The Mathematics Behind Google’s PageRank

Ilse Ipsen

Department of Mathematics
North Carolina State University
Raleigh, USA

Joint work with Rebecca Wills
Two Factors

Determine where Google displays a web page on the Search Engine Results Page:

1. PageRank (links)
   A page has high PageRank if many pages with high PageRank link to it

2. Hypertext Analysis (page contents)
   Text, fonts, subdivisions, location of words, contents of neighbouring pages
PageRank

An objective measure of the citation importance of a web page [Brin & Page 1998]

- Assigns a rank to every web page
- Influences the order in which Google displays search results
- Based on link structure of the web graph
- Does not depend on contents of web pages
- Does not depend on query
More PageRank More Visitors
PageRank

... continues to provide the basis for all of our web search tools

http://www.google.com/technology/

- “Links are the currency of the web”
- Exchanging & buying of links
- BO (backlink obsession)
- Search engine optimization
Overview

- Mathematical Model of Internet
- Computation of PageRank
- Is the Ranking Correct?
- Floating Point Arithmetic Issues
Mathematical Model of Internet

1. Represent internet as graph
2. Represent graph as stochastic matrix
3. Make stochastic matrix more convenient ⇒ Google matrix
4. dominant eigenvector of Google matrix ⇒ PageRank
The Internet as a Graph

Link from one web page to another web page

Web graph:

Web pages = nodes
Links = edges
The Web Graph as a Matrix

$S = \begin{pmatrix}
0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 \\
0 & 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 
\end{pmatrix}$

Links = nonzero elements in matrix
Properties of Matrix $S$

- **Row $i$ of $S$:** Links from page $i$ to other pages
- **Column $i$ of $S$:** Links into page $i$
- $S$ is a stochastic matrix:
  - All elements in $[0, 1]$
  - Elements in each row sum to 1
- Dominant left eigenvector:
  \[
  \omega^T S = \omega^T \quad \omega \geq 0 \quad \|\omega\|_1 = 1
  \]
- $\omega_i$ is probability of visiting page $i$
- But: $\omega$ not unique
Google Matrix

Convex combination

\[ G = \alpha S + (1 - \alpha)11v^T \]

- Stochastic matrix \( S \)
- Damping factor \( 0 \leq \alpha < 1 \)
  e.g. \( \alpha = .85 \)
- Column vector of all ones \( 11 \)
- Personalization vector \( v \geq 0 \quad ||v||_1 = 1 \)

Models teleportation
PageRank

\[ G = \alpha S + (1 - \alpha) \mathbf{1} \mathbf{v}^T \]

- \( G \) is stochastic, with eigenvalues:
  \[ 1 > \alpha |\lambda_2(S)| \geq \alpha |\lambda_3(S)| \geq \ldots \]
- Unique dominant left eigenvector:
  \[ \pi^T G = \pi^T \quad \pi \geq 0 \quad \| \pi \|_1 = 1 \]
- \( \pi_i \) is PageRank of web page \( i \)

[Haveliwala & Kamvar 2003, Eldén 2003, Serra-Capizzano 2005]
How Google Ranks Web Pages

• Model:
  Internet → web graph → stochastic matrix $G$

• Computation:
  PageRank $\pi$ is eigenvector of $G$

  $\pi_i$ is PageRank of page $i$

• Display:
  If $\pi_i > \pi_k$ then
  page $i$ may* be displayed before page $k$

* depending on hypertext analysis
Facts

• The anatomy of a large-scale hypertextual web search engine [Brin & Page 1998]
• US patent for PageRank granted in 2001
• Google indexes 10’s of billions of web pages (1 billion = $10^9$)
• Google serves $\geq$ 200 million queries per day
• Each query processed by $\geq$ 1000 machines
• All search engines combined process more than 500 million queries per day

[Desikan, 26 October 2006]
Computation of PageRank

*The world’s largest matrix computation*
[Moler 2002]

- Eigenvector
- Matrix dimension is 10’s of billions
- The matrix changes often
  250,000 new domain names every day
- **Fortunately:** Matrix is sparse
Power Method

Want: $\pi$ such that $\pi^T G = \pi^T$

Power method:

Pick an initial guess $x^{(0)}$
Repeat

$[x^{(k+1)}]^T := [x^{(k)}]^T G$

until “termination criterion satisfied”

