

# 迭代方法求PageRank

---

[2017-02-17 Updated: 2017-02-17](#) 5.6K

本文介绍如何用迭代的方法计算PageRank。

目录 [\[hide\]](#)

- [1. PageRank](#)
- 2. [迭代方法](#)
  - [2.1 初始化节点PR值](#)
  - [2.2 创建随机图 \(stochastic graph\)](#)
  - [2.3 迭代计算](#)

## 1. PageRank

---

博文《[网页排序算法PageRank](#)》介绍了PageRank，计算PageRank可以用迭代的方法也可以用代数的方法，其背后的数学基本运算是是一样的，即：

$$PR(p_i) = \frac{1-d}{N} + d \sum_{p_j \in B(p_i)} \frac{PR(p_j)}{L(p_j)}$$

下文结合图1介绍如何用迭代方法求PageRank。

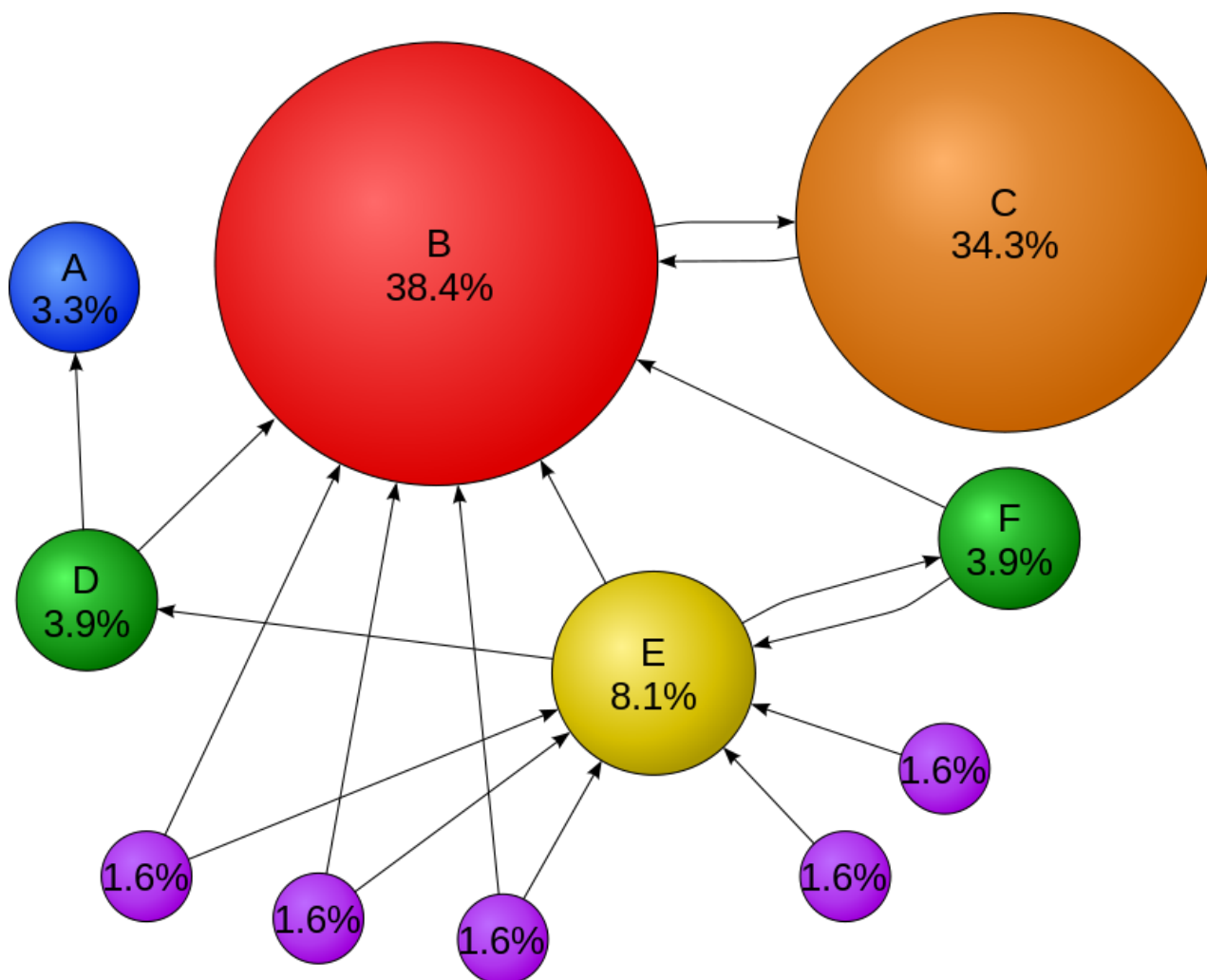


Fig. 1: PageRanks for a simple network (image from [Wikipedia](#)).

