文章编号: 1006-3080(2000)05-0455-04

信息检索中基于链接的网页排序算法

王 奇, 宋国新*, 邵志清
(华东理工大学计算机科学与工程系, 上海 200237)

摘要: 介绍超链接环境下基于链接的网页排序算法, 比较和分析了 PageRank 算法和 HITS 算法, 指出了 PageRank 算法更适合于搜索引擎的服务器端, 而 HITS 算法更适合于搜索引擎的客户端。还构造并初步实现了在信息检索中, 应用超链接环境下网页排序算法的综合模型。

关键词: 信息检索; 超链接; 网页排序; PageRank; HITS
中图分类号: G354.4 **文献标识码**: A

Link-based Ranking Algorithms in Information Retrieval

WANG Qi, SONG Guo-xin, SHAO Zhi-qing*
(*Department of Computer Science and Engineering ECUST, Shanghai* 200237, *China*)

Abstract: Link-based ranking algorithms in hyperlinked environment are discussed in this paper. The comparison and analysis of PageRank and HITS shows that PageRank is more suitable to the server of a search engine while HITS to the client. An integrated model using Link-based ranking algorithms is constructed and initially implemented.

Key words: information retrieval; hyperlink; ranking; PageRank; HITS

万维网是 Internet 信息的主要载体之一, 通过超链接万维网被连接成一个网络, 人们沿着这些超链接可以漫游网络。搜索引擎可以帮助人们尽快地找到所需要的信息, 但是目前多数搜索引擎是基于分类或关键词逻辑组配的检索方式, 用户的一个查询请求往往会检索出庞大的结果集, 而用户所需要的信息却只是其中一小部分, 面对如此多的结果, 用户仍然不知所措, 因此, 如何提供一些有效的工具和方法, 帮助人们高效地获取所需信息是研究者们所面对的重大课题。

人们曾经对具有代表性的三个搜索引擎 Yahoo、Hotbot 和 Excite 作了测试和比较。通过对与信息检索密切相关的 31 个检索表达式检索, 发现 Yahoo 检索结果中仅有 37% 的有用信息, Hotbot 检索结果中仅有 5.6% 的有用信息, Excite 有用信息所占比例更小。显然, 在普通的万维网使用者面前, 搜索引擎的性能比上述测试的结果还要差。提高搜索引擎检索精度的研究很多, 除了提高索引库质量外, 主要有以下方面:

(1) 自然语言理解。用户通过输入一句自然语言来替代原有的关键词和逻辑操作组配, 通过对查询请求的精确理解来使结果更接近于要求。

(2) 排序。通过计算出检索结果中每个网页的价值, 对检索结果进行排序。

(3) 聚类(Clustering)。将庞大的结果集进行自动分类。

(4) 智能代理。通过对用户日常检索过程的学习, 获得用户的个人检索习惯, 从而使处理查询时有针对性。

计算网页价值的一种切实有效的途径是利用万维网链接结构本身所包含的丰富信息。本文首先介绍超链接环境下的网页排序算法, 比较和分析了 PageRank 算法[1]和 HITS 算法[2], 指出了 PageR-

基金项目: 上海市科技发展基金(995114022)和上海市青年科技启明星计划资助项目
收稿日期: 2000-04-21
作者简介: 王 奇(1974-), 男, 上海人, 硕士研究生, 主要研究方向为信息检索和网络数据库等。

ank 算法更适合于搜索引擎的服务器端,而 HITS 算法更适合于搜索引擎的客户端。本文还构造并初步实现了在信息检索中,应用超链接环境下网页排序算法的综合模型。

1 网页排序算法

社会网络、文献引用关系图、万维网等都可以表示成图 $G=(V,E)$;邻接矩阵 A 表示链接关系,A_{ij} 表示每条边的权值,可以理解为节点 i 对 j 的认可程度。

Katz[3]在研究社会网络的基础上,提出了一种计算名望(Standing)的基于路径的方法:$P_{ij}^{\langle r\rangle}$ 表示从 i 到 j 长度为 r 的路径的数目,衰减系数 b 为小于 1 的常数,$Q_{ij}=\sum_{r=1}^{\infty}b^rP_{ij}^{\langle r\rangle}$ 表示 i 与 j 的耦合度,$s_j=\sum_{i=1}^{\infty}Q_{ij}$ 表示 j 点的名望。通过矩阵变换可以得到一个比较直接的矩阵公式 $(I-bA)^{-1}-I$,其中第 j 列的和就是 s_j。Hubbell[4]提出了网络节点的权值传播模型,e_j 表示一个对于 s_j 的初始估计,$s_j=e_j+\sum A_{ij}s_i$,通过矩阵变换,可以得到向量 $s=(I-A^T)^{-1}e$。

