大数据分析实验
PageRank

華中科技大學
计算机科学与技术学院
School of Computer Science &Technology，HUST

目录

一 任务背景

一 任务背景

- 1、学习pagerank算法并熟悉其推导过程；
- 2、实现pagerank算法，理解阻尼系数的作用；
- 3、将pagerank算法运用于实际，并对结果进行分析

二 任务描述

二 任务描述

◆基本任务

- 利用实验一得到的出现次数最多前1000个的title之间的引用关系<title,<title1,…,titlek>>，由title为节点构造有向图，编写pagerank算法的代码，根据每个节点的入度计算其pagerank值，迭代直到误差小于10^-8

二 任务描述

◆ 进阶任务

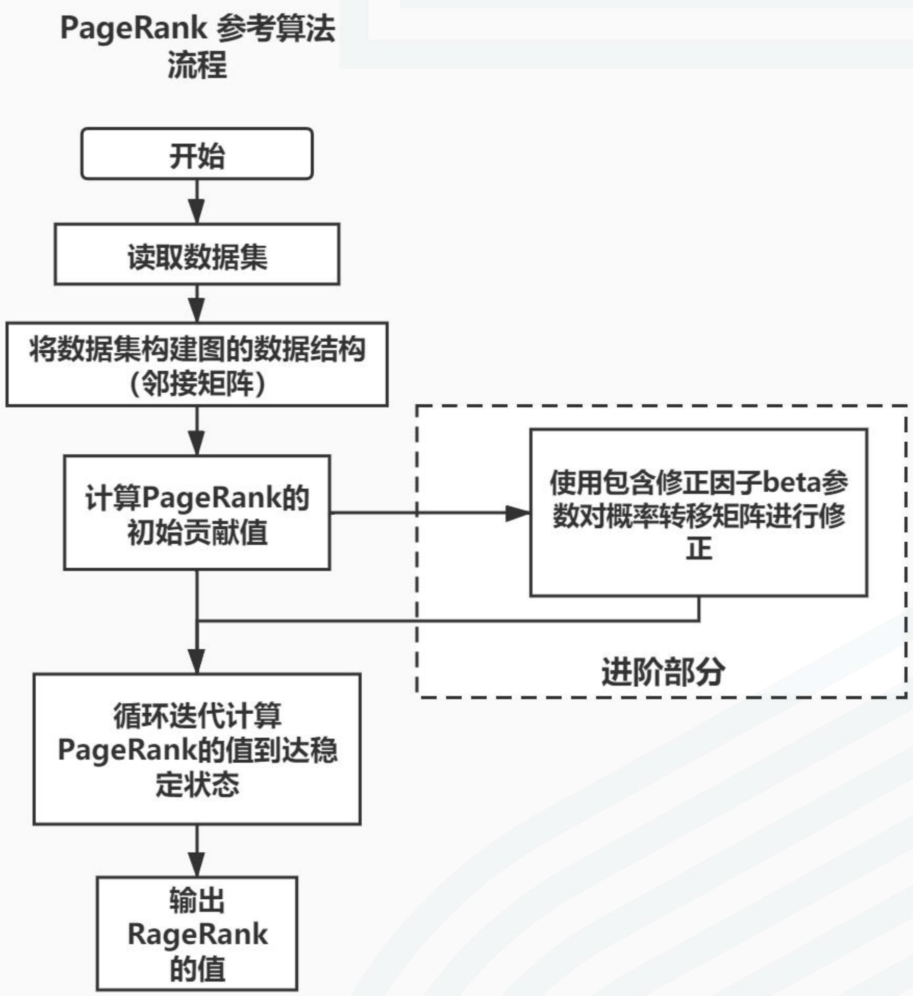
- 实验进阶版考虑加入teleport β，对概率转移矩阵进行修正，解决dead ends和spider trap的问题。

