Link Analysis: Page Rank

CMSC 498J: Social Media Computing

Department of Computer Science
University of Maryland
Spring 2016

Hadi Amiri
hadi@umd.edu
Announcement

• HW3 is out
  ▫ Due date: 04/14/2016 11:00AM
Lecture Topics

• Page Rank
  ▫ Analysis of PageRank
• Random Walks
  ▫ Analysis of Random Walk
• Applications
HITS- Recap

**Algorithm**
1. Set hub and authority scores to 1.
2. Choose a number of steps \( k \).
3. Perform \( k \) hub-authority updates:
   1. Apply AUR to the current set of scores.
   2. Then apply HUR to the resulting scores.
4. Normalize authority and hub scores

\[
a^{(k)} = (M^T M)^{k-1} M^T h^{(0)}
\]

\[
h^{(k)} = (M M^T)^k h^{(0)}.
\]

- **AUR:** Authority score of page \( p \):
  - sum of the hub scores of pages that point to \( p \).

- **HUR:** Hub Score of page \( p \):
  - sum of the authority scores of pages that \( p \) points to.
Page Rank

• A page is important if it is linked / endorsed by other important pages (iterative process)
  ▫ dominant mode of endorsement among
    • academic or governmental pages,
    • bloggers,
    • scientific literature, or even
    • personal pages!

• Each node has one score, PageRank score!

• Votes / Endorsements pass directly from one page to another (across outgoing links)!

Named after Google co-founder Larry Page.
The weight of a node's endorsement:
- Its current PageRank score.
- Nodes that are currently viewed as more important make stronger endorsements.

Source: http://en.wikipedia.org/wiki/PageRank
Page Rank- Cnt.

- **PageRank Update Rule:**
  1. Each page divides its current PageRank equally across its out-going links
     - passes *equal shares* to the pages it points to.
     - If a page has **no out-going** links, it passes all its current PageRank to itself.
  2. Each page updates its new PageRank to be the sum of the shares it receives.
Page Rank- Cnt.

**Algorithm**

1. Set initial PageRank of all nodes to $1/n$.
2. Perform $k$ updates to the PageRank values:
   1. Apply PageRank Update Rule
Page Rank - Cnt.

• **PageRank intuitive view:**
  - "fluid" that circulates through the network
  - passing across edges, and
  - pooling at the nodes that are the most important!
Page Rank- Cnt.

• Sum of PageRank values in the network?
  ▫ **Remains constant** as PageRank is never created nor destroyed!
    • just moved around from one node to another.
  ▫ Each page takes its PageRank, divides it up, and passes it along links

• We don’t need to normalize values anymore!
  ▫ In contrast to HITS!
Page Rank- Cnt.

- $n=8$
- Initially $p_i = 1/8$ for all nodes

A acquires a lot of PageRank
B and C benefit in the next step.
B and C are more important than D, E, F, G, and H
Page Rank- Cnt.

- n=8
- Initially $p_i = 1/8$ for all nodes

**Equilibrium Values of PageRank**

When we reach the limiting PageRank values:
1. PageRank values sum to 1, and
2. If we apply the PageRank Update Rule, the values at every node remains the same
   - values regenerate themselves exactly when they are updated.
• Issue with page rank algorithm?
  ▫ “Wrong” nodes can end up with all the PageRank in the network!

F and G point to each other!

PageRank that flows from C to F and G can never circulate back into the rest of the network

For large $k$, PageRank values converge to $1/2$ for each of F and G, and 0 for all other nodes.

The links out of C function as a kind of “slow leak” that eventually causes all the PageRank to end up at F and G.
• Issue with page rank algorithm?
  ▫ “Wrong” nodes can end up with all the PageRank in the network!

As long as there are small sets of nodes that can be reached from the rest of the graph, but have no paths back, then PageRank will build up there!
Page Rank - Cnt.

\[ M = \begin{bmatrix} 1/2 & 1/2 & 0 \\ 1/2 & 0 & 0 \\ 0 & 1/2 & 1 \end{bmatrix} \]

\[
\begin{bmatrix} yahoo \\ Amazon \\ Microsoft \end{bmatrix} = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix}
\]
Page Rank- Cnt.

