Page Rank
Las last time ...

- Complexity theory for MapReduce
- PageRank
Today ...

- PageRank
  - Topic specific
  - Combatting spam
- Assignment 2
  - Matrix multiplication
  - Page Rank
Efficient Computation of PageRank

- Transition Matrix $M$ is very sparse
- Store locations of non-zero entries
- In general for sparse matrices
  - $(i, j, M_{ij}) \rightarrow 4+4+8$ bytes
- Further compression possible for transition matrix
  - Store degree of column plus indices
  - Number of links on a page plus the indices of those pages
Topic Sensitive PageRank

- Weight certain pages more because of their topic
- Allows personalization of results to users
  - Ideally a separate page rank vector for each user
  - Not scalable
- Create one vector for each of a small set of topics
  - Basis vectors
  - Determine weights for each individual user
    - size $\rightarrow$ number of basis vectors
Biased Random Walks

- Identify certain pages that represent a given topic
- (re) introduce random surfers to only topic specific pages
- Let $S$ be the set of integers consisting of the indices of topic-specific pages, and $e_S$ be a vector that is 1 in $S$ and 0 elsewhere
- Topic sensitive PageRank

$$v' = \beta M v + \frac{(1-\beta)e_S}{|S|}$$
Decide on the topics for which we shall create specialized PageRank vectors
- Manually
- From Data

Pick the set $S$ for each of these topics, and use that set to compute the topic-sensitive PageRank vector for that topic

Determine which topics are of most interest to a particular user/query

Use the PageRank vectors for those topics in ordering the results
Link Spam

- Techniques for artificially increasing the PageRank of a page
- Spam Farm

Diagram:
- Inaccessible Pages
- Accessible Pages
- Own Pages
Analysis of a Spam farm

- $\beta \rightarrow$ taxation parameter
- $n \rightarrow$ total number of webpages
- Target $t$ with $m$ supporting pages
- Let $x$ be the amount of PageRank contributed by accessible pages
- Let us compute $y$, the PageRank of $t$
Analysis of a Spam farm

- PageRank of each supporting page

\[ \frac{\beta y}{m} + \frac{1 - \beta}{n} \]

- PageRank of target

\[ y = x + \beta m \left( \frac{\beta y}{m} + \frac{1 - \beta}{n} \right) + \frac{1 - \beta}{n} \]
Analysis of a Spam farm

- PageRank of each supporting page

\[
\frac{\beta y}{m} + \frac{1 - \beta}{n}
\]

- PageRank of target

\[
y = x + \beta^2 y + \beta (1 - \beta) \frac{m}{n}
\]
Analysis of a Spam farm

- PageRank of each supporting page
  \[
  \frac{\beta y}{m} + \frac{1 - \beta}{n}
  \]

- PageRank of target
  \[
  y = \frac{x}{1 - \beta^2} + \frac{\beta}{1 + \beta} \frac{m}{n}
  \]

- If $\beta = 0.85$
Analysis of a Spam farm

- PageRank of each supporting page

\[
\frac{\beta y}{m} + \frac{1 - \beta}{n}
\]

- PageRank of target

\[
y = 3.6 x + 0.46 \frac{m}{n}
\]
Combating Link Spam

- **TrustRank**: variation of topic-sensitive PageRank
- **Spam mass**: calculation that identifies spam farms
Trust Rank

- topic-sensitive PageRank, where the topic is a set of pages believed to be trustworthy
- Manually select trustworthy pages
- Avoid trustworthy sites where anyone can create links
  - Many websites prevent users from entering URLs in comments
- Domains where membership is controlled
  - .edu .gov etc ...
Spam Mass

- Measure the fraction of the pagerank that comes from spam
- Compute the ordinary pagerank ($r$) and trustrank ($t$) of a page
  
  - Spam mass $= \frac{r-t}{r}$
  
- Negative or small positive spam mass $\rightarrow$ not spam
- Closer to 1 $\rightarrow$ spam