Google's Mechanism for Ranking Web-Pages

"The World's Largest Matrix Computation"

Aim and Problem Statement

- We aim at understanding the mathematics behind the most successful search engine i.e. Google by using a simplified version of the Random Surfer Algorithm.
- Did you ever think that how do the most relevant web pages appear generally in the first page of a Google search? How does Google find exactly the pages which can provide most suitable result for your search?
- It happens due to Google's **PageRank Algorithm**.
- PageRank Algorithm is a ranking system originally devised by ***Larry Page*** and ***Sergey Brin*** – the founders of Google.

The concept of PageRank

- PageRank* is a function that assigns a real number to each page in the Web. The intent is that the higher the PageRank of a page, the more "important" it is.
- The underlying assumption is that the more "important" web-pages will receive more links from other web-pages. Links work as votes, raising the credibility and showcasing the web-site as high-quality.
- There are several algorithms for assignment of PageRank, and in fact variations on the basic idea can alter the relative PageRank of any two pages.

*The term PageRank comes from Larry Page, the inventor of the idea.

The Random Surfer Algorithm

- The PageRank theory holds that an imaginary surfer who is randomly clicking on links will eventually stop clicking. The PageRank value of a page reflects the **chance** that the random surfer will land on that page by clicking on a link. The probability, at any step, that the person will continue is a damping factor *d*.
- For a web of pages A, B, C, D,… the PageRank of A is given by:

$$PR(A) = \frac{1-d}{N} + d\left(\frac{PR(B)}{L(B)} + \frac{PR(C)}{L(C)} + \frac{PR(D)}{L(D)} + \cdots\right).$$

- $PR(.)$ denotes PageRank and $L(.)$ denotes number of outbound links.
- Generally the damping factor will be set around 0.85.
- Links from a page to itself, or multiple outbound links from one single page to another single page, are ignored.

Mathematical analysis

- Let us consider a web consisting of n-pages and let us denote the importance (PageRank) of i-th page by x_i where $1 \leq i \leq n$. In the simplified random surfer model, with dampening factor $(d) = 1$, the importance of i-th page is calculated by the equation:

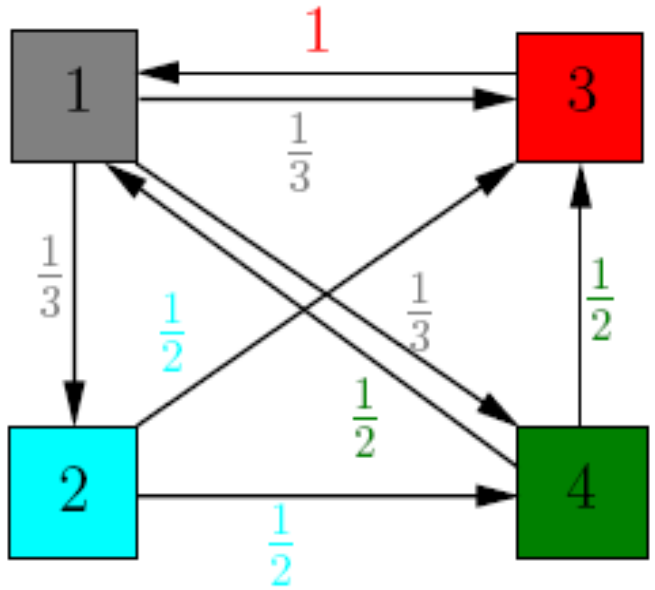
$$x_i = \sum_j \frac{x_j}{n_j} \qquad (1)$$

when j-th page contains n_j outbound links, one of which links to the i-th page.

- Dealing with an example will be more suitable here. So let's go for it.

Example

- Consider a tiny web consisting of four pages only as depicted in the below diagram which shows the "links" from one web page to another web page.



- A real web may consist of millions of web-pages.

- Now by the formula (Eq. (1)) we have:

$$
\begin{aligned}
x_1 &= 0\,x_1 + 0\,x_2 + 1\,x_3 + \frac{1}{2}x_4, \\
x_2 &= \frac{1}{3}\,x_1 + 0\,x_2 + 0\,x_3 + 0\,x_4, \\
x_3 &= \frac{1}{3}\,x_1 + \frac{1}{2}\,x_2 + 0\,x_3 + \frac{1}{2}x_4, \\
x_4 &= \frac{1}{3}\,x_1 + \frac{1}{2}\,x_2 + 0\,x_3 + 0\,x_4.
\end{aligned}
\qquad (2)
$$

- The coefficients in the first equation are obtained as follows:
  - Link from Page #1 to itself does not count. Hence coefficient of x_1 is zero.
  - There is no link from Page #2 to Page #1. Hence coefficient of x_2 is zero.
  - Page #3 has only one outbound link that too to Page #1. Hence coefficient of x_2 is 1.
  - Page #4 has two outbound links, one of them to Page #1. Hence coefficient of x_4 is $\frac{1}{2}$.

- The system of equations (Eq. (2)) can be written as $\boldsymbol{X = AX}$.
- Let's take a pause here and think…

What does the equation $\boldsymbol{AX = X}$ tell about $\boldsymbol{X}$???