- **Link farms**
  - Pages heavily linked to each other
  - Created by automated programs
  - Fooling search engines
Page Rank- Cnt.

- **Scaled PageRank Update Rule:**
  - Pick a scaling factor $0 < s < 1$
  - Apply the PageRank Update Rule as before.
  - Then scale down all PageRank values by a factor of $s$.
    - The total PageRank in the network?
      - shrinks from 1 to $s$.
  - Divide residual $(1-s)$ PageRank equally over nodes
    - giving $(1-s)/n$ to each node.

**Common values for $s$ are in the range of 0.8 to 0.9!**

*The above rule follows from the “fluid” intuition for PageRank*
- Why all the water on earth doesn’t inexorably run downhill and reside exclusively at the lowest points?
- There’s a counter-balancing process at work:
- Water also evaporates and gets rained back down at higher elevations!
Page Rank- Cnt.

• Repeated application of the Scaled PageRank Update Rule *converges* to a set of limiting PageRank values as $k$ goes to infinity!

• Do the resultant values depend on the choice of $s$?
  ▫ Yes, different update rules for different values of $s$. 
Spectral Analysis of PageRank

PageRank Update Rule

• Let $N_{ij}$ to be the portion of $i$’s PageRank that circulate to $j$ in one update step:
  - $N_{ij} = 0$ if $i$ doesn’t link to $j$
  - $N_{ij} = 1/l_i$ Otherwise
    • Where $l_i$ is out-degree of $i$
  - If $i$ has no outgoing links, then we define $N_{ii} = 1$
    • A node with no outgoing links passes all its PageRank to itself
Spectral Analysis of PageRank - Cnt.

Figure 14.13: The flow of PageRank under the Basic PageRank Update Rule can be represented using a matrix $N$ derived from the adjacency matrix $M$: the entry $N_{ij}$ specifies the portion of $i$’s PageRank that should be passed to $j$ in one update step.
Spectral Analysis of PageRank - Cnt.

- \( r \in n \times 1 \)
  - vector representing PageRanks of all \( n \) nodes
- PageRank Update Rule:

\[
    r_i \leftarrow N_{1i}r_1 + N_{2i}r_2 + \cdots + N_{ni}r_n.
\]

\[
    r \leftarrow N^T r.
\]
Scaled PageRank Update Rule

- Let \( \tilde{N}_{ij} \) to be the portion of \( i \)'s PageRank that circulate to \( j \) in one update step:
  - The updated PageRank is scaled down by a factor of \( s \), and the residual \((1 - s)\) units are divided equally over all nodes

\[
\begin{align*}
  r_i &\leftarrow \tilde{N}_{1i}r_1 + \tilde{N}_{2i}r_2 + \cdots + \tilde{N}_{ni}r_n. \\
  r &\leftarrow \tilde{N}^T r.
\end{align*}
\]
Spectral Analysis of PageRank - Cnt.

• Start from an initial PageRank vector $r^{<0>}$ and produce a sequence of vectors $r^{<1>}$, $r^{<2>}$, ... each is obtained from the preceding one via multiplication by $\tilde{N}^T$.
  ▫ Unwind this process:

$$r^{<k>} = (\tilde{N}^T)^k r^{<0>}.$$  

• The Scaled PageRank Update Rule converges to a limiting vector $r^{<*>}$!

$$\tilde{N}^T r^{<*>} = r^{<*>}$$

▫ $r^{<*>}$ should be an eigenvector of $\tilde{N}^T$ with eigenvalue of 1.

• Reed page 376 for the remaining of this proof.
Random Walks

• **Random walk** on a network:
  ▫ Start by choosing a page at random
    • Pick each page with equal probability.
  ▫ Follow links for a sequence of \( k \) steps:
    • In each step, pick a random out-going link from the current page, and follow it to where it leads.
      • If there is no out-going links, stay at current node.

• An equivalent formulation of PageRank that leads to exactly the same definition!
**Claim**: The probability of being at a page $p$ after $k$ steps of random walk is precisely the PageRank of $p$ after $k$ applications of the PageRank Update Rule.