为了便于讨论，将图1下方的节点分别标上 $G, H, I, J, K$ ，如下图所示：

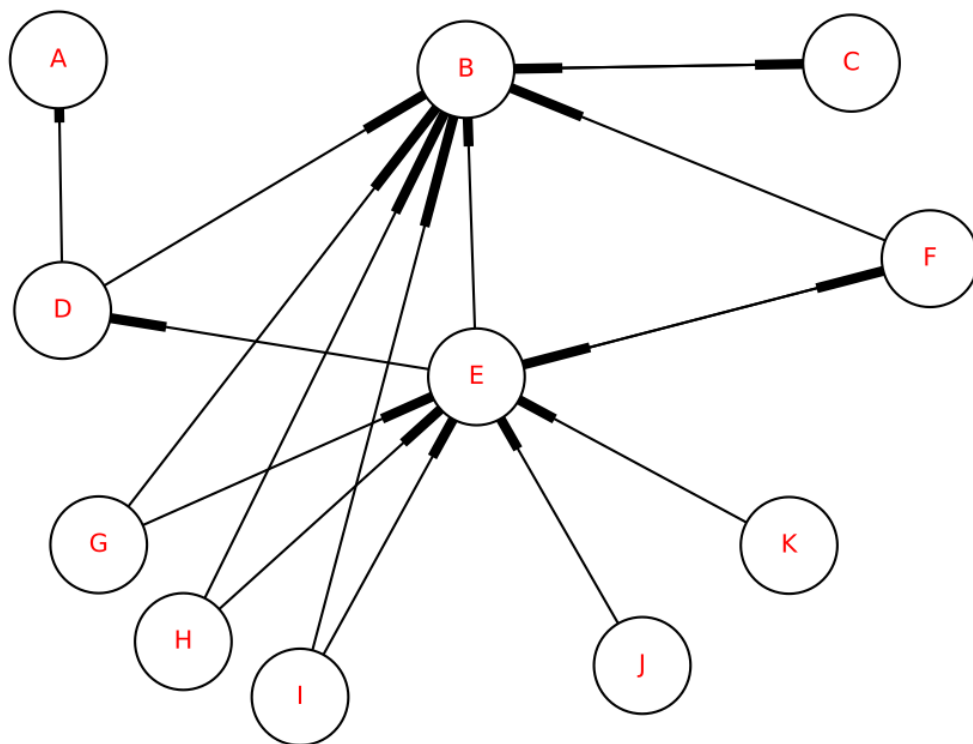


Fig. 2: Label nodes in Fig. 1.

## 2. 迭代方法

### 2.1 初始化节点PR值

如果没有给节点指定PR初始值，那么每个节点的PR初始化为 $1/N$  ( $N$ 为节点数目)，以图1为例，节点的PR初始值为 `1/11`：

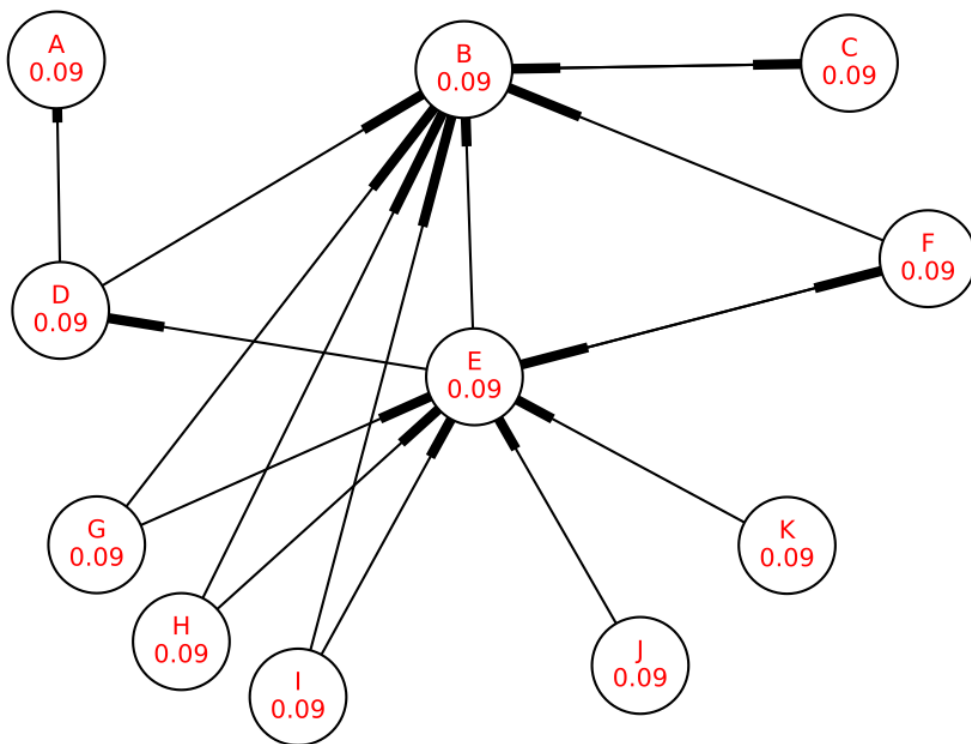


Fig. 3: The graph with starting value of PageRank iteration for each node.