Garfield[5]提出的影响因子(impact factor)一直是科学期刊权威性和影响力的评判标准,影响因子一般都通过引用关系来计算,比较简单的是计算某份期刊中论文被引用次数的平均值,这纯粹是对图中节点入度的计算。Pinski 和 Narin[6]提出了一种改进方法,定义科技期刊 j 的影响力权值为 $w_j=\sum A_{ij}w_i$,所以对权值向量 w 有 $A^Tw=w$,w 为 A^T 的主特征向量。而 Geller[7]则发现这种方法恰恰反映了以下一种随机过程,从任意一个期刊 j 开始,随机选择在 j 中出现的被引用期刊。Doreian[8]类似地再对 Hubbell 的权值传播模型设计了一种相应的递归计算方法来计算名望。

1992 年,Botafogo[9]等人,定义了在超文本环境下的 index 节点(入度大于平均值)和 reference 节点(入度小于平均值)。1997 年,Carrière 和 Kazman[10]提出网页的一种排名由它的出度和入度之和决定,还是没有利用到万维网的有向特性。1998 年,Page[1]提出了 PageRank 算法,并具体应用到了搜索引擎 google[11]中,这类似于 Pinski-Narin 计算影响力权值的权值传播模型的网页排名方法。Kleinberg[2]提出 HITS 算法,将网页分成 hub 网页和 authority 网页,通过迭代可以最终得到排名最高的 hub 网页和 authority 网页。

下面将分析近期具有代表性的两个网页排序算法:PageRank 算法和 HITS 算法。

1.1 PageRank 算法

PageRank 算法是 Page L 提出的,并在优秀的搜索引擎 google 中得到了应用。PageRank 的递归描述如下:如果一个网页的所有入链(backlinks)的排名之和比较高,则这个网页有比较高的排名。网页 u 的入链为指向 u 的链接,出链为从 u 出发的链接。令 F_u 为网页 u 指向的所有网页的集合,$N_u=|F_u|$,即 u 的出度;B_u 为指向 u 的所有网页的集合,常系数 $c<1$ 用于保证所有网页排名值的总和保持为常量。理想情况下网页 u 的排名值 R_u 可由以下公式计算

$$R_u=c\sum_{v\in B_u}\frac{R_v}{N_v}$$

定义邻接矩阵 A,如果存在 u 到 v 的超链,$A_{u,v}=1/N_u$,否则为 0,上面的公式可以转化为 $R=cAR$,所以取 R 是 A 的主特征向量,c 为主特征值,因此保证了计算过程的收敛性,经过对初始向量的多次迭代可以得到收敛后的 R。

以上方法的缺点是,当两个页面互相指向,但不指向任何其他页面,并存在指向这两个页面的链接,在迭代中,将造成陷阱,不断地累加排名值而不传递出去。为此,提出了改进的模型:

$$R'_u=c\sum_{v\in B_u}\frac{R'_v}{N_v}+cE_u$$

其中,E_u 表示网页 u 的初始排名,c 取最大可能值,且 $\|R'\|_1=1$。上面公式可转化为,$R'=c(AK+E)$,又因为 $\|R'\|_1=1$,可以得到 $R'=c(A+E\times l)R'$,其中 l 为全 1 向量,所以 R 是 $(A+E\times 1)$ 的特征向量。取 S 为 Web 页面上的任意向量(如 E),计算 PageRank 的基本算法如下:

$R_0\leftarrow S$

loop:

$R_{i+1}\leftarrow AR_i$

$d\leftarrow\|R_i\|_1-\|R_{i+1}\|_1$

$R_{i+1}\leftarrow R_{i+1}+dE$

$\delta\leftarrow\|R_{i+1}-R_i\|_1$

while $\delta>\epsilon$

可以看到,d 的适当选择可以加速收敛速度,并影响 E 在计算过程中的作用。在每一步迭代中,可以通过乘上某个因子来对 R 规范化。

万维网中存在大量悬空链接(Dangling links),即指向没有出链的节点链接,在具体计算过程中,这

些链接不直接影响其他节点的排名, 暂时去掉这些链接将大大减少工作量, 同时保持对最终排名没有大的影响。

1.2　HITS 算法

HITS 算法的目标是针对某个查询 σ 得出最有价值的网页。他认为网页可以分为两类, Authorities 和 Hubs, Authorities 为具有较高价值的网页, Hubs 为指向较多 Authorities 的网页。

