PageRank Analysis, Implementation & Optimization

Chaoran Chen
Outline

- Motivation
- Review
- Implementation
- Visualization
- Experiment results on large datasets
- Conclusion
- References
Motivation

- Need for PageRank:
  The Search engines store billions of web pages which overall contain trillions of web URL links. So, there is a need for an algorithm that gives the most relevant pages specific to a query.

- Need for Distributed Environment:
  Trillions of links implies huge data storage required. So Map-Reduce and Distributed Storage is needed.

- How to improve?
PageRank $x_p$ of $p$ is computed by taking into account the set of pages $pa[p]$ pointing to $p$.

$$x_p = d \sum_{q \in pa[p]} \frac{x_q}{h_q} + (1 - d).$$

Here $h_q$ is the outdegree of $q$, that is the number of hyperlinks outgoing from $q$. Let $d$ be a factor used for normalization so that the total rank of all web pages is constant.

When stacking all the $x_p$ into vector $x$, we get

$$x = dWx + (1 - d)I.$$

Here $W = \{w_{i,j}\}$ is transition matrix. $w_{i,j} = 1/h_j$ if there is a hyperlink from $j$ to $i$ and $w_{i,j} = 0$ otherwise.
Review

- **Stochastic Interpretation:**
\[ x(t + 1) = dWx(t) + (1 - d)I_N \]

PageRank dynamic system (random walk), stable after \( n \) iterations, proved by Markov Chain Theory.

- **Dumping factor** \( d \):
  - If \( d = 0 \), all the PageRanks equals 1.
  - If \( d = 1 \), many pages would have a zero PageRank.

- **Dangling pages:** pages w/o hyperlinks

  Handling: introducing a dummy node or removing dangling pages.
Communities and Energy Balance

A community could be a set of pages on a given topic, the researchers’ home pages or a Website; the corresponding energy is a measure of its authority.

\[ E_i = |I| + E_{iin} - E_{iout} - E_{idp} \]

- \( |I| \): # of pages, “default energy”
- \( E_{iin} \): Page Rank inside the community, communities with many references have a high authority
- \( E_{iout} \): Page Rank outside the community, having hyperlinks outside the community leads to decrease energy
- \( E_{idp} \): the presence of pages without hyperlinks yields a loss of energy
Energy Calculation, determined by d, W and x

\[ \text{Energy Loss} = E_I^{\text{out}} + E_I^{\text{dp}} \]
Review

- Page Promotion:
  Splitting into multiple pages.
  The same content divided into many small pages yields a higher score than the same content into a single large page. Increase the PageRanks.
Implementation

- Convert each URL into a unique integer.
- Store each hyperlink in a database using the integer IDs to identify pages.
- PR(n) = Transition Matrix * PR(n-1)

```
1 http://www1.hollins.edu/
2 http://www.hollins.edu/
3 http://www1.hollins.edu/Docs/CompTech/Network/webmail_faq.htm
4 http://www1.hollins.edu/Docs/Forms/GetForms.htm
5 http://www1.hollins.edu/Docs/misc/travel.htm
6 http://www1.hollins.edu/Docs/GVCalendar/gvmain.htm
7 http://www1.hollins.edu/docs/events/events.htm
```
Implementation

- Dataset:
  Wiki-Vote, Nodes: 7115, Edges: 103689
  soc-Epinions, Nodes: 75879, Edges: 508837

- Input format:

```
# FromNodeId ToNodeId
30 1412
30 3352
30 5254
4 30
4 38
```

<table>
<thead>
<tr>
<th>W</th>
<th>30</th>
<th>38</th>
<th>1412</th>
<th>3352</th>
<th>5254</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>1/2</td>
<td>1/2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>30</td>
<td>0</td>
<td>0</td>
<td>1/3</td>
<td>1/3</td>
<td>1/3</td>
</tr>
</tbody>
</table>
How to calculate PR1?

PR1 = Transition Matrix * PR0

<table>
<thead>
<tr>
<th>To\From</th>
<th>WA</th>
<th>WB</th>
<th>WC</th>
<th>WD</th>
</tr>
</thead>
<tbody>
<tr>
<td>WA</td>
<td>0</td>
<td>1/2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>WB</td>
<td>1/3</td>
<td>0</td>
<td>0</td>
<td>1/2</td>
</tr>
<tr>
<td>WC</td>
<td>1/3</td>
<td>0</td>
<td>0</td>
<td>1/2</td>
</tr>
<tr>
<td>WD</td>
<td>1/3</td>
<td>1/2</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

transition.txt (dataset)

PR0

<table>
<thead>
<tr>
<th>A</th>
<th>1/4</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>1/4</td>
</tr>
<tr>
<td>C</td>
<td>1/4</td>
</tr>
<tr>
<td>D</td>
<td>1/4</td>
</tr>
</tbody>
</table>

PR0.txt

initial vector

PR1

<table>
<thead>
<tr>
<th>A</th>
<th>9/24</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>5/24</td>
</tr>
<tr>
<td>C</td>
<td>5/24</td>
</tr>
<tr>
<td>D</td>
<td>5/24</td>
</tr>
</tbody>
</table>
Implementation

- Apache Hadoop MapReduce
- Preprocessing:

  Client read input from the file that contains the dataset and build the transition matrix $W$.

  Client read input from initial PageRank file PR0.txt to build the PR vector $x$. 

Implementation

- Maper: leverage the job onto multiple machines.
- Reducer: compute the ranking value on different machines and combine the results into a single final result.
- The input and output of Mapper and Reducer are <Key, Value> pairs, which can be stored in HBase.
Visualization

Experiment Results

- Stability
- The result will converge after certain number of iterations.
Experiment Results

- Different dumping factor
- When $d$ approaches to 1(0.8), the loss can be an important percentage of the available energy
- Better performance when $d$ is between 0.2 and 0.6
Experiment Results

- Optimization: Page Promotion
- Tried Splitting the content of pages to promote the performance.
In order to maximize the efficiency of the search engines, we need to reduce the energy loss of the system as much as possible.

PageRank is strongly affected by the choice of the dumping factor $d$. If $d$ approaches 1, the loss can be an important percentage of the available energy.

Page Promotion is an effective way to promote the overall PageRank of the entire page community.

Future work: Trying different Page Promotion strategies in more large datasets and compare the performance. Build a real-world web ranking application for ranking different networks.
References

- Page, L., Brin, S., Motwani, R., and Winograd, T.
  The PageRank citation ranking: Bringing order to the web.
- MONICA BIANCHINI, MARCO GORI, and FRANCO SCARSELLI
  Inside PageRank.
  University of Siena, 2000.
- Surajkumar Harikumar and Manikandan Srinivasan
  Google’s PageRank Algorithm : An Analysis, Implementation and Relevance today.
Thanks!