## 2.2 创建随机图 (stochastic graph)

随机图 (stochastic graph) 是一个有向带权图，边的权重被normalized，使得每个节点的outedges的权重加起来为1。事实上，边的权重即为 $1/L(p_j)$ ，图1的随机图如下：

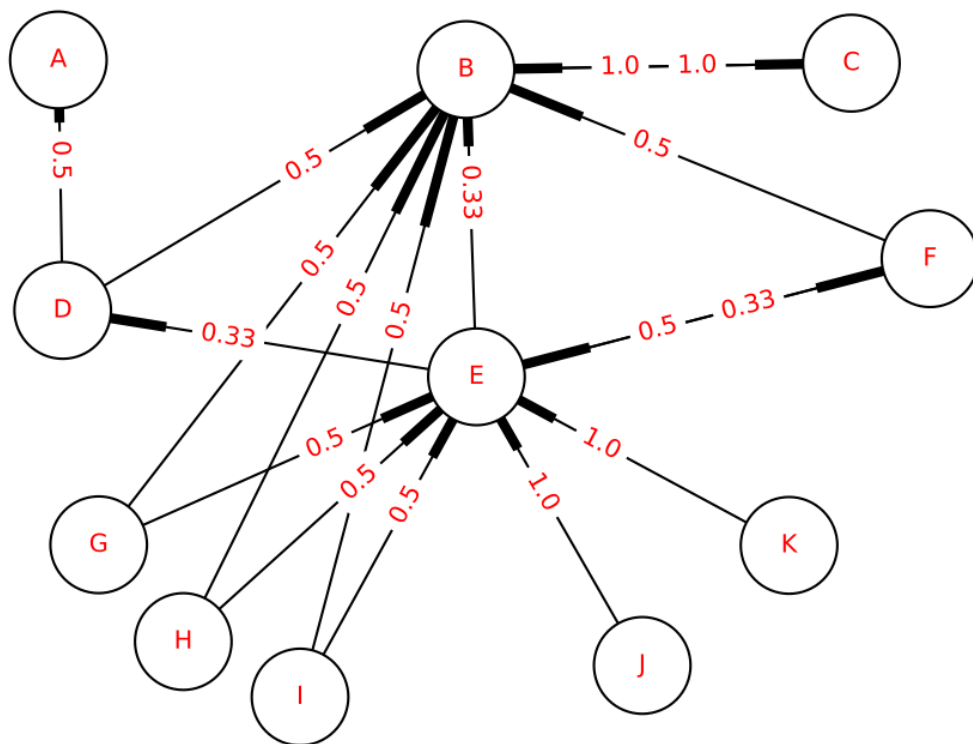


Fig. 4: The stochastic graph

比如，节点 D 有两条出链， $D \rightarrow A$  和  $D \rightarrow B$ ，所以他们的边权重都是0.5。

## 2.3 迭代计算

遍历所有节点，将每个节点的PR值平均分给其出链的节点，即  $\sum_{p_j \in B(p_i)} \frac{PR(p_i)}{L(p_i)}$ ，乘以阻尼系数  $d$ ，再加上  $(1 - d)/N$ 。第一次迭代结果如下图所示（有些箭头没显示出来，NetworkX可视化很弱）：