不同于 PageRank 直接存取搜索引擎的页面索引, 对整个万维网页面进行排名, HITS 从现有的搜索引擎(如 Altavista)中获取查询结果, 从中选出接近目标的 t 个网页作为根集(R_σ), 然后根据这些网页的入链和出链进行前后扩展, 再选出最接近目标的若干网页, 构成目标网页的一个子集(S_0), 令 $\Gamma^+(p)$ 表示 p 指向的网页, $\Gamma^-(p)$ 表示指向 p 的网页。算法如下:

```
Set S0: = R0
For each page p ∈Rσ
  Add all page in Γ+ (p) to Sσ
  If |Γ- (p)|≤d then
  Add all page in Γ- (p) to Sσ
  Else
     Add an arbitrary set of d pages from
Γ- (p) to Sσ
  End If
Loop
Return Sσ
```

最后得到的子集 S_σ 具有规模小、关联页面多、包含 authority 网页尽量多的特点。但它们之间的链接还可以进行筛选: 同一网站上的页面存在很多内部互相链接, 这些链接绝大多数并不表示一种认可, 应当在最后结果中予以删除; 另外, 从一个网站指向另一个网站的同一网页的链接超过一定数目(如 4 个), 应删除这些链接, 因为这很可能是"This site desighed by... "这样的链接。在这样一个经过筛选的网页集合中, 再来计算它们的 authority 权重和 hub 权重。

页面 p 的 authority 权重用 $x^{\langle p\rangle}$ 表示, hub 权重用 $y^{\langle p\rangle}$ 表示, 满足规范化条件, $\sum_{p\in S_\sigma}(x^{\langle p\rangle})^2=1$, $\sum_{p\in S_\sigma}(y^{\langle p\rangle})^2=1$。网页权重的传递分为两种方式, 即 I 操作(hub 到 authority)和 O 操作(authority 到 hub):

I 操作: $x^{\langle p\rangle}\leftarrow\sum_{q:(q,p)\in E}y^{\langle q\rangle}$, (E 为边的集合)

O 操作: $y^{\langle p\rangle}\leftarrow\sum_{q:(p,q)\in E}x^{\langle q\rangle}$,

预先设定迭代次数 k, 迭代算法如下:

令 x_0 和 y_0 为 1(全 1 向量)

For i= 1 to k

　对(x_{i-1}, y_{i-1})进行 I 操作, 得到 x'_i

　对(x'_i, y_{i-1})进行 O 操作, 得到 y'_i

　x'_i 规范化为 x_i

　y'_i 规范化为 y_i

Next i

Return (x_k, y_k).

可以证明[1], 给定一个初始向量 x_0 和 y_0, 迭代过程收敛, 收敛的最终结果 x^* 为 A^TA 的主特征向量, y^* 为 AA^T 的主特征向量。

1.3　PageRank 和 HITS 的比较

PageRank 和 HITS 的迭代算法都利用了特征向量作为理论基础和收敛性依据。这也是超链接环境下的这类算法的一个共同特点。

从两者的权值传播模型来看, PageRank 基于随机冲浪(random surfer)模型将网页权值直接从 authority 网页传递到 authority 网页; 而 HITS 将 authority 网页的权值经过 hub 网页的传递进行传播。

从两者的处理对象来看, 都是针对整个万维网上的网页的一个子集进行排序、筛选, 没有一个搜索引擎能够将万维网上的网页全部搜索下来。但是, PageRank 的处理对象是一个搜索引擎上当前搜索下来的所有网页, 一般在几千万个页面以上; 而 HITS 的处理对象是搜索引擎针对具体查询主题所返回的结果, 从几百个页面扩展到几千几万个页面。PageRank 在 Google 上对 7,500 万个 URL 进行排序的实际运行结果是, 完成一次迭代需要 6min, 收敛后加入悬空链接再迭代, 共耗时 5h。

