cAv, L(cAv) \geqslant (\rho + \epsilon),$$

which contradicts that the largest L is ρ. Therefore, $Av = \rho v$. Since $v \in S$ is non-negative, Av has all positive entries, so ρv is positive, and it follows that $v > 0$.

ii): Since $1 \leq$ geometric multiplicity of an eigenvalue $\leq$ algebraic multiplicity of an eigenvalue, for the purpose of proving just statement ii), start off by taking for granted that that the geometric multiplicity of ρ is 1. Since A and A^T have the same eigenvalues, suppose w is the eigenvector corresponding to A^T, then $A^T w = \rho w$ and $(A^T w)^T = (\rho w)^T$, so $w^T A = \rho w^T$. Suppose U is the A-invariant subspace of orthogonal complements of W such that $w^T Au = \rho w^T u = 0$, then we can extend the eigenvector v of ρ, where $v \notin U$, into a basis $\mathbb{B} = \{v, b_1, ..., b_{n-1}\}$ of $\mathbb{R}^n$, where $\{b_1, ..., b_{n-1}\}$ is a basis of U. For a similar matrix A' of A with respect to the basis $\mathbb{B}$, the

$$\mathsf{A}' = \left(\begin{array}{c|ccc} \rho & 0 & \cdots & 0 \\ \hline 0 & & & \\ \vdots & & \mathsf{B}_{(n-1)\times(n-1)} & \\ 0 & & & \end{array}\right)$$

characteristic polynomial of A' is $(x - \rho)p_B(x)$, where $p_B(x)$ is the characteristic polynomial of a real matrix B with respect to the basis $\mathbb{B}$. If ρ is an eigenvalue of B, then an eigenvector of A' is the eigenvector v preceded by a zero entry. Since e_1 is also an eigenvector of ρ for A', ρ would have geometric multiplicity of at least 2, which means that A would also have ρ with geometric multiplicity at least 2 since A and A' are similar. This contradicts the initial assumption that the geometric multiplicity of ρ is 1, so the algebraic multiplicity of ρ is 1.

v): If u is a positive eigenvector of A with eigenvalue μ, where $0 < \mu \leqslant \rho$. If $\exists \epsilon$ such that $u' = v - \epsilon u$, then for each entry i, $(Au')_i = \rho v_i - \mu \epsilon u_i \geqslant \rho (v_i - \epsilon u_i) = \rho u'_i$. Therefore, $Au' = \rho u'$ since ρ is known to be the spectral radius of A. Thus, it follows that u, u' are scalar multiples of v and $\mu = \rho$.

iv): For the sake of contradiction, suppose $\mu \neq \rho$ but $|\mu| = \rho$, and that y is an eigenvector of μ with norm 1. Then for $|y| \in \mathbb{C}^n$,

$$(A|y|)_i = \sum_j A_{ij} |y_j| = \sum |A_{ij} y_j| \geqslant \left| \sum_j A_{ij} y_j \right| = |\mu y_i| = \rho |y_i| ,$$

which means that $|y|$ is an eigenvector of ρ and $|y| = v$. Note that each entry A_{ij} is real and positive, so $e^{i\theta} y$ is positive for some θ, which means that it is a positive scalar multiple of v. Since ρ is the only eigenvalue corresponding to the eigenvector v according to v), $\mu = \rho$, which gives a contradiction since $\mu \neq \rho$. Therefore, $|\mu| < \rho$.

□

Specifically for a stochastic matrix, 1 is a distinct eigenvalue, and the absolute value any other eigenvalue is less than 1, as illustrated below [3].