- $\boldsymbol{X}$ is the eigenvector corresponding to the eigenvalue 1.
- In the example discuss in the previous slide we have

$$A = \begin{bmatrix} 0 & 0 & 1 & 1/2 \\ 1/3 & 0 & 0 & 0 \\ 1/3 & 1/2 & 0 & 1/2 \\ 1/3 & 1/2 & 0 & 0 \end{bmatrix}.$$

- **The eigenvector, corresponding to the eigenvalue 1, provides non-trivial solution to the system and is called importance vector or PageRank vector.**

MatLab Code:

- Enter the matrix A:

```
>> A = [0 0 1 1/2; 1/3 0 0 1/2; 1/3 1/2 0 0;1/3 1/2  0 0]
```

- Find the eigenvalues and eigenvectors of the matrix 'A' using "eigs" command which yields eigenvalues in descending order.

```
>> [V, D] = eigs(A);
```

- By the property of transition matrices, the maximum eigenvalue of the transition matrix must be ONE. Hence the eigenvector corresponding to the eigenvalue 1 will be extracted by:

```
>> u = V[:,1];
```

- Normalizing the eigenvector u such that $sum(x) = 1$.

```
>>  x = u/sum(u); % here x is the PageRank vector.
```

- In the present case the PageRank vector is:

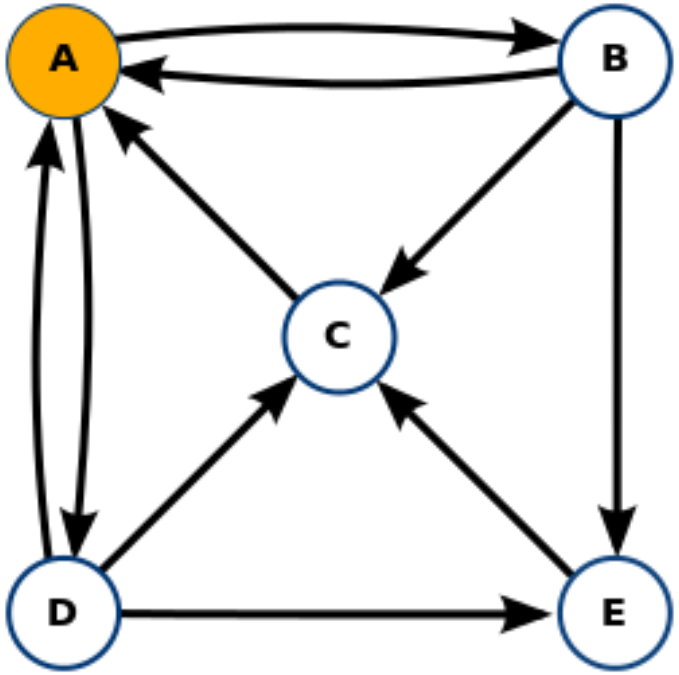
$$x = \begin{bmatrix} 0.3871 \\ 0.1290 \\ 0.2903 \\ 0.1935 \end{bmatrix},$$

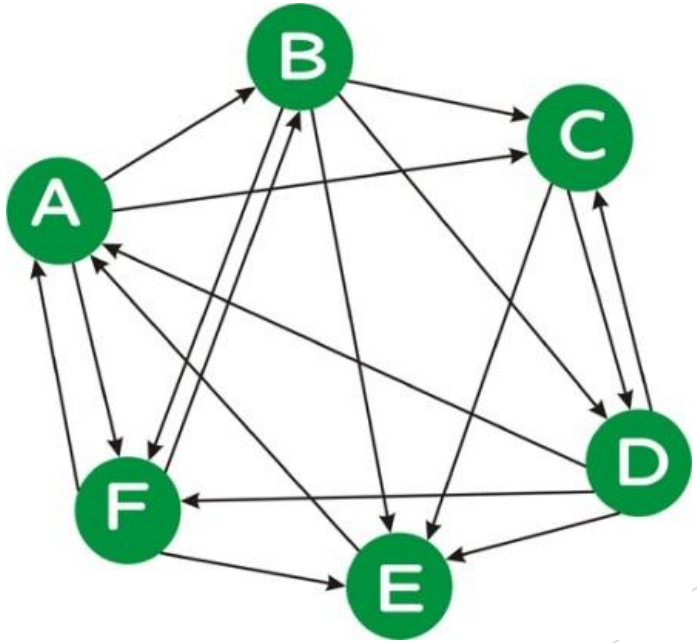
which suggests that the Page #1 is the most important page and Page #2 is the least important page in the given web.

- Check whether $A\,x = x$???

Exercise Problems

- Write the transition matrix for the webs shown below and arrange the pages in the order of their importance.





References

- PageRank – Wikipedia
- Notes on PageRank by Chris Mccormick @ WordPress
- The World' s Largest Matrix Computation – Math Works
- Google PageRank – Math Works
- PageRank Algorithm - The Mathematics of Google Search – Cornell University