从两者的具体应用来看, PageRank 应用于搜索引擎服务端, 可以直接用于标题查询并获得较好的结果; 若要用于全文本查询, 需要与其他相似度判定标准(向量模型等)进行复合, 以针对具体查询形成最终排名; 搜索机器人(Crawler)可以将 PageRank 做为搜索优先次序的标准; 算法中 E 的取值可以用来定制个人化搜索引擎。HITS 一般用于全文本搜索引擎的客户端, 对于宽主题的搜索相当有效, 可以用于自动编撰万维网分类目录; 通过找到指向某网页的 Hub 网页并以此为根集 R_σ, 可以查找该网页的相关网页; 也可用于元搜索引擎的网页排序。对于窄主题的检索, HITS 现在的能力还较弱, 因为根

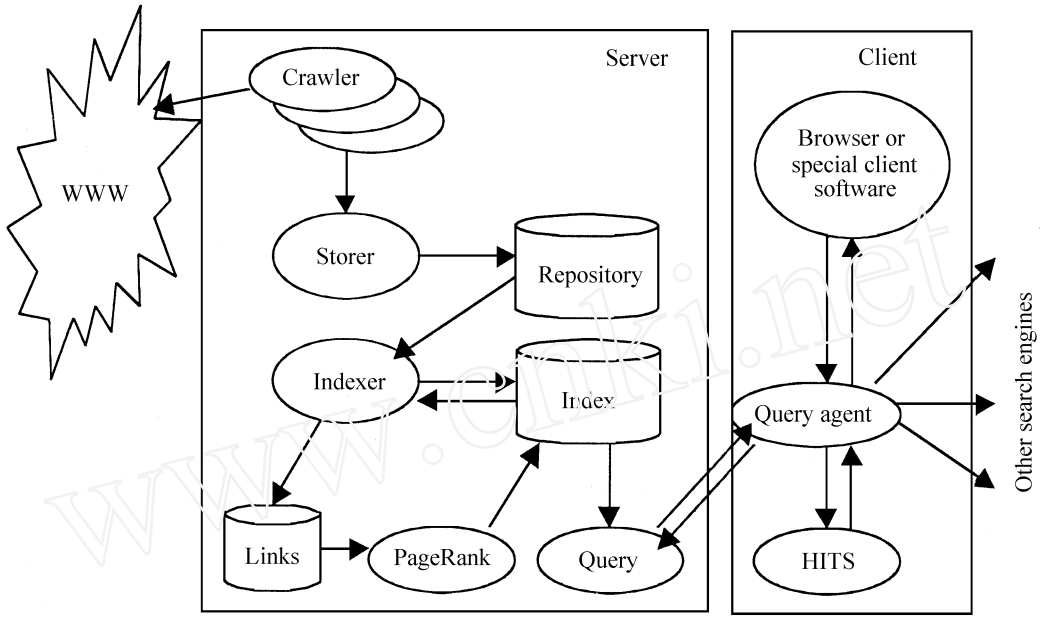

图1 信息综合检索模型

Fig. 1 An integrative model of information retrieval

集太小,筛选的效果将不会很大。

2 信息检索综合模型

PageRank 算法和 HITS 算法是具有代表性的两个网页排序算法,前者更适合于搜索引擎的服务器端,后者更适合于搜索引擎的客户端。我们构造和初步实现了以下的信息检索综合模型,来充分体现超链接环境下的网页排序算法在信息检索中的应用。

模型说明:服务器端,多个搜索机器人(Crawler)在万维网上搜索,存储管理器(Storer)将搜索下来的页面进行简单的预处理存入网页仓库,索引器(Indexer)从网页仓库读取页面,并对其进行索引建立索引库,同时产生一张链接表,其中存储有关链接的信息,PageRank 依据这张链接表计算每个网页的排名值,放入索引库中。客户端,浏览器或专用客户程序接受用户的查询请求,发送给查询代理,查询代理通过自然语言处理、查询扩展、用户定制等技术自动将查询请求转变为更为精确的形式,发送给不同的全文本搜索引擎;搜索引擎服务器端从索引库搜索出结果,将 PageRank 与其他相关度评价标准结合进行排序,将排名高的结果返回客户端;客户端再对这些结果采用基于 HITS 的算法进行排序,同时结合自动聚类技术,将最符合用户要求的网页以分类的形式呈现给用户。