**Claim**: The PageRank of a page $p$ is the probability that a random walk across links will end up at $p$, as we run the walk for larger and larger $k$. 
Random Walks- Cnt.

- Interpretation of the leakage issue in terms of Random Walks:
  - As the walk runs for more and more steps:
    - The probability of reaching F or G is converging to 1;
    - Once it reaches F or G, it is stuck there forever.
    - The probability of being at F (G) is converging to 1/2
    - The probabilities are converging to 0 for all other nodes.
Analysis of Random Walk

- Let $b_1, b_2, \ldots, b_n$ denote the probabilities of being at nodes 1, 2, \ldots, $n$ respectively in a given step.
- What is the probability of going to node $i$ in the next step?
  - For each $j$ that links to $i$, if the walk is at node $j$, there is a $1/l_j$ chance that it moves from $j$ to $i$ in next step
    - $l_j$ is the number of links out of $j$.
  - The walk has to actually be at node $j$ for this to happen, so node $j$ contributes $b_j(1/l_j) = b_j/l_j$ to the probability of being at $i$ in the next step.
  - Sum $b_j/l_j$ over all nodes $j$ that link to $i$ gives the probability of being at $i$ in the next step.
Analysis of Random Walk - Cnt.

• What is the probability of going to node $i$ in the next step?

\[ b_i = \sum_{j=1}^{n} N_{ji}b_j \]

$1/l_j$ chance that it moves from $j$ to $i$

$j$ contributes $b_j/l_j$ to the probability of being at $i$ in the next step

Sum $b_j/l_j$ over all nodes $j$ that link to $i$
Analysis of Random Walk- Cnt.

• Overall probability that the walk is at \( i \) in the next step is the sum of \( \frac{b_j}{l_j} \) over all nodes that link to \( i \)

\[
b_i \leftarrow N_{1i}b_1 + N_{2i}b_2 + \cdots + N_{ni}b_n.
\]

\[
b \leftarrow N^Tb.
\]

• Scaled version

\[
b_i \leftarrow \tilde{N}_{1i}b_1 + \tilde{N}_{2i}b_2 + \cdots + \tilde{N}_{ni}b_n.
\]

\[
b \leftarrow \tilde{N}^Tb.
\]

The probability of being at a page \( x \) after \( k \) steps of the scaled random walk (next slide) is the PageRank of \( x \) after \( k \) applications of the Scaled PageRank Update Rule
Random Walks- Cnt.

- Random Walks in terms of Scaled PageRank Update Rule:
  - With probability $s$, the walker follows a random edge as before;
  - With probability $(1 - s)$, the walker jumps to a random node anywhere in the network
    - choosing each node with equal probability.
Applications

- **Impact Factor of Scientific Journals**

  **Impact Factor** for a scientific journal: The average number of citations received by papers published in the given journal over the past two years.

  In-links indicate **collective attention** that the scientific community pays to papers published in the journal.
Applications- Cnt.

• **Fighting Lung Cancer Using PageRank**

metastatic lung cancer does not progress in a single direction from primary tumor site to distant locations, **which has been the traditional medical view**. Instead, cancer cell movement around the body likely occurs in more than one direction at a time.

**certain organs tend to spread cancer cells** more aggressively, while others tend to act as sponges for cancer cells!

Applications- Cnt.

• Applying PageRank to the Molecular Universe

Because the PageRank of a molecule affects how it will act in a chemical reaction — and water is involved in almost every biological process. By understanding how a network of trillions of molecules interact, scientists can produce much more accurate models of chemical reactions.

Video: https://www.youtube.com/watch?v=-tLQ_pDZY7Y
• Google trick tracks extinctions

Google's algorithm for ranking web pages can be adapted to **determine which species are critical for sustaining ecosystems.**

Modification of food webs from ecological considerations to satisfy the two constraints required for application of the algorithm.

Source: Stefano et. al., Googling food webs: can an eigenvector measure species’ importance for coextinctions? Plos comput biol 09
Questions?
Reading

- Ch.14 Link Analysis and Web search [NCM]
- Ch.05 Link Analysis [MMD]