For the row-stochastic matrix G^T, $G^T v = 1v$ for $v = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$, so 1 is an eigenvalue of G^T. Since G^T and G have the same characteristic polynomial, it follows that they have the same eigenvalues [1], which means 1 is also an eigenvalue of G. For any other eigenvalue λ of G^T with eigenvector u so that $G^T u = \lambda u$, if u_j is the entry with the largest absolute value, the j^{th} entry of the equation satisfies

$$|\lambda| \cdot |u_j| = \left| \sum_{i=1}^{n} g_{ij} u_i \right| \leq \sum_{i=1}^{n} g_{ij} \cdot |u_i| \leq \sum_{i=1}^{n} g_{ij} \cdot |u_j|$$

for entries $g_{ij} \in G$ [3]. Since the columns of G sums up to 1, $|\lambda| \cdot |g_j| \leq 1 \cdot |g_j|$, so it follows that $|\lambda| \leq 1$. Applying the Perron-Frobenius Theorem, 1 is the spectral radius ρ of G with multiplicity 1, and so $|\lambda| < 1$ for any other eigenvalue λ as a consequence of the theorem.

In particular, there is a unique, positive steady-state vector that is an eigenvector corresponding to the eigenvalue 1, according to the Perron-Frobenius Theorem. This is the pagerank vector that the Google matrix G will eventually converge to [6].

2.2.2 PageRank Vector

For the sake of simplicity, let's assume that the eigenvalues of the Google Matrix G are distinct. In other words, if G is an $n \times n$ matrix, then there are n eigenvalues $\lambda_1, ..., \lambda_n$ of G (where $\lambda_1 = 1$), with corresponding linearly independent eigenvectors $v_1, ..., v_n$ (normalized so that $\|v_i\| = 1$). Therefore, if G is the matrix of an operator $T \in \mathcal{L}(V)$, then $V = E(\lambda_1, T) \oplus \cdots \oplus E(\lambda_m, T)$ and T is diagonalizable [1]. Hence, the Google Matrix can be written as $G = PDP^{-1}$, where P is an invertible matrix with columns consisting of the eigenvectors of G and D is the diagonal matrix of G with eigenvalues $\lambda_1, ..., \lambda_n$ along its diagonal.

Suppose there is an initial pagerank vector x_0, then after k iterations,

$$G^k x_0 = (PDP^{-1})^k x_0 = PD^k P^{-1} x_0.$$

Since all eigenvalues except for $\lambda_1 = 1$ have absolute value less than 1, as k approaches infinity,

$$\lim_{k\to\infty} D^k = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix},$$

which means that the pagerank vector converges to a scalar multiple cv_1 of the steady-state vector v_1 [6].

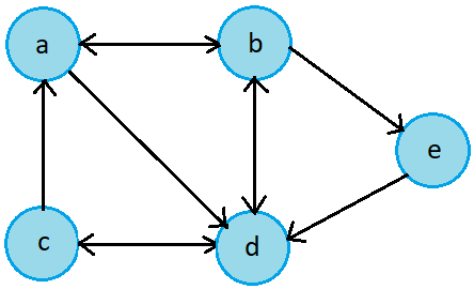
$$\lim_{k\to\infty} G^k x_0 = \mathbf{P} \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix} \mathbf{P}^{-1}\mathbf{x}_0 = \begin{bmatrix} v_1 & 0 & 0 & 0 & 0 \end{bmatrix} P^{-1}x_0 = cv_1$$

After normalizing the entries of v_1, the resulting entries v_{1i} for $i = 1, ..., n$ can be interpreted as the final probability distribution for each page i. Using this result, the score of each page is then determined [6].

Note that even if the eigenvalues of G are not distinct, i.e. the multiplicity of some λ is greater than 1, the process still applies. In this case, if working on a finite-dimensional complex vector space, G could be diagonalized by finding the Jordan Canonical Form of G.

2.3 Example

As a simple example of how the process works, suppose that there are 5 web pages that a surfer can go to. The web graph with the links displayed is shown below.



