On the Google Search
25 Billion Dollar Eigenvector

T.T.Moh

The title of my talk is a copy of the title of an article by K. Bryan and T. Leise (however they used a title *\$ 25,000,000,000 Eigenvector,*). I think it is fun. BTW, they did not copy-right it.

What we will discuss is the method to rank many elements (sometimes $10,000$ websites). The principle involved in the following discussions can be applied to the ranking of teams (for instance, Olympic Games, college football teams, etc.), or search engine for websites (Google search engine and Baidu search engine, etc.). We shall give a discussion about the mathematics involved and the speed of solving the ranking problems,

First before we get to the problem of ranking a large number of elements, we will have a simple example.

The Ranking Problem of A Chess Tournament:

Ranking of Chess Players: Let us consider a round robin chess tournament for six players $P_1, P_2, P_3, P_4, P_5, P_6$. We use their records to form the following matrixr,. where $a_{ij} = 1$ if P_i defeats P_j, and in this situation $a_{ji} = 0$. If P_i ties with P_j, then $a_{ij} = a_{ji} = 0.5$. We always take $a_{ii} = 0.5$. Let the results of this tournament be represented by the following matrix of the relative strengths,

$$A = \begin{bmatrix} 0.5, & 1, & 1, & 0, & 1, & 1 \\ 0, & 0.5, & 0, & 1, & 1, & 0 \\ 0, & 1, & 0.5, & 1, & 1, & 1 \\ 1, & 0, & 0, & 0.5, & 0, & 0 \\ 0, & 0, & 0, & 1, & 0.5, & 1 \\ 0, & 1, & 0, & 1, & 0, & 0.5 \end{bmatrix}$$

At the beginning we are ignorant of the relative strength of each player and let e_1 be the vector $[1, 1, 1, 1, 1, 1]^T$ to assume their equal strengths. Then we may take e_2 as follows to indicate their temporary strengths after the tournament,

$$e_2 = Ae_1 = \begin{bmatrix} 0.5, & 1, & 1, & 0, & 1, & 1 \\ 0, & 0.5, & 0, & 1, & 1, & 0 \\ 0, & 1, & 0.5, & 1, & 1, & 1 \\ 1, & 0, & 0, & 0.5, & 0, & 0 \\ 0, & 0, & 0, & 1, & 0.5, & 1 \\ 0, & 1, & 0, & 1, & 0, & 0.5 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 4.5 \\ 2.5 \\ 4.5 \\ 1.5 \\ 2.5 \\ 2.5 \end{bmatrix}$$

We may take e_2 as the ranking vector, and conclude the order of palyers as $P_1 = P_3 > P_2 = P_5 = P_6 > P_4$. This kind of ranking is used in newspapers and Olympic Games. However, there are two problems: (1) the first player and the third player have the same score 4.5. Does the fact that the first player defeated the third player at their head-on contest mean something? (2) The fourth player defeated a strong player (the first player), does it mean something?

To amend the situation, we shall consider the strengths of all players as indicated by temporary vector $e_2 = Ae_1$, and adopt the rule that (1) if player P_i defeats player P_j, then P_i adds P_j's strength to his/her own strength, and (2) if player P_i ties with player P_j, then P_i adds half of P_j's strength to his/her own strength. In the comparison of all college football teams in US, this step is called *consider the schedules* or *consider the strength of the oponents*. We then compute $e_3 = Ae_2 = A^2e_2$. We have

$$e_3 = Ae_2 = A^2e_1 = \begin{bmatrix} 0.25, & 3, & 1, & 4, & 3, & 3 \\ 1, & 0.25, & 0, & 2, & 1, & 1 \\ 1, & 2, & 0.25, & 4, & 2, & 2 \\ 1, & 1, & 1, & 0.25, & 1, & 1 \\ 1, & 1, & 0, & 2, & 0.25, & 1 \\ 1, & 1, & 0, & 2, & 1, & 0.25 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 14.25 \\ 5.25 \\ 11.25 \\ 5.25 \\ 5.25 \\ 5.25 \end{bmatrix}$$

