

结合网页内容分析的 PageRank 算法初探

李树青

(南京财经大学信息工程学院 南京 210000)

摘要 作为一种相当成功的基于超链分析的算法,PageRank 算法可以有效地衡量网页重要度权值,然而进一步的研究也表明,这种纯粹依赖于超链分析的算法由于没有考虑到网页内容对网页重要度权值的影响,所以在一定程度上会造成偏差。因此,合理的将两者进行结合,充分利用网页内容特征对 PageRank 算法进行改进,可以极大的提高这种算法的有效性。

关键词 超链分析 PageRank 算法 优化

1 概述

超链分析技术主要是指利用网页间存在的各种超链指向,对网页之间的引用关系进行分析,依据网页链入数的多少计算该网页的重要度权值,一般认为,如果 A 网页有超链指向 B 网页,相当于 A 网页投了 B 网页一票,即 A 认可 B 网页的重要性。深入理解超链分析算法,可以根据链接结构把整个网页文档集看作一个有向的拓扑图,其中每个网页都构成图中的一个结点,网页之间的链接就构成了结点间的有向边,按照这个思想,可以根据每个结点的出度和入度来评价网页的作用。

其中有代表性的算法主要是 Larry Page 等人设计的 PageRank 算法和 Kleinberg 创造的 HITS 算法。其中 PageRank 算法在实际使用中的效果要好于 HITS 算法,这主要是由于以下原因:a. PageRank 算法可以一次性、脱机并且独立于查询地对网页进行预计算,以得到网页重要度的估计值,然后在具体的用户查询中,结合其他查询指标值再一齐对查询结果进行相关性排序,从而节省了系统查询时的运算开销;b. PageRank 算法是利用整个网页集合进行计算的,不像 HITS 算法易受到局部连接陷阱的影响而产生“主题漂移”,所以现在这种技术广泛地应用在许多信息检索系统和网络搜索引擎中,Google 搜索引擎的成功也表明了以超链分析为特征的网页相关度排序算法日益成熟。

但是 PageRank 算法由于只考虑到网页间的超链关系并仅仅以此进行网页重要度的分析,所以不可避免地会产生很多问题,其中,比较明显的问题在于它在计算每个网页具体的重要度权值的时候,根本没有考虑到任何网页本身内容特征对权值的影响,如 PageRank 算法完全忽略了网页具有不同的主题,不同的主题应该具有不同的重要度权值,进一步说,在用户查询的时候,网页重要程度值的大小与查询所表达的主题关系很大,其实,在 HITS 算法中恰恰考虑了这种因素,所以它更易于表达与特定查询主题相关的相关度排序,有效地在 PageRank 算法中考虑查询主题对网页权重值的影响是一个有效改进此算法的重要方法;再如,PageRank 算法也没有考虑网页的创建时

间,并不对新旧网页进行有效的区分,相反,按照 PageRank 的既有算法甚至会产生旧网页比新网页具有较高重要度权值的可能性。这些都是本文准备要解决的问题。

2 传统 PageRank 算法回顾

虽然 PageRank 认为网页的链入超链数可以反应该网页的重要程度,但是现实中人们在设计网页的各种超链时往往并不严格,有很多网页的超链纯粹是为了诸如网页导航、商业推荐等目的而制作的,显然这类网页对于它所指向网页的重要度贡献程度并不高,但是,由于算法的复杂性,PageRank 没有过多考虑网页超链内容对网页重要度的影响,只是使用了两个相对简单的方法:其一,如果一个网页的链出网页数太多,则它对每个链出网页重要度的认可能力相应降低;其二,如果一个网页由于本身链入网页数很低造成它的重要度降低,则它对它的链出网页重要度的影响也相应降低。综上所述,一个网页的重要性决定着同时也依赖于其他网页的重要性。

按照这个思路,Page 给出了 PageRank 的简单定义:

$$\forall_u R(u) = C \sum_{v \in B(u)} R(v) / N(v)$$

此处的 u 表示一个网页, $R(u)$ 表示网页 u 的 PageRank 值, $B(u)$ 表示链接到网页 u 的网页集合,即网页 u 的链入网页集合, $N(v)$ 表示从网页 v 向外的链接数量,即网页 v 的链出网页数, C 为规范化因子,用于保证所有网页的 PageRank 总和为常量(如为 1)。

具体计算时,可以给每个网页一个初始的 PageRank 值,然后反复迭代运算,即:

$$\forall_v R^{(i+1)}(v) = \sum_{u \in B_v} R^{(i)}(u) / N_u$$

此处的 v 代表所有的网页集合,每一个第 $i+1$ 次的 PageRank 值都是基于上次的 PageRank 值重新计算的。具体的迭代次数在实际运算中是有限的。