Each iteration is a matrix vector multiply
Matrix Vector Multiply

\[
x^T G = x^T \left[ \alpha S + (1 - \alpha) \mathbb{1} v^T \right]
\]

Cost: \# non-zero elements in \( S \)

A power method iteration is cheap
Error Reduction in 1 Iteration

\[ \pi^T G = \pi^T \quad G = \alpha S + (1 - \alpha) \parallel v \parallel^T \]

\[
\begin{align*}
[x^{(k+1)} - \pi]^T &= [x^{(k)}]^T G - \pi^T G \\
&= \alpha [x^{(k)} - \pi]^T S
\end{align*}
\]

Error:
\[
\parallel x^{(k+1)} - \pi \parallel_1 \leq \alpha \parallel x^{(k)} - \pi \parallel_1
\]
Error in Power Method

\[ \pi^T G = \pi^T \quad G = \alpha S + (1 - \alpha) \| v \|^T \]

Error after \( k \) iterations:

\[ \| x^{(k)} - \pi \|_1 \leq \alpha^k \underbrace{\| x^{(0)} - \pi \|_1}_{\leq 2} \]

[Bianchini, Gori & Scarselli 2003]

Error bound does not depend on matrix dimension
Advantages of Power Method

- Simple implementation (few decisions)
- Cheap iterations (sparse matvec)
- Minimal storage (a few vectors)
- Robust convergence behaviour
- Convergence rate independent of matrix dimension
- Numerically reliable and accurate (no subtractions, no overflow)

But: can be slow
PageRank Computation

• Power method
  Page, Brin, Motwani & Winograd 1999
  Bianchini, Gori & Scarselli 2003

• Acceleration of power method
  Kamvar, Haveliwala, Manning & Golub 2003
  Haveliwala, Kamvar, Klein, Manning & Golub 2003
  Brezinski & Redivo-Zaglia 2004, 2006
  Brezinski, Redivo-Zaglia & Serra-Capizzano 2005

• Aggregation/Disaggregation
  Ipsen & Kirkland 2006
PageRank Computation

- **Methods that adapt to web graph**
  - Broder, Lempel, Maghoul & Pedersen 2004
  - Kamvar, Haveliwala & Golub 2004
  - Haveliwala, Kamvar, Manning & Golub 2003
  - Lee, Golub & Zenios 2003
  - Lu, Zhang, Xi, Chen, Liu, Lyu & Ma 2004
  - Ipsen & Selee 2006

- **Krylov methods**
  - Golub & Greif 2004
  - Del Corso, Gullí, Romani 2006
PageRank Computation

- **Schwarz & asynchronous methods**
  Bru, Pedroche & Szyld 2005
  Kollias, Gallopooulos & Szyld 2006

- **Linear system solution**
  Arasu, Novak, Tomkins & Tomlin 2002
  Arasu, Novak & Tomkins 2003
  Bianchini, Gori & Scarselli 2003
  Gleich, Zukov & Berkin 2004
  Del Corso, Gullí & Romani 2004
  Langville & Meyer 2006
PageRank Computation

- Surveys of numerical methods:
  Langville & Meyer 2004
  Berkhin 2005
  Langville & Meyer 2006 (book)
Is the Ranking Correct?

\[ \pi^T = (0.23, 0.24, 0.26, 0.27) \]

- \[ x^T = (0.27, 0.26, 0.24, 0.23) \]
  \[ \|x - \pi\|_\infty = 0.04 \]
  Small error, but incorrect ranking

- \[ y^T = (0, 0.001, 0.002, 0.997) \]
  \[ \|y - \pi\|_\infty = 0.727 \]
  Large error, but correct ranking
What is Important?