We may now use e_3 as the new temporary relative strength vector and use it as the ranking vector. At this iteration, the first player is pulled apart from the third player, and the fourth player gets an equal rank iwith the remaining players. Note that this is the way that the "computer ranking" works for the college football teams in USA. If we carry the argument one step further, we find $e_4 = Ae_3 = A^3e_1$ as follows,

$$e_4 = Ae_3 = A^2e_2 = A^3e_1 =$$
$$\begin{bmatrix} 4.125, & 5.75, & 0.75, & 12, & 5.75, & 5.75 \\ 2.5, & 2.125, & 1, & 3.25, & 1.75, & 2.5 \\ 4.5, & 4.25, & 1.125, & 8.25, & 4.25, & 4.25 \\ 0.75, & 3.5, & 1.5, & 4.125, & 3.5, & 3.5 \\ 2.5, & 2.5, & 1, & 3.25, & 2.125, & 1.75 \\ 2.5, & 1.75, & 1, & 3.25, & 2.5, & 2.125 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \approx \begin{bmatrix} 34 \\ 13 \\ 27 \\ 17 \\ 13 \\ 13 \end{bmatrix}$$

We can show that if we carry on the argument further, the order $P_1 > P_3 > P_4 > P_2 = P_5 = P_6$ will remain the same. We may conclude that the preceding order is correct.

Note that A is a *non-negative matrix* (see the definition below) and A^k is a *positive matrix* (see the definition below) for $k \geq 3$.

Using Matlab, we found the largest eigenvalue of A is $\lambda = 2.6106$ which is a **positive** real number, and the ratio of the largest eigenvalue over the next largest absolute value of eigenvalues is 1.8596, which shows that $lim_{k\mapsto\infty}(A/\lambda)^k v$ converges fast for any random vector v. Let $lim_{k\mapsto\infty}(A/\lambda)^k v = u$, then we have $(A/\lambda)u = u$, thus u is an eigenvector of (A/λ). Furthermore, either u or $-u$ is **positive** and can be treated as the ranking vector which we shall use eventually. (see below)

□

The principle involved in the above discussions can be applied to other duel games and explained by **Perron theorem**.

Before we discuss the statement of **Perron theorem**, let us give the following definitions.

Definition 1: Let $M = (m_{tj})$ be a real matrix. We say M is *positive*, $M > 0$, iff $m_{tj} > 0$ for all t, j.

□

Definition 2: Let $M = (m_{tj})$ be a real matrix. We say M is *non-negative*, $M \geq 0$, iff $m_{tj} \geq 0$ for all t, j.

□

Notations: Let $M = (m_{tj})$ be a square matrix. Let $\sigma(M)$ be the set of eigenvalues of M. Let $\rho(M)$ be the largest absolute value of all eigenvalues. Let $|M| = (|m_{tj}|)$. Especially, $|v|$ is defined for any vector v.

□

We have the following important **Perron Theorem** (1907) for positive square matrices, which is later generalized to **Perron-Frobenius Theorem** for some non-negatve matrices by Frobenius (1912). In the following statements, we only use the matrix A to act from left as Av. Similar statements also work for the matrix A to act from right as $v^T A$. Since both ways share the same characteristic polynomial, the eigenvalues from both ways must be the same. The only thing we have to pay attention is that for a fixed eignevalue r, the left eigenspace associated with r may be different from the right eigenspace associated with the same eigenvalue r.

Perron Theorem: Let $M = (m_{tj})$ be a real positive square matrix. Then we have,
(1): Let $r = \rho(M)$, then $r > 0$ and $r \in \sigma(M)$. Further, $r >$ the absolute value of any other eigenvalue.
(2): The eigenvalue r has a **positive** vector v as its associated eigenvector. Any other eigenvalue has **no** non-negative associated eigenvector.
(3): The eigenvalue r is a **simple root** of the characteristic polynomial of M.