上述过程在本质上可以表达为特征向量的计算,首先每个网页文档的 PageRank 值可以表示一个向量,即一个 N 行 1 列的向量(N 为所有的网文档数),为了便于计算,开始时可以给每

个元素的值设为 $1/N$ 。

$$\text{Rank} = [1/N]_{n \times 1}$$

设 M 为一个随机矩阵,它的纵横行列数分别为整个网页集合的文档数,每个矩阵元素值表示两两网页之间的链接关系,即如果网页 D_i 指向 D_j ,则矩阵元素 M_{ij} 对应的值为 $1/N_i$ (N_i 表示 D_i 的链出网页数);如果网页 D_i 不指向 D_j ,则 M_{ij} 值为 0。

$$M = \begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1n} \\ m_{21} & m_{22} & \cdots & m_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ m_{n1} & m_{n2} & \cdots & m_{nn} \end{bmatrix}$$

所以,PageRank 值的计算就可以表示为:

$$\text{Rank} = M \times \text{Rank}$$

而且这个过程是个反复迭代的过程,直至 Rank 值最终收敛。但是,实际的网页结构并非表现为一个完全牢固的链接图,不是所有的网页都可以从其他网页通过超链接来达到,而 PageRank 值的计算正依赖于此,所以 Page 等人就提出了改进方案,对存在的等级沉没(Rank Sink)和等级泄漏(Rank Leak)等问题进行了有效的解决。整个网页图中的一组紧密链接的网页如果没有外出的链接就产生等级沉没,一个独立的网页如果没有外出的链接就产生等级泄漏。所以,Page 改进措施为:一是剔除产生等级泄漏的独立网页以消除其不利影响;二是给产生等级沉没的网页添加一个指向链入网页的返回链接,此时使得所有网页 PageRank 值的计算就不完全依赖现有链接了,所以修正的 PageRank 计算公式为:

$$\forall_u R(u) = C \sum_{v \in B(u)} R(v)/N(v) + CE(u)$$

$E(u)$ 是个常量,它可以抑制 PageRank 值的传播,使得所有网页的 PageRank 值至少会为 $E(u)$,而不会为 0。具体的 $E(u)$ 值可以有多种取法,简单的做法可以设为 p ,如取 $1/N$ (N 为网页文档总数)。

从特征向量的角度来考察,可以设置 P 列向量以代表每个网页文档都有的、相同的 $E(u)$

$$P = [1/N]_{n \times 1}$$

设置 D 向量以代表网页文档的链出网页数是否为 0,即

$$D_i = \begin{cases} 1 & \text{如果网页 } D_i \text{ 的链出网页数为 0} \\ 0 & \text{如果网页 } D_i \text{ 的链出网页数非 0} \end{cases}$$

则上述 PageRank 的计算可以进一步表达为:

$$\text{Rank} = C(M + P \times D^T) \times \text{Rank} + CP$$

3 结合网页内容的 PageRank 算法

事实上,PageRank 值虽然在一定程度上表达了网页与用户查询的相关度,但是存在的问题也是明显的,其中最明显的在于 PageRank 的计算是独立于用户查询而进行的前期运算,根本没有考虑用户查询的具体要求,这样,网页的 PageRank 值只是一个基于全部网页集合计算出来的一个独立值。而用户的查询是多变的,同样的查询词语放在不同的查询词语集合中,代表着不同的含义,如“Process”,在“Project Management Process”中代表着“进度”,而在“Windows Process Program”中代表着“进程”,所以,依据用户的查询主题可以得出不同的查询词语义,

进而可以判断出同一个网页应该根据不同的查询主题拥有不同的 PageRank 值才可以表达上述情况。

事实上,有些学者已经证明网页结构易于受到页面主题的影响,如 Chakrabarti 和 Pennock 等。一般来说,一个网页易于链接到其他主题相同的网页,这也正是 PageRank 为什么可以独立于查询进行计算的原因,同时也反过来告诉我们应该使用这种特性来加强基于链接的网页重要度的计算效果。

但是,查询的主题只有在用户输入查询词时才能得到,所以简单地利用查询词的主题计算 PageRank 会导致系统增加查询时的计算开销,造成查询性能的极大下降。所以,最好的做法应该是既能利用查询本身的信息来影响基于连接产生的值,同时也应该减少查询的处理时间,有些学者已经做出一些研究,如 Richardson 和 Domingos 等,他们对所有可能的查询词语计算相应的 PageRank 值,但是此方法过于消耗处理时间和存储,并且不易于使用查询的上下文内容来扩展。