三 算法流程

三 算法流程

华中科技大学
计算机科学与技术学院
School of Computer Science &Technology，HUST

◆ PageRank算法流程：



三 算法流程

◆概率转移矩阵计算与迭代：

- **Stochastic adjacency matrix** M
  - Let page i has d_i out-links
  - If $i \to j$, then $M_{ji} = \frac{1}{d_i}$ else $M_{ji} = 0$
    - M is a **column stochastic matrix**
      - Columns sum to 1
- **Rank vector** r**:** vector with an entry per page
  - r_i is the importance score of page i
  - $\sum_i r_i = 1$
- **The flow equations can be written**

$$r_j = \sum_{i \to j} \frac{r_i}{\mathrm{d}_i}$$

$$\boldsymbol{r} = \boldsymbol{M} \cdot \boldsymbol{r}$$

三 算法流程

◆概率转移矩阵计算与迭代：

- **Given a web graph with *n* nodes, where the nodes are pages and edges are hyperlinks**
- **Power iteration:** a simple iterative scheme
  - Suppose there are *N* web pages
  - Initialize: $\mathbf{r}^{(0)} = [1/N,....,1/N]^T$
  - Iterate: $\mathbf{r}^{(t+1)} = \mathbf{M} \cdot \mathbf{r}^{(t)}$
  - Stop when $|\mathbf{r}^{(t+1)} - \mathbf{r}^{(t)}|_1 < \varepsilon$

$$r_j^{(t+1)} = \sum_{i \to j} \frac{r_i^{(t)}}{\mathrm{d}_i}$$

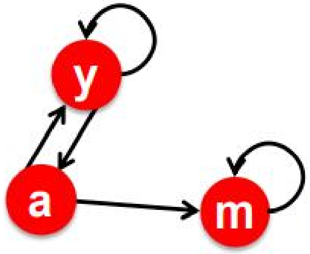
d_i …. out-degree of node i

$|\mathbf{x}|_1 = \sum_{1 \le i \le N} |x_i|$ is the L_1 norm
Can use any other vector norm, e.g., Euclidean

三 算法流程

◆进阶：上述方法中的问题1 Spider Traps

- **Power Iteration:**
  - Set $r_j = 1$
  - $r_j = \sum_{i \to j} \frac{r_i}{d_i}$
    - And iterate



m is a spider trap

	y	a	m
y	½	½	0
a	½	0	0
m	0	½	1

$$r_y = r_y/2 + r_a/2$$
$$r_a = r_y/2$$
$$r_m = r_a/2 + r_m$$

- **Example:**

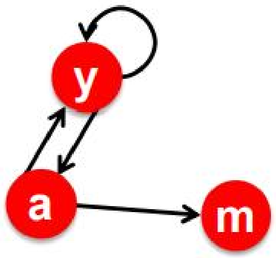
$$\begin{bmatrix} r_y \\ r_a \\ r_m \end{bmatrix} = \begin{matrix} 1/3 & 2/6 & 3/12 & 5/24 & & 0 \\ 1/3 & 1/6 & 2/12 & 3/24 & \ldots & 0 \\ 1/3 & 3/6 & 7/12 & 16/24 & & 1 \end{matrix}$$

Iteration 0, 1, 2, …

All the PageRank score gets “trapped” in node m.

◆进阶：上述方法中的问题2 Dead Ends

- **Power Iteration:**
  - Set $r_j = 1$
  - $r_j = \sum_{i \to j} \frac{r_i}{d_i}$
    - And iterate



	y	a	m
y	½	½	0
a	½	0	0
m	0	½	0

$r_y = r_y/2 + r_a/2$

$r_a = r_y/2$

$r_m = r_a/2$

- **Example:**

$$\begin{bmatrix} r_y \\ r_a \\ r_m \end{bmatrix} = \begin{matrix} 1/3 & 2/6 & 3/12 & 5/24 & & 0 \\ 1/3 & 1/6 & 2/12 & 3/24 & \dots & 0 \\ 1/3 & 1/6 & 1/12 & 2/24 & & 0 \end{matrix}$$

Iteration 0, 1, 2, …

Here the PageRank "leaks" out since the matrix is not stochastic.

三 算法流程

华中科技大学 计算机科学与技术学院 School of Computer Science &Technology，HUST

◆ 进阶：对概率转移矩阵进行修正

- **PageRank equation** [Brin-Page, '98]

$$r_j = \sum_{i \to j} \beta \frac{r_i}{d_i} + (1-\beta)\frac{1}{N}$$

- **The Google Matrix A:**

$[1/N]_{N\times N}$…N by N matrix where all entries are 1/N

$$A = \beta\, M + (1-\beta)\left[\frac{1}{N}\right]_{N\times N}$$

- **We have a recursive problem: $\boldsymbol{r} = \boldsymbol{A} \cdot \boldsymbol{r}$**

  And the Power method still works!

- **What is β ?**
  - In practice $\beta = 0.8, 0.9$ (make 5 steps on avg., jump)

四 验收流程

四 验收流程

- title及其对应的pagerank值；
- 验收时对代码的大致解释；
- 验收时的提问与回答。