Proof: (1) If $r = 0$, then M is nilpotent, which is not true. Let us define a set $S = \{s : Mv \geq sv \ for \ some \ v \ \ non\text{-}negative \ \ and \ \ \neq 0\}$. Clearly the set S is non-empty and has some positive elements. For instance, let $v = [1, 1, \cdots, 1]^T$ and $s = min(\{\sum_j m_{tj}\})$. Then $s > 0$ and $Mv \geq sv$. Let $r = sup\overline{S}$. We claim $r \in \sigma(M)$, i.e., r is an eigenvalue.

By considering only the vector v such that $||v|| = 1$ for any norm $||\ ||$, we have a compact set. Then we pick a subsequence v_i such that $Mv_i \geq \lambda_i v_i$ where $\lambda_i :\mapsto r$ with the further property that $v_i :\mapsto v$, and we have $Mv \geq rv$. If it is an equality, then r is an eignevalue. Let us assume that it is not an equality (which does not mean that it is a straight inequality). Let $u = Mv - rv$. Then u is non-negative and non-zero. Therefore $0 < Mu = M(Mv) - r(Mv)$. Let $w = Mv$, we have $Mw > rw$. We may increase the value of r by ϵ, and the preceding inequality would still work. It contradicts the assumption that r is the maximal possible one. Hence we show that r is an eigenvalue.

Let λ be any other eigenvalue (i.e., $\lambda \neq r$) with associated eigenvector z. Then we have $Mz = \lambda z$. Take the absolute values on both sides, we have $M|z| \geq |Mz| = |\lambda||z|$. Therefore $|\lambda| \in S$ and $|\lambda| \leq r$. Suppose $|\lambda| = r$ and we must have $M|z| = |\lambda||z|$ (otherwise,by a previous argument, r can be increased, a contradiction). Hence $M|z| = |Mz|$, $\sum_j m_{tj}|z_j| = |\sum_j m_{tj} z_j|$. The preceding equation happens only if $z_j = e^{\theta i}|z_j|$ for all j. Furthermore, we have $Me^{\theta i}|z| = \lambda e^{\theta i}|z|$, thus $M|x| = \lambda|z|$. Therefore, $\lambda = |\lambda|$. We conclude $\lambda = r$ which contradicts with the assumption that λ is distinct from r.
(2) We have $Mv = rv$ for some non-negative v. The left side Mv is a

positive vector, hence the right side rv is a positive vector, thus v is positive. Similarly M has a positive left eigenvector w^T associated with r. Let u be an (right) eigenvector associated with another eigenvalue λ. Then we have

$$rw^T u = w^T M u = \lambda w^T u \qquad and\ r \neq \lambda$$

hence

$$w^T u = 0$$

Since w^T is positive, then $w^T u$ is positive.A contradiction. Therefore (2) is verified.
(3) Let us make some simplifications. We replace M by M/r, and assume $\rho(M) = 1$. Furthermore, we may select the basis so that M is in the Jordan canonical form J. In other words, we may assume that

$$J = P^{-1}MP = \begin{bmatrix} J_1, & 0, & 0, & 0, & 0 \\ 0, & J_2, & 0, & 0, & 0 \\ 0, & 0, & \cdot & 0, & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 0, & 0, & 0, & 0, & J_h \end{bmatrix}.$$

where J_ts are Jordan blocks. It is easy to see that

$$J^k = P^{-1}M^kP = \begin{bmatrix} J_1^k, & 0, & 0, & 0, & 0 \\ 0, & J_2^k, & 0, & 0, & 0 \\ 0, & 0, & \cdot & 0, & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 0, & 0, & 0, & 0, & J_h^k \end{bmatrix}.$$