所以,最好的考虑是实现利用已有的主题进行网页不同主题的 PageRank 值的计算,同时在查询时,不直接利用查询词进行主题计算,而是通过归类算法将用户查询词表达的主题概括为一组既有主题,这样既可以加快查询时的运算,同时也能充分考虑用户查询的特点。

具体步骤为:

首先修正 P 向量的定义,原先的 P 向量代表每个网页文档都有的 $E(u)$ 值,在不同的主题限制下,可以进一步将其改进为不同网页文档具有不同的值,即对于主题 T_j ,网页文档 D_i 的 P 向量分量值为 P_{ji} :

$$P_{ji} = \begin{cases} \frac{1}{|T_j|} & i \in T_j \\ 0 & i \notin T_j \end{cases}$$

其中 $|T_j|$ 表示满足此主题的网页总数,此时的 P 向量会在同一主题范围中只考虑当前主题相关文档的词语分布情况,而且不同的主题会对同一网页设置为不同的向量分量。具体在实际计算中,可以使用网页归类算法对网页主题进行判定,如多项朴素贝叶斯分类算法,也可以使用现有的分类目录直接判定,如 Taher H. Haveliwala 利用 Open Directory 项目提供的数据库主题进行网页分类判断。

然后,在用户查询时,根据查询词语进行主题判断,如使用多项朴素贝叶斯分类算法计算查询与每个主题的相关度:

$$P(c_j | q) = \frac{P(c_j) \cdot P(q | c_j)}{P(q)}$$

其中, $P(q | c_j)$ 易于从 P 向量中得到,对 $P(c_j)$ 和 $P(q)$ 的理解并不直观,简单的做法可以设为同一值,复杂的做法可以考虑对于不同的用户使用不同的值来表达用户的偏好度。

最后,利用传统的文本索引查询算法得到与原始查询相关的文档,并利用上述的修正 P 向量来计算每个文档不同主题的 PageRank 值,具体做法为:设 r_{jd} 为网页文档 D 在主题 T_j 限制下的 PageRank 值,即基于主题 T_j 的 PageRank 值,利用一个网页的所有不同主题的 PageRank 值并结合上述的 $P(c_j | q)$ 加权得到最终一篇网页文档对于用户查询的基于主题的 PageRank 值,即

$$\text{Rank}_{jd} = \sum_j P(T_j | q) \cdot r_{jd}$$

从上述说明可以看出,计算结合网页内容的 (下转第 38 页)

$W = (0.1, 0.2, 0.1, 0.2, 0.4)$, “ \circ ”运算采用加权平均的 $M(\cdot +)$

模型,即 $S_i = \sum_{j=1}^5 W_j * r_{ij}$

由此可以计算:

$$S_{\text{生化文献中心}} = \{(\overline{0.081} \quad \overline{0.071}) (\overline{0.138} \quad \overline{0.144}) (\overline{0.216} \quad \overline{0.238}) (\overline{0.043} \quad \overline{0.044}) (\overline{0.013} \quad \overline{0.012})\}$$

$$S_{\text{语言文学文献中心}} = \{(\overline{0.108} \quad \overline{0.13}) (\overline{0.223} \quad \overline{0.243}) (\overline{0.124} \quad \overline{0.112}) (\overline{0.027} \quad \overline{0.023}) (\overline{0.003} \quad \overline{0.007})\}$$

此结果不是很直观,需进一步进行单值化处理为最终评价结果 $V, V = S \circ P$, 其中 $P = (95, 85, 75, 65, 55)$, “ \circ ”运算采用 $M(\cdot +)$ 模型,由此可以计算:

$$V_{\text{生化文献中心}} = ((\overline{39.135} \quad \overline{40.355}) = \overline{79.49}$$

$$V_{\text{语言文学文献中心}} = (\overline{40.435} \quad \overline{43.285}) = \overline{83.72}$$

3.5 评价结果分析 由上述综合评价结果表明,语言文学文献中心的综合服务质量(83.72)要好于生化文献中心(79.49),但两个室都正在向好的方向发展,未来服务质量都有望更好。

运用同样方法,通过计算发现:

生化文献中心的单项指标评价结果分别为:服务环境88.7、服务技能76.4、服务效率75、服务态度84.4、服务效果77.4

语言文学文献中心的单项指标评价结果分别为:服务环境84.5、服务技能81.3、服务效率77.1、服务态度88.6、服务效果84.1

可见,生化文献中心的服务环境虽然目前较语言文学文献中心好,但后者正日渐变好,而前者有恶化之趋势;语言文学文献中心工作人员的服务技能较生化文献中心的高,但两者都在提高;生化文献中心的服务技能虽然目前不如较语言文学文献中心,但前者正日渐提高,而后者有下降之势;语言文学文献中心工作人员的服务态度及服务效果较生化文献中心的好,且日后将更好。

