PageRank 算法简介

一、 背景介绍

搜索引擎的工作步骤大致可以分为爬虫爬取网页信息、建立网页索引、处理用户搜索的请求、返回用户请求的查询结果。那么在第四步中，如何对返回的网页集合进行排序成为搜索引擎的核心问题，本文所要着重介绍的 PageRank 算法则是为解决这一问题而诞生的。

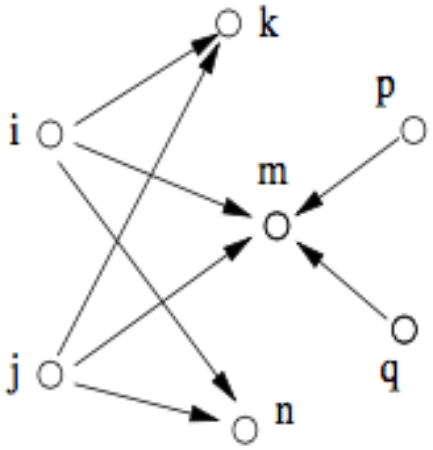
PageRank 算法由佩奇(Larry Page)和布林(Sergey Brin)在 1996 年提出，当时的他们还在斯坦福读研究生，此后他们根据自己提出的算法模型注册了公司，也就是如今大名鼎鼎的谷歌公司。

二、 正文

1. 模型假设
   a) 数量假设：在 Web 图模型中，如果一个页面节点接收到的其他网页指向的入链数量越多，那么这个页面越重要。
   b) 质量假设：指向页面 A 的入链质量不同，质量高的页面会通过链接向其他页面传递更多的权重。

   在开始阶段，模型分配给每一个页面相同的权重值，然后通过迭代不断更新每个页面的权重值，直到模型收敛，便能求得最优解。因此，我们可以发现 PageRank 实际上是主题无关的，即与用户的搜索请求无关，在文章的拓展延伸部分我们还会介绍一种 PageRank 的一种变体－主题敏感 PageRank。
2. 基本思想

   假设网络拓扑图如图所示：



假如我们要计算 m 网页的权重值，那么我们需要所有计算 $\sum_{(j,\ m)\in E} W(j) * P$，也就是所有链接到 m 的页面的权重乘以他们跳转到这个页面的概率，而这个概率通常通过每个节点出度的倒数来衡量，这样就是满足一个节点跳转到其他节点的概率总和为 1 了。

对于 m 来说，其权重则为

$$w(m) = w(p) + w(q) + \frac{w(j)}{3} + \frac{w(i)}{3}$$

为了方便我们的计算我们将其表示为矩阵的形式，我们定义一种概率转移矩阵 p，$P^T = L^T D_{out}^{-1}$，其中 L 为网络的邻接矩阵，而 D 则为表示出度的对角矩阵，因此概率转移矩阵 P 是随机矩阵，因为其每行的元素之和为 1。

通过将初始权重向量$\underset{x}{\rightarrow}$与概率转移矩阵相乘，得到新的权重向量，然后不断迭代这个过程，当新的权重向量与上一次的权重向量的差异控制在一定的阈值之内时，我们就说模型达到收敛，此时得到的权重向量即为最终解。

假设H_0为初始分布，则计算过程如下

$$H_n = P^n H_0$$

因此我们可以看到实际上 PageRank 是一种随机游走的模型，蕴含着马尔科夫随机链的思想，它模拟出一个虚拟的人，通过其在网络上的随机游走来将网页权重进行传递，当模型稳定之后，权重也会达到一个最终的收敛值。

3. 核心问题

a) 处理“悬挂网页”问题－随机性修正

什么是悬挂网页？悬挂网页即是那些没有对外链接的网页，假如随机游走模型走到悬挂网页时，便没有可能从悬挂网页走出，导致其对应的行向量整体为 0，不符合我们的转移概率的定义。