We assume that J_1 is associated with eigenvalue 1 with the largest size m. We claim that $m = 1$. Let us assume that $m > 1$. Note that we have $M = PJP^{-1}$. Let $v = [v_1, v_2, \cdots, v_n]^T$ be a positive eigenvector associated with 1. Clearly we have $M^k v = v$ for all integer $k \geq 1$. Note that $min_t\{v_t\} = c > 0$. Let $||\ ||_\infty$ be the norm defined as

$$||B||_\infty = max\{\sum_j |b_{tj}|\}$$

Then $||v||_\infty = max\{|v_t|\}$. Let us compute $||J_1^k||_\infty$ for J_1 associated with eigenvalue 1 and of size $m > 1$. We have

$$J_1^k = \begin{bmatrix} 1, & 0, & 0, & 0, & 0 \\ k, & 1, & 0, & 0, & 0 \\ \cdot, & k, & 1 & 0, & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot, & \cdot, & \cdot, & k, & 1 \end{bmatrix}.$$

It follows easily that $||J^k||_\infty :\mapsto \infty$ as $k \mapsto \infty$. We have $||J^k||_\infty = ||P^{-1}MP||_\infty \leq ||P^{-1}||_\infty ||M^k||_\infty ||P||_\infty$, which means

$$||M^k||_\infty \geq \frac{||J^k||_\infty}{||P^{-1}||_\infty ||P||_\infty} \mapsto \infty.$$

Let $M^k = (m_{tj}^{(k)})$. Recall that $M^k v = v$. Then we have

$$||v||_\infty = max\{|v_t|\} = ||M^k v||_\infty = max\{\sum_j m_{tj}^{(k)} |v_j|\} \geq$$

$$max\{\sum_j m_{tj}^{(k)}\} min\{|v_t|\} = ||M^k||_\infty c \mapsto \infty$$

which is impossible.

We conclude that all Jordan blocks associated with 1 are 1×1. We want to show that there is only one Jordan block associated with 1. Suppose that there are two. Let u, v be linearly independent eigenvectors. We may assume that t-th component v_t of v is not zero. Let $w = u - (u_t/v_t)v$. Then w is also an eigenvector associated with 1. We have $Mw = w$. Take absolute values on both sides, we have $|M||w| \geq |Mw| = |w|$. If the inequality is not equal, we may find a larger eigenvalue, which is impossible. Therefore, we must have $M|w| = |w|$. Since the left hand is positive, so must be the right hand side, which is impossible as we know that the t-th component is zero. □

Definition 3: The number r in the preceding theorem is called the **Perron root**. The unique eigenvector $v = [v_1, v_2, \cdots, v_n]^T$ associated with r such that $\sum_t v_t = 1$ is called the **Perron vector**.

□

Although it will not be used in our discussions of Google, we will mention the generalization made by Frobenius for the completeness. Frobenius found a class of non-negative matrices such that the statements of Perron theorem are valid. A square matrix said to be irreducible if the following definition is satisfied.

Definition 4: A matrix $A_{n\times n}$ is said to be *irreducible* iff there is no permutation matrix P such that

$$P^{-1}AP = \begin{bmatrix} X, & Y \\ 0, & Z \end{bmatrix}$$

where X, Z are square matrices.

$\square$

Definition 5: A matrix $A_{n\times n}$ is said to be *primitive* iff A^k is positive for some positive integer k.

$\square$

Proposition 1: Every primitive matrix is irreducible.

Proof: It is sufficient to prove that the k-th power of a reducible (i.e., not irreducible) matrix A is reducible, which is trivial.

$\square$

We will state the following proposition without giving a proof. Interesting reader may consult C. Meyer: Chapter 8 of *Matrix Analysis and Applied linear algebra*, SIAM, http://www.matrixanalysis.com/Chapter8.pdf.

Proposition 2: An irreducible matrix $A = (a_{tj})$ is a primitive matrix iff a_{tt} is not 0 for some t.