The matrix P with respect to this web graph is

$$P = \begin{bmatrix} 0 & 1/3 & 1/2 & 0 & 0 \\ 1/2 & 0 & 0 & 1/2 & 0 \\ 0 & 0 & 0 & 1/2 & 0 \\ 1/2 & 1/3 & 1/2 & 0 & 1 \\ 0 & 1/3 & 0 & 0 & 0 \end{bmatrix}.$$

After adjusting for dangling nodes,

$$P_* = \begin{bmatrix} 0 & 1/3 & 1/2 & 0 & 1/5 \\ 1/2 & 0 & 0 & 1/2 & 1/5 \\ 0 & 0 & 0 & 1/2 & 1/5 \\ 1/2 & 1/3 & 1/2 & 0 & 1/5 \\ 0 & 1/3 & 0 & 0 & 1/5 \end{bmatrix}, G = \begin{bmatrix} 0.030 & 0.313 & 0.455 & 0.030 & 0.200 \\ 0.455 & 0.030 & 0.030 & 0.455 & 0.200 \\ 0.030 & 0.030 & 0.030 & 0.455 & 0.200 \\ 0.455 & 0.313 & 0.455 & 0.030 & 0.200 \\ 0.030 & 0.313 & 0.030 & 0.030 & 0.200 \end{bmatrix}.$$

Using an initial pagerank vector x_0 with all 1's as it's entries, the resulting pagerank vector from the first 8 iterations using this Google Matrix is shown below.

Iteration	x_a	x_b	x_c	x_d	x_e
0	0.200000	0.200000	0.200000	0.200000	0.200000
1	0.205667	0.234000	0.149000	0.290667	0.120667
2	0.180138	0.261455	0.174047	0.267547	0.116813
3	0.197907	0.240124	0.163566	0.274466	0.123937
4	0.188620	0.251828	0.167717	0.272730	0.119105
5	0.192879	0.246322	0.166158	0.273042	0.121599
6	0.191080	0.248688	0.166715	0.273054	0.120463
7	0.191794	0.247736	0.166527	0.273003	0.120940
8	0.191525	0.248099	0.166586	0.273038	0.120752

The eigenvector v_1 of G is calculated (using Python) to be

$$v_1 = \begin{bmatrix} 0.191597 \\ 0.248001 \\ 0.166573 \\ 0.273026 \\ 0.120804 \end{bmatrix}$$

after normalization, which has entries that are very close to the values of the pagerank vector after 8 iterations. Based on these results, page d would be ranked the highest.

3 Conclusion

Probability is a branch of mathematics with many applications in important aspects of the world. With probability, an insight into the future could be gained by attempting to predict future events. Some of the applications of probability include actuarial science in risk management, financial risk assessments, analyzing biological and ecological trends (ex: Punnett squares), and many more. Even the browser that internet users rely on the most, Google, depends on probability and the theory of stochastic processes for its algorithms to function. To obtain a better view of the important processes of the world, a basic understanding of the theories of probability is crucial.

References

[1] Axler, S. (2015). Linear Algebra Done Right (Third ed.). San Francisco, CA: Springer.

[2] Grimmett, G., & Welsh, D. (2014). Probability: An Introduction (Second ed.). Oxford: Oxford University Press.

[3] Margalit, D., & Rabinoff, J. (2018, November 18). Interactive Linear Algebra [1553]. Retrieved May 3, 2019, from https://textbooks.math.gatech.edu/ila/1553/stochastic-matrices.html

[4] Quinlan, R. (2018). Proof of the Perron-Frobenius Theorem. Retrieved May 3, 2019, from http://www.maths.nuigalway.ie/ rquinlan/linearalgebra/

[5] Rai, P., & Lal, A. (2016). Google PageRank Algorithm: Markov Chain Model and Hidden Markov Model. International Journal of Computer Applications, 138(9), 9-13. doi:10.5120/ijca2016908942

[6] Shum, K. (2013, April 3). Notes on PageRank Algorithm. Retrieved May 3, 2019, from http://home.ie.cuhk.edu.hk/ wkshum/papers/pagerank.pdf