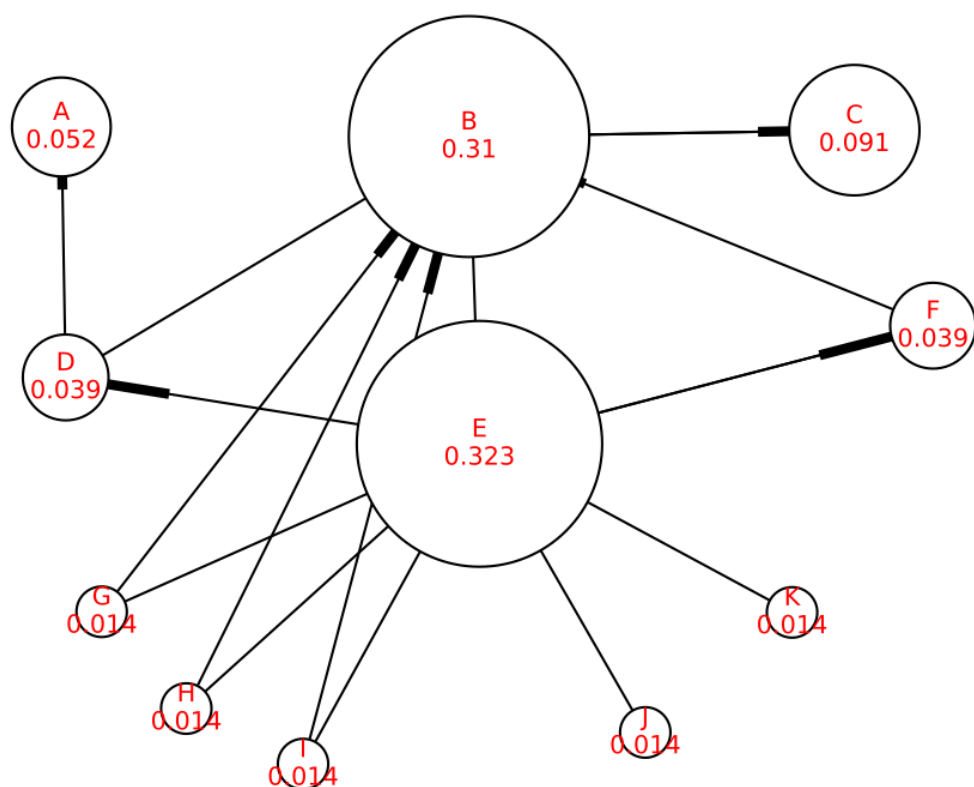


Fig. 5: PageRank after one iteration

那什么时候程序结束呢。将迭代后的PR值跟前一次比较，如果差别很少（如  $PR'(A) - PR(A) < 1.0e - 6$ ），就可以停止迭代了。

在本例中，需要66次迭代，最后得到的PageRank，如下图：

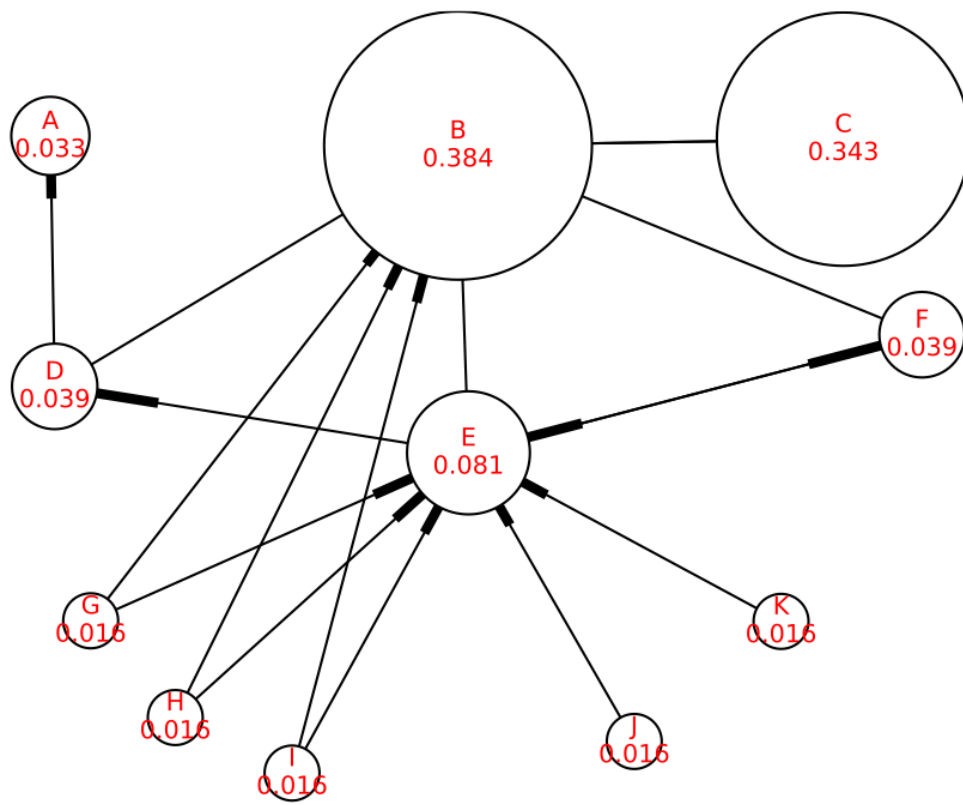


Fig. 6: Stable PageRank values (66 iterations)