这一模型中,服务器端的网页排序采用了 PageRank 算法,客户端的查询代理采用了 HITS 算法,两端分别进行了初步实现。对于一般查询需求,为了保证查询速度,用户可以直接通过浏览器而不通过查询代理,因为 HITS 算法的运行需要一定的时间。针对某些特殊需要,例如查找文献,查找某个主题的主要网页等,用户有明确的查询需求,可以采用查询代理,运用 HITS 算法,尽管花费了一定的时间,却获得了较好的结果。

对客户端稍加修改,这一模型还可用于自动编纂万维网的分类主题目录。例如,预先提供一个主题目录结构,每个主题预设一些关键词,通过对每个主题分别进行自动查询处理,获取最有价值的网页列表。

超链接环境下基于链接结构的网页排序算法在信息检索中的作用是明显的,具有广泛的应用前景。下一步的工作将是,在服务器端结合其他网页相关度评价模型(如向量模型),客户端结合智能代理技术,自然语言处理技术,聚类技术等,进一步优化结果,满足信息检索的需要。

(下转第465页)

化方法的变结构控制因优化了控制器参数而基本消除了抖振, 常规变结构控制却难以做到。用试凑法求得的参数为:

$$D = [1.954\ 1 \quad 30.594\ 5]$$
$$k_1 = 0.1$$
$$k_2 = 0.9$$

3 结　论

综合优化方法用于焦化塔的双线性建模和变结构控制设计, 提高了动态模型精度, 优化了变结构控制器参数, 保证了控制品质, 消除了抖振, 这是试凑法难以做到的, 为进一步的工业实施打下了基础。但是, 在线实施还需要考虑双线性模型的在线修正问题。

参考文献:

[1] 华向明. 双线性系统建模与控制[M]. 上海: 华东化工学院出版社, 1990.

[2] Utkin V I. Variable structure systems with sliding model [J]. IEEE Tran on Automatic Control, 1977, **22**(2): 201-207.

[3] 高为炳. 变结构控制理论基础[M]. 北京: 中国科学技术出版社, 1990.

[4] Platin B E. An application of variable structure systems with sliding mode to a remotely operated vehicle[J]. 11th IFAC World Congr, 1990, **8**: 105-111.

[5] Sarptunk S Z. on the seebility of discrete-time sliding mode control systems[J]. IEEE Tran on Automatic Control, 1987, **32**(10): 930-932.

[6] 李　兵. 优化新方法及其应用研究[D]. 上海: 华东理工大学, 1996.

(上接第 458 页)

参考文献:

[1] Page L. The PageRank Citation Ranking: Bring Order to the Web[OL]. Stanford Digital Libraries Working Paper, http://www-diglib.stanford.edu, 1999.

[2] Kleinberg J M. Authoritative sources in a hyperlinked environment [C]. [s.l]: Proceedings of the Ninth Annua ACM-SIAM Symposium on Discrete Algorithms, 1998. 668-677.

[3] Katz L. A new status index derived from sociometric analysis [J]. Psychometricka, 1953, **18**: 39-43.

[4] Hubbell C H. An input-output approach to clique identification [J]. Sociometry, 1965, **28**: 377-399.

[5] Garfield E. Citation analysis as a tool in journal evaluation [J]. Science, 1972, **178**: 471-479.

[6] Pinski G, Narin F. Citation influence for journal aggregates of scientific publications: theory, with application to the literature of physics [J]. Inf Proc and Management, 1976, **12**: 297-312.

[7] Geller N. On the citation influence methodology of Pinski and Narin [J]. Inf Proc and Management, 1978, **14**: 93-95.

[8] Doreigan P. A measure of standing for citation networks within a wider environment [J]. Inf Proc and Management, 1994, **30**: 21-31.

[9] Botafogo R, Rivlin E, Shneiderman B. Structural analysis of hypertext: identifying hierarchies and useful metrics [J]. ACM Trans Inf Sys, 1992, **10**: 142-180.

[10] Carrière J, Kazman R. WebQuery: searching and visualizing the Web through connectivity [OL]. http://www.cgl.uwaterloo.ca/Projects/Vanish/webquery-1.html, 1997.

[11] Brin S, Page L. Anatomy of a large-scale hypertextual Web search engine [OL]. Stanford Digital Libraries Working Paper, http://www-diglib.stanford.edu, 1999.