作如下修正：

$$S = P + ea^T/N$$

用数学语言来说， 这相当于是把 H 的列向量中所有的零向量都换成 e/N (其中 e 是所有分量都为 1 的列向量，N 为互联网上的网页总数)。如果我们引进一个描述 “悬挂网页” 的指标向量 a，它的第 i 个分量的取值视 Wi 是否为 “悬挂网页” 而定——如果是 “悬挂网页”， 取值为 1，否则为 0。

b) 素性修正

实际上我们在网络上浏览时，不仅有可能通过点击当前页面的链接来浏览下一个页面，还有可能直接打开另外一个网页，因此佩林和布林将这部分可能添加到模型当中去，称之为素性修正。

$$G = \alpha S + (1-\alpha)ee^T/N$$

其中 e 为单位向量，修正之后的模型有$1-\alpha$的概率跳转到与当前无关的网页中去。

我们可以观察到通过这样的素性修正，整个矩阵变为了正矩阵（将某些 0 元素变为正数），根据马尔科夫随机链中的性质：当转移概率矩阵为素矩阵（A^n>0 则 A 为素矩阵，n>0）时,马尔科夫过程存在极限分布，就是其平稳分布。通过素性修正，模型的合理性与准确性得到了数学上的证明。

c) 关于α参数

实际上 α参数与模型的收敛速度有关，参数越小，模型的收敛速度越快，但是又不能太小，因为不能影响到模型原本网页排序思路。因此α一般都在 0.8~0.9 之间，通常取 0.85。

三、 话题延伸

1. 主题敏感 PageRank

在前边的讨论中我们已经得出 PageRank 是主题无关的结论了，但是如何根据用户不同的搜索请求显示不同的排序结果呢？

主题敏感 PageRank 模型分为以下几步来解决这个问题：

1) 划分网页的种类
2) 确定网页的归属类别
3) 在每个类别内作 PageRank 算法
4) 根据用户搜索请求返回固定类别的网页排序结果

在主题敏感模型进行 PageRank 算法时，其转移概率矩阵计算公式如下：

$$G = \alpha S + (1-\alpha)\frac{s}{|s|}$$

其中 s 即为主题的指示向量，当此网页属于这个主题时，对应元素为 1，否则为 0。

2. HITS 算法

与 PageRank 算法基于随机游走模型不同，HITS 算法基于一种相互增强的模型。HITS 将页面分为两类：Hub 页面与 Authority 页面。

算法基于两个假设：

a) 好的 Authority 页面会被很多好的 Hub 页面指向
b) 好的 Hub 页面会被指向很多好的 Authority 页面

Authority 页面与 Hub 页面相互增强，构成了 HITS 算法模型。

假设$\underset{x}{\rightarrow}$代表 Authority score, $\underset{y}{\rightarrow}$代表 Hub score，其增强过程如下所示：

$$x_i = \sum_{j:e_{ji}\in E} y_j, \quad y_i = \sum_{j:e_{ij}\in E} x_j$$

其迭代过程如下所示：

$$\begin{aligned} c\mathbf{x}^{(t+1)} &= \mathcal{I}^{op}(\mathcal{O}^{op}(\mathbf{x}^{(t)}) = L^T L\mathbf{x}^{(t)} \\ c\mathbf{y}^{(t+1)} &= \mathcal{O}^{op}(\mathcal{I}^{op}(\mathbf{y}^{(t)}) = LL^T\mathbf{y}^{(t)} \end{aligned}$$

其中 c 代表归一化常量，L 代表邻接矩阵，我们可以发现实际上迭代过程的终点，也就是最优值实际上是$L^T L$与LL^T的特征向量，也就是我们对 L 对 SVD 分解所用到的特征向量。

四、 总结

当今许多搜索引擎的主要算法都是在 PageRank 算法的基础上形成的，例如利用稀疏矩阵简化矩阵运算等，通过这篇文章希望大家对搜索引擎有一个笼统的认识并且对算法背后数学知识有一定的了解，发掘自己对科研的兴趣，树立远大的目标。

五、 引用文献

1) http://blog.csdn.net/hguisu/article/details/7996185
2) http://www.changhai.org/articles/technology/misc/google_math.php 谷歌背后的数学

3) PageRank, HITS and a Unified Framework for Link Analysis, LBNL Tech Report 49372, Nov 2001

4) Google PageRank and Beyond: The Science of Search Engine Rankings