$\square$

We have the following theorem,

Perron-Frobenius Theorem: Let $M = (m_{tj})$ be a real irreducible non-negative square matrix. Then we have,
(1): Let $r = \rho(M)$, then $r > 0$ and $r \in \sigma(M)$. We have r is a simple root of the characteristic polynomial.
(2): The eigenvalue r has a positive vector v as its associated eigenvector. If an eigenvalue λ with $\lambda \neq r$, then it has no non-negative associated eigenvector.
(3): Furthermore if M is primitive, then the root r is the only eigenvalue which has the largest absolute value.

Proof: We shall only prove the case that A is primitive. For the general case, see C. Meyer's book Chapter 8 of *Matrix Analysis and Applied linear algebra*, SIAM, http://www.matrixanalysis.com/Chapter8.pdf. Applying Perron theorem to M^k. We have the above statement for M^k. We have to establish a relation between the eigenvalues of M and M^k. Let us select a suitable basis for the vector space so that M is expressed in it Jordan canonical form. Then it is easy to see that the diagonal items are the eigenvalues $\{\lambda_t\}$ of M. Furthermore, the diagonals of M^k will be $\{\lambda_t^k\}$. It follows that the characteristic polynomial of M^k is $\prod(\lambda - \lambda_t^k)^{n_t}$, which establishs a bijictive correspondent between the set of eigenvalues of $\{\lambda_t\}$ of M and the eigenvalues $\{\lambda_t^k\}$ of M^k.

The only thing we have to prove is that there is a positive eigenvector v associated with the largest eigenvalue r. Pass to M^k, we know that v is a eigenvector associated with the largest eigenvalue r^k of M^k. Hence it must be either positive or $e^{i\theta}v$ is positive, for some θ. So r has a positive associated eigenvector.

□

We have the following definition,

Definitin 6: A non-negative square matrix $A = (a_{tj})$ is called a *stochastic matrix* iff every column adds to 1, i.e.,

$$\sum_t a_{tj} = 1 \; for \; all \; j$$

A vector u is called a *stochastic vector* iff it is a stochastic matrix. □

We have the following proposition for a stochastic matrix.

Propositioni 3: Let $A = (a_{tj})$ be a irreducible stochastic matrix. Then 1 is its largest eigenvalue.

Proof: Let $v = [1, 1, \cdots, 1]$. Then clearly $vA = v$. Therefore 1 is an eigenvalue of A and with a (left) positive eigenvector v. It follows from Perron-Frobenius Theorem that 1 is the largest eigenvalue of A for all vectors from the right.

□

PageRank

Note that any non-negative matrix A without a zero column can be *normalized* to a stochastic matrix, i.e., replace every column by the new column by dividing the old column by its sum. Now we have enough Mathematical background to discuss the Google search. (1) Once the user inputs some keywords, the Google Engine collects all website $W_1, W_2, \cdots, W_n$ (In generali, it assumes that $n \leq 10,000$) with those keywords (this is common to all search engine, and we will not discuss it). (2) the Google engine ranks all collected websites. For this purpose, let us precede as follows.

Let us form a matrix $A = (a_{tj})$, where a_{tj} = the number of references from website W_j to website W_t (it is our way to count *citations*). Then we normalize every non-zero column by replacing a_{tj} by $a_{tj}/\sum_t a_{tj}$ and still call the matrix $A = [a_1, a_2, \cdots, a_n]$, where a_j are the column vectors of A. We now define a new matrix $B = [b_1, b_2, \cdots, b_n]$ where b_j are the column vectors of B as follows: if $a_j = 0$, then $b_j = [1/n, 1/n. \cdots, 1/n]^T$; if $a_j \neq 0$, then $b_j = 0$. Then $A + B$ will be a stochastic matrix. The trouble is that

$A + B$ is only non-negative, while may not be primitive. Anyway, it is troublesome to check if it is primitive. We may speculate the way Google uses to proceed. (1), This is very likely to be the way Google uses. Google creates another $n \times n$ stochastic matrix Q named *taste*, it is a random matrix and take a random vector u, and formulate $C = x(A+B) + (1-x)Q$ where $0 < x < 1$. Then C is primitive for x nearing 1 (in fact, Sergey Brin and Larry Page state that it is good to take $x = 0.85$) and we know that the largest eigenvalue is 1. Now the work is to find the all-important eigenvector (Perron vector) v associated with 1. The values of components of v will give us the ranking of the websites. We shall find the Perron vector v using the following power methed,

