An Ensembling of Pagerank Accelerators

An inner-outer solver combined with robust matrix subsampling

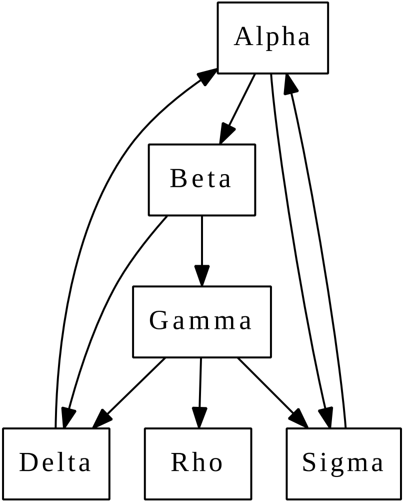
Brock Hargreaves

CPSC 517
Sparse Matrix Computation

December 10th, 2013

Illustration: Miniature web example

Goal: Based on their interactions, rank these web pages



Miniature web example: Adjacency Matrix

$$
\begin{array}{c c}
 & \begin{array}{cccccc} \mathit{Alpha} & \mathit{Beta} & \mathit{Gamma} & \mathit{Delta} & \mathit{Rho} & \mathit{Sigma} \end{array} \\
\begin{array}{c} \mathit{Alpha} \\ \mathit{Beta} \\ \mathit{Gamma} \\ \mathit{Delta} \\ \mathit{Rho} \\ \mathit{Sigma} \end{array} &
\begin{pmatrix}
0 & 0 & 0 & 1 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
1 & 0 & 1 & 0 & 0 & 0
\end{pmatrix}
\end{array}
$$

Let n_j be the out degree of the j'th column, the $j'th$ column sum.

The PageRank Formulation

Raw Pagerank of a page

$$x_i = \sum_{j \to i} \frac{x_j}{n_j} \text{ ; } n_j \text{ is the out-degree of page j}$$

Then the problem is to find a vector x that satisfies $x = \bar{P}x$ where

$$\bar{P}_{j,i} = \begin{cases} \frac{1}{n_i} & : i \to j \\ 0 & : i \not\to j \end{cases}$$

Handling dangling nodes

Idea: Perturb pages with no outlinks

$$P = \bar{P} + ud^T \ ; \ d_i = \begin{cases} 1 & \text{if } n_i = 0 \\ 0 & otherwise \end{cases}$$

Further allow users to teleport anywhere in the graph (and gives us a nice stochastic matrix):

$$A = \alpha P + (1 - \alpha)\, ve^T$$

Then we wish to find x such that

$$Ax = x$$

and apply solver.

$A =$

$$\begin{array}{c|cccccc} & Alpha & Beta & Gamma & Delta & Rho & Sigma \\ \hline Alpha & .025 & .025 & .025 & .875 & .166 & .875 \\ Beta & .450 & .025 & .025 & .025 & .166 & .025 \\ Gamma & .025 & .450 & .025 & .025 & .166 & .025 \\ Delta & .025 & .450 & .308 & .025 & .166 & .025 \\ Rho & .025 & .025 & .308 & .025 & .166 & .025 \\ Sigma & .450 & .025 & .308 & .025 & .166 & .025 \end{array}$$

Power method on $Ax = x$

$$x = [.321, .17, .106, .136, .064, .200]^T$$

An Inner-Outer Iteration for Computing PageRank

D. Gleich ,A. Gray ,C. Greif and T. Lau 2010 [1]

Reformulate $Ax = x$ as:

$$(I - \alpha P)x = (1 - \alpha)v$$

Note:

$$(I - \beta P)x = (1 - \beta)v$$

is easier to solve for $\beta < \alpha$ (Check eigenvalues)

Inspired by this β business, consider the stationary iteration:

$$(I - \beta P)x_{k+1} = (\alpha - \beta)Px_k + (1 - \alpha)v$$

Idea: Fix the right hand side:

$$f = (\alpha - \beta)Px_k + (1 - \alpha)v$$

and solve(approximately) an easier subproblem:

$$(I - \beta P)y = f$$

via the inner iteration $y_{j+1} = \beta P y_j + f$. Then set $x_{k+1} = y$, update the right hand side and repeat.

Fast Pagerank approximation by adaptive sampling

W. Liu, G. Li, J. Cheng [2] 2013

Idea: Sample the transition matrix P, while preserving the edge distribution of the input graph.

Algorithm 1 Non uniform sampling(A,α,θ) (A arbitrary)

1: Initialize: An empty priority queue, Q, $Z = 0$, $s = \frac{N}{\alpha^2}$
2: **for** each $A_{i,j}$ **do**
3: Set $Z \leftarrow Z + A_{i,j}^2$
4: draw $r_{i,j}$ uniformly and independently from $[0, 1]$
5: insert $A_{i,j}$ in Q with key $k_{i,j} = max\{\frac{sA_{i,j}^2}{r_{i,j}}, \frac{sA_{i,j}^2}{\theta^2 r_{i,j}^2}\}$
6: remove all elements with key smaller than Z
7: **end for**
8: Set $\tilde{A}$ as the contents of Q

Sampling strategies

1. Direct sampling: Sample the transition matrix and then apply inner-outer iterations

2. Adaptive sampling: At each outer iteration, adjust sampling rate and add more samples from transition matrix

Example: eu-2005
n= 862,664 , nnz = 19,235,140

Applying only the sampling operator:
Naive first try MATLAB implementation : $\approx$ 21 days
Mex compiled C++ priority queue : $\approx$ 20 minutes

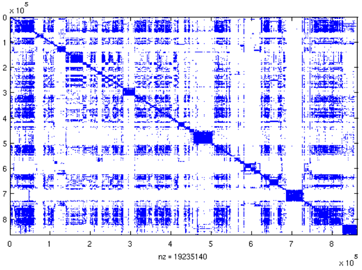


Figure: nnz = 19,235,140

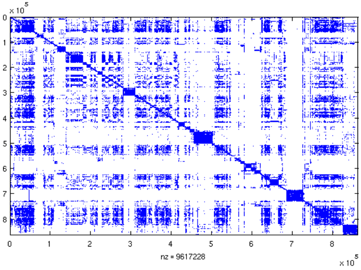


Figure: nnz = 9,617,228

In progress

- Quantize gains made (speed vs accuracy) using a combination of the inner-outer solver and the two sampling techniques described.
- Write an (even more) efficient implementation of sampling algorithm
- Investigation parameter choices for sampling (currently using author suggestions)

Prediction: Significant reduction in computation with comparable accuracy

References

David F. Gleich, Andrew P. Gray, Chen Greif, and Tracy Lau.
An inner-outer iteration for PageRank.
SIAM Journal of Scientific Computing, 32(1):349–371, February 2010.

Wenting Liu, Guangxia Li, and James Cheng.
Fast pagerank approximation by adaptive sampling.
Knowledge and Information Systems, pages 1–20, 2013.