Numerical value ↔ ordinal rank

ordinal rank:
position of an element in an ordered list

Very little research on ordinal ranking
Rank-stability, rank-similarity

[Lempel & Moran, 2005]
[Borodin, Roberts, Rosenthal & Tsaparas 2005]
Ordinal Ranking

Largest element gets Orank 1

\[ \pi^T = (0.23, 0.24, 0.26, 0.27) \]
\[ \text{Orank}(\pi_4) = 1, \text{Orank}(\pi_1) = 4 \]

\[ x^T = (0.27, 0.26, 0.24, 0.23) \]
\[ \text{Orank}(x_1) = 1 \neq \text{Orank}(\pi_1) = 4 \]

\[ y^T = (0, 0.001, 0.002, 0.997) \]
\[ \text{Orank}(y_4) = 1 = \text{Orank}(\pi_4) \]
Problems with Ordinal Ranking

When done with power method:

- **Popular termination criteria do not guarantee** correct ranking
- Additional iterations can **destroy** ranking
- Rank convergence depends on: \( \alpha \), \( v \), initial guess, matrix dimension, structure of web graph
- Even if **successive** iterates have the same ranking, their ranking may **not be correct**

[Wills & Ipsen 2007]
Ordinal Ranking Criterion

Given:

Approximation $x, x \geq 0$

Error bound $\beta \geq \|x - \pi\|_1$

Criterion: $x_i > x_j + \beta \implies \pi_i > \pi_j$

Why?

$$(x_i - \pi_i) - (x_j - \pi_j) \leq \|x - \pi\|_1 \leq \beta$$

$$x_i - (x_j + \beta) \leq \pi_i - \pi_j$$

$$0 < x_i - (x_j + \beta) \implies 0 < \pi_i - \pi_j$$

[Kirkland 2006]
Applicability of Criterion
Properties of Ranking Criterion

• Applies to any approximation, provided error bound is available
• Requires well-separated elements
• Tends to identify ranks of larger elements
• Determines partial ranking
• Top-k, bucket and exact ranking
• Easy to use with power method
Top-k Ranking

Given:

**Approximation** $x$ to PageRank $\pi$

**Permutation** $P$ so that $\tilde{x} = Px$ with

$\tilde{x}_1 \geq \ldots \geq \tilde{x}_n$

Write: $\tilde{\pi} = P\pi$

Suppose $\tilde{x}_k > \tilde{x}_{k+1} + \beta$
Top-k Ranking

\[ \tilde{x}_1 \leq \text{Orank}(\tilde{\pi}_1) \leq k \]

\[ \vdots \]

\[ \tilde{x}_k \leq \text{Orank}(\tilde{\pi}_k) \leq k \]

\[ \uparrow \beta \downarrow \]

\[ \tilde{x}_{k+1} \geq \text{Orank}(\tilde{\pi}_{k+1}) \geq k + 1 \]

\[ \vdots \]

\[ \tilde{x}_n \geq \text{Orank}(\tilde{\pi}_n) \geq k + 1 \]
Exact Ranking

Given:

Approximation $x$ to PageRank $\pi$
Permutation $P$ so that $\tilde{x} = Px$ with $\tilde{x}_1 \geq \ldots \geq \tilde{x}_n$

Write: $\tilde{\pi} = P\pi$

If $\tilde{x}_{k-1} > \tilde{x}_k + \beta$ and $\tilde{x}_k > \tilde{x}_{k+1} + \beta$ then

$\text{Orank}(\tilde{\pi}_k) = k$
Exact Ranking

\[ \tilde{x}_1 \quad \vdots \quad \tilde{x}_{k-1} \quad \tilde{x}_k \quad \vdash \beta \quad \text{Orank}(\tilde{\pi}_k) \geq k \]

\[ \tilde{x}_k \quad \vdash \beta \quad \text{Orank}(\tilde{\pi}_k) \leq k \]

\[ \tilde{x}_{k+1} \quad \vdots \quad \tilde{x}_n \]
Bucket Ranking

Given:

Approximation $x$ to PageRank $\pi$

Permutation $P$ so that $\tilde{x} = Px$ with $\tilde{x}_1 \geq \ldots \geq \tilde{x}_n$

Write: $\tilde{\pi} = P\pi$

Suppose $\tilde{x}_{k-i} > \tilde{x}_k + \beta$ and $\tilde{x}_k > \tilde{x}_{k+j} + \beta$

$\tilde{\pi}_k$ is in bucket of width $i + j - 1$
Bucket Ranking

\[
\tilde{x}_1 \quad \vdots \quad \tilde{x}_{k-i} \quad \uparrow \beta \quad \text{Orank}(\tilde{\pi}_k) \geq k - i + 1 \\
\tilde{x}_k \quad \downarrow \beta \quad \text{Orank}(\tilde{\pi}_k) \leq k + j - 1 \\
\vdots \quad \vdots \\
\tilde{x}_{k+j} \quad \vdots \\
\tilde{x}_n
\]
Experiments