Let us use the notations of a basis $\{v = w_1, w_2, \cdots, w_n\}$ such that

$$Cw_1 = w_1 \ \ or \ \ Cw_t = \lambda_j w_t \ \ or \ Cw_t = \lambda_j w_t + w_{t+1} \ where \ |\lambda_j| < 1$$

Then we have

$$C^2 w_1 = w_1 \ or \ C^2 w_t = \lambda_j^2 w_t \ or \ C^2 w_t = \lambda_j^2 w_t + c_1^2 \lambda_j w_{t+1} + \lambda_j w_{t+2} \ where \ |\lambda_j| < 1$$

It is easy to see that

$$C^k w_1 = w_1 \ or \ C^k w_t = \lambda_j^k w_t \ or \ C^k w_t = \lambda_j^k w_t + c_1^k \lambda_j^{k-1} w_{t+1} + \cdots \ where \ |\lambda_j| < 1$$

We can easily conclude that

$$lim_{k \mapsto \infty} C^k w_1 = w_1 \qquad lim_{k \mapsto \infty} C^k w_t = 0 \ for \ all \ t > 1$$

Let us randomly pick up a vector $u = a_1 w_1 + w_2 v_2 + \cdots + a_n w_n$ with $a_1 \neq 0$. Then by the above argument, we have $lim_{k \mapsto \infty} C^k u = a_1 w_1$. After normalizing $a_1 w_1$, we find the Perron vector $w_1 = v$. Hence we have the Google ranking.

According to Larry Page (an inventor of PageRank, the other inventor is Sergey Brin), starting with a random vector u it suffices to take $k = 50$ to get a good approximation of $a_1 w_1$ for the ranking purpose. There are many fast ways of computing C^k in computational mathematics.

Or (2) We take a matrix O such that $o_{tj} = 1/n$ for all t, j, a vector $u = (1/n, 1/n, \cdots, 1/n)^T$, and $D = x(A+B) + (1-x)O$. Similarly, we take $x = 0.85$. Note that u is a stochastic vector, and Dv is stochastic iff D, v are stochastic and $Ov = u$ if v is stochastic. Let $u_1 = u$ and inductively

$$u_{m+1} = Du_m = x(A+B)u_m + (1-x)Ou_m = xAu_m + xBu_m + (1-x)Ou_m$$

$$= xAu_m + xBu_m + (1-x)u.$$

since it is easy to see that inductively all u_m's are stochastic. In the preceding expression, the first term xAu_m is with a *sparse* matrix A where most entries are zero. It is likely that there are only n non-zero entries (we assume that among the 10,000 websites, there are only about 10,000 cross references), so to compute it out, first we pre-compute xA and xB (it will used throughtout the computing process), likely we need only n multiplications for the computation of xA, since xB is a matrix with 0 or x/n (for those columns corresponding to 0 columns of A only), we need only 1 multiplication, then we compute $(xA)u_m$. With xA known, then only needs n multiplications. The second term xBu_m is with the columns of B either 0 or u. It is easy to see that $xBu_m = (a, a, \cdots, a)^T$, where $a = x/n(\sum_{t_j} u_{t_j})$ where t_j runs through only those 0 columns of A and u_{t_j} are the entries of u_m which corresponding to t_j and there are at most 1 multiplications needed. To find u_{m+1} from u_m we need $n+1$ multiplications plus $n+1$ multiplications of pre-computations (which can be used again and again), Note that the number of multiplications is linear in n. For instance let $n = 10,000$ and $m = 50$. Then we need $510,051$ multiplications to find the Google rank. It is a fast job.

Note that the value of x does not affect the speed of computation, we may take x to be as close to 1 as possible, say $x = 0.99$.

$\square$