生化文献中心人员应向语言文学文献中心人员学习,借鉴他们的经验,改善服务态度、提高服务技能、提升服务效果;语言文学文献中心人员在提高服务效率方面可向生化文献中心人员讨

教其近期的工作方法;生化文献中心的服务环境虽然不错,但其自骄现象应引起重视。通过全馆12个阅览室的评价结果,可以对比全馆所有阅览室的服务质量优劣,也能勾勒出全馆服务质量图,从而在全馆层面制订各个室之间的互帮互学计划,达到提高全馆服务质量的自的。

4 结束语

科学的评价有助于管理部门准确地获取信息、指导决策、促进工作质量的提高。动态模糊综合评价法利用动态模糊理论,采用统计学的方法,尽量避免个人人为因素的影响,使得评价结果具有一定的科学性和客观性,而且操作简单。

动态模糊综合评价不仅考虑了评价过程中的模糊性,也融入了动态性,不仅能反映图书馆不同等级的模糊程度,而且可以对将来的情况进行评估或给出预测;而且此方法把动态模糊现象与数学方法统计在一起,从而将定性评价转化为定量评价,对图书馆的工作做出科学公正的评价。动态模糊综合评价,能更客观地反映图书馆的综合评价结果,应该成为图书馆未来评价的主要方法。

此方法可以推广到资源评价、人员考核等过程中,减少主观因素,客观全面动态地给出评价结果。

参考文献

- 1 李凡长,朱维华著. 动态模糊逻辑及其应用. 昆明:云南科技出版社,1997
- 2 李海涛等. 图书馆服务质量评价分析. 情报杂志,2003;(9)
- 3 刘文彬. 模糊综合评价系统研究与实现[学位论文]. 天津:河北工业大学,2003
- 4 朱以彬. 网络环境下高校图书馆服务模式及读者满意度研究. 中国科学技术信息研究所硕士学位论文
- 5 郑全太. 模糊论在图书馆评价中的应用研究. 图书馆工作与研究,1997;(2)

(责编:梅王京)

(上接第35页) PageRank 值主要是利用修正过的 P 向量来实现的,其实这种思想完全可以用在其他方面,比如由于 PageRank 值的计算依赖于网页间的超链,所以,旧网页比新网页一般具有更多地被链入的可能,所以通常旧网页的 PageRank 值要高于新网页,而这一点恰恰使得用户不易于获取新的网页。因此,通过合理地调整 P 向量来表达网页的新旧程度对网页重要度权值的影响,就可以在一定程度上克服这个缺点。如设网页的创建时间为 t_i ,显然创建时间越久,相应的 P 向量分量权重就越小,设 D 为搜索引擎本次在计算新的 PageRank 值时重新从网络上下载网页并更新数据库的日期时间, d_i 为某一下载的网页制作日期时间, $d_i - D$ 的值就能反映网页的新旧程度,据此将 P 向量表达为:

$$P_{ji} = \begin{cases} \frac{d_i - D}{T_j | \sum (d_i - D)} & i \in T_j \\ 0 & i \notin T_j \end{cases}$$

这种定义的缺点在于对主题相关的旧网页附加过低甚至为负的权重,所以,合理的安排应该允许用户自己调节对网页

主题相关和网页时间新旧的偏好度,再次定义为:

$$P_{ji} = \begin{cases} \frac{\alpha}{T_j |} + \frac{(1 - \alpha)(d_i - D)}{\sum (d_i - D)} & i \in T_j \\ 0 & i \notin T_j \end{cases}$$

参考文献

- 1 Google inc. <http://www.google.com>
- 2 Taher H. Haveliwala. Topic - Sensitive PageRank: A Context - Sensitive Ranking Algorithm for Web Search. <http://dbpubs.stanford.edu:8090/pub/2002-6/topic-sensitive-pagerank-tkde.pdf>
- 3 The Google Search Engine: Commercial Search Engine Founded by the Originators of PageRank. <http://www.google.com/>
- 4 宋聚平,王永成等. 对网页 PageRank 算法的改进. 上海交通大学学报,2003;(3)
- 5 许涛,吴淑燕. Google 搜索引擎及其技术简介. 信息检索技术,2003;(4)
- 6 阎放等. Google 搜索引擎 PageRank 技术的优化. 情报科学,2002;(12)
- 7 曹军. Google 的 PageRank 技术剖析. 情报杂志,2002;(10)
- 8 宋建康,张礼平. Web 结构挖掘算法探讨. 华东理工大学学报,2003;(10)

(责编:梅王京)