<table>
<thead>
<tr>
<th>$n$</th>
<th># buckets</th>
<th>1. bucket</th>
<th>last bucket</th>
</tr>
</thead>
<tbody>
<tr>
<td>9,914</td>
<td>4,307</td>
<td>1</td>
<td>7%</td>
</tr>
<tr>
<td>3,148,440</td>
<td>34,911</td>
<td>1</td>
<td>95%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$n$</th>
<th>exact rank</th>
<th>exact top 100</th>
<th>lowest rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>9,914</td>
<td>32%</td>
<td>79</td>
<td>9,215</td>
</tr>
<tr>
<td>3,148,440</td>
<td>0.76%</td>
<td>100</td>
<td>151,794</td>
</tr>
</tbody>
</table>
Buckets for Small Matrix
Power Method Ranking

Simple error bound:

\[ \| x^{(k)} - \pi \|_1 \leq \frac{2\alpha^k}{\beta} \]

Simple ranking criterion:

If \( x_i^{(k)} > x_j^{(k)} + 2\alpha^k \) then \( \pi_i > \pi_j \)

But: \( 2\alpha^k \) is too pessimistic (not tight enough)
Power Method Ranking

Tighter error bound:

\[ \| x^{(k)} - \pi \|_1 \leq \frac{\alpha}{1 - \alpha} \left( \| x^{(k)} - x^{(k-1)} \|_1 + \beta \right) \]

More effective ranking criterion:

If \( x_i^{(k)} > x_j^{(k)} + \beta \) then \( \pi_i > \pi_j \)
Floating Point Ranking

If $x_{i}^{(k)} > x_{j}^{(k)} + \beta$ then $\pi_{i} > \pi_{j}$

\[
\beta = \frac{\alpha}{1 - \alpha} \|x^{(k)} - x^{(k-1)}\|_1 + e
\]

$e$ is round off error from single matvec

IEEE double precision floating point arithmetic:

\[
e \approx cm10^{-16}
\]

$m \approx \text{max \# links into any web page}$
Expensive Implementation

To prevent accumulation of round off

- Explicit normalization of iterates
  \[ \mathbf{x}^{(k+1)} = \frac{\mathbf{x}^{(k+1)}}{\|\mathbf{x}^{(k+1)}\|_1} \]

- Compute norms, inner products, matvecs with compensated summation

- Limited by round off error from single matvec

- Analysis for matrix dimensions \( n < 10^{14} \)
  in IEEE arithmetic \( (\epsilon \approx 10^{-16}) \)
Difficulties for XLARGE Problems

• Catastrophic cancellation when computing

\[ \beta = \frac{\alpha}{1 - \alpha} \| x^{(k)} - x^{(k-1)} \|_1 + e \]

• Bound \( \beta \) dominated by round off \( e \)

• Compensated summation insufficient to reduce higher order round off \( \mathcal{O}(n\epsilon^2) \)

• Doubly compensated summation too expensive: \( \mathcal{O}(n \log n) \) flops
Possible Remedies

- Lump dangling nodes  [Ipsen & Selee 2006]
  Web pages w/o outlinks:
  pdf & image files, protected pages, web frontier
  Up to 50%-80% of all web pages
- Remove unreferenced web pages
- Use faster converging method
  Then 1 power method iteration for ranking
- Relative ranking criteria?
Google orders web pages according to: PageRank and hypertext analysis

PageRank = left eigenvector of $G$

$$G = \alpha S + (1 - \alpha) 1 \, 1^T$$

Power method: simple, robust, accurate

Convergence rate depends on $\alpha$ but not on matrix dimension

Criterion for ordinal ranking

Round off serious for XLarge problems
User-Friendly Resources

- Rebecca Wills:  
  *Google’s PageRank: The Math Behind the Search Engine*  
  Mathematical Intelligencer, 2006

- Amy Langville & Carl Meyer:  
  *Google’s PageRank and Beyond The Science of Search Engine Rankings*  
  Princeton University Press, 2006

- Amy Langville & Carl Meyer:  
  Broadcast of On-Air Interview, November 2006  
  Carl Meyer’s web page