

网络结构挖掘算法研究

陈学进

(安徽工业大学 计算机学院, 安徽 马鞍山 243002)

摘 要:网络结构挖掘是以超链接分析为基础,从链接结构中获取有用的知识,利用这些知识,重新组织结构,使内容逻辑结构更加合理。深入研究现有的网络结构挖掘系统,并在对其核心算法 PageRank 和 HITS 中所存在的问题作了详细分析的基础上提出了自己的改进算法,主要是对每个网页定义这三个参数:PageRank, Authority, Hub, 并进行分析与优化,以便得到更好的查询结果,最后设计了一个改进网络结构挖掘系统原型,根据实验结果进行分析。

关键词:网络结构挖掘;链接分析;PageRank;HITS

中图分类号:TP393

文献标识码:A

文章编号:1673-629X(2009)05-0041-04

Research of Algorithm for Web Structure Mining

CHEN Xue-jin

(Computer College of Anhui University of Technology, Maanshan 243002, China)

Abstract: Web structure mining is based on hyperlink analysis. It has been gained useful information from man-made links structure. Pages can be sorted making use of it. And important content pages can also be found so that can reform web structure to gain better content structure. And go deep into researching algorithm used in existing web structure system. And improves its core algorithm. Mainly analyze and optimize three data of page: PageRank and Authority and Hub so that can gain the best query result. Also design an improvement web structure system prototype with experimental result and data analysis at last.

Key words: Web structure mining; hyperlink analysis; PageRank; HITS

0 引言

网络挖掘指在 WWW 上挖掘潜在的、有用的模式及隐藏的信息过程。根据对网络数据的感兴趣程度不同,网络挖掘一般可以分为三类:网络内容挖掘、网络结构挖掘、网络用法挖掘。其中网络结构挖掘是对网络的链接结构进行分析,以对超链接分析来评估基础网络资源,从而发现有用模式,提高搜索质量。

网络结构挖掘所得到的模式,可以揭示许多蕴涵在网络内容之外的隐含着的有用信息。如通过文档之间的超链接,可以挖掘出文档之间引用关系,从而有助于找到与用户请求相关的权威页面;通过分析 Web 网页内部树形结构,可以发现与给定页面集合相关的其它页面;网络页面的 URL 同样可以反映页面的类型以及页面之间的从属关系,通过分析页面的 URL 信息,可以找到改变了位置的网络页面的新位置。

以上是网络结构挖掘的基本思想,要真正地充分

发挥网络结构挖掘的优越性还有许多细节需要考虑,也还有许多问题需要解决。目前,比较成功的网络结构挖掘系统有采用 PageRank 算法的 Google 搜索引擎和采用 HITS 算法的 Clever 系统,由于它们充分利用页面间的链接结构关系,所以搜索结果明显优于基于相似度的搜索引擎^[1,2]。

1 传统的结构挖掘算法分析

PageRank 算法是网络超链接结构分析中最成功的代表之一,该算法由 Stanford 大学的 Brin 和 Page 提出,是评价网页权威性的一种重要工具。PageRank 的具体定义如下:将网络对应成有向图,设 W 为该有向图结点的集合, $N = |W|$, F_i 是页面 i 指向的所有页面的集合, T_i 是指向页面 i 的所有页面的集合。对每个出度为 0 的结点 S , 设 $FS = \{\text{有向图中全部 } N \text{ 个结点}\}$, 则所有其他结点的 $T_i = \{B \in S\}$, 这样可以将结点 S 所具有的 PageRank 值均匀地传递给其他所有页面。PageRank 的具体迭代公式为:

$$PR(A) = (1 - d) + d(PR(T_1)/C(T_1) + \dots + PR(T_n)/C(T_n))$$

收稿日期:2008-09-11

基金项目:安徽省自然科学基金项目(KJ2007B245)

作者简介:陈学进(1972-),男,安徽六安人,副教授,硕士,研究方向为计算机软件理论及数据挖掘。

其中, $PR(A)$: 页面 A 的网页级别; $PR(T_i)$: 页面 T_i 的网页级别, 页面 T_i 链向页面 A ; $C(T_i)$: 页面 T_i 链出的链接数量; d : 阻尼系数, 取值在 $0-1$ 之间。

PageRank 的实现过程为: 将网页的 URL 对应成唯一的整数, 把每一个超链接用其整数 ID 存放到索引数据库中, 经过预处理(如去除数据库中的悬摆指针)之后, 设每个网页的初始 PR 值为 1, 通过以上的递归算法计算每一个网页的 PageRank 值, 反复进行迭代, 直至结果收敛。

按照威望值可以对网络图中的页面排序, 排序的结果反映了各个页面的重要的程度, 它可以用来辅助传统 IR 技术来产生更加精确的搜索结果: 首先是按照用户提供的关键字搜索, 然后根据威望值排序的结果输出。PageRank 首先应用在 Google 搜索引擎中。从 Google 所产生的实际效果来看, PageRank 确实大大改善了搜索结果的精确度。

Google 是结合文本的方法来实现 PageRank 算法的, 所以只返回包含查询项的网页, 然后根据网页的 rank 值对搜索到的结果进行排序, 把 rank 值最高的网页放置到最前面, 但是如果最重要的网页不在结果网页集中, PageRank 算法就无能为力了。

HITS 首先利用一个传统的文本搜索引擎(例如 AltaVista) 获取一个与主题相关的网页根集合 (root set)。然后向根集合中扩充那些指向根集合中网页的网页和根集合中网页所指向的网页, 这样就获得了一个更大的基础集合 (base set)。假设最终基础集合中包含 N 个网页, 那么对于 HITS 算法来说, 输入数据就是一个 $N \times N$ 的相邻矩阵 A , 其中如果网页 i 存在一个链接到网页 j , 则 $A_{ij} = 1$, 否则 $A_{ij} = 0$ 。

HITS 算法只计算主特征向量, 也就是只能发现 T 集合中的主社区, 忽略了其它重要的社区。事实上, 其它社区可能也非常重要。HITS 算法最大的弱点是处理不好主题漂移问题, 也就是紧密链接 TKC 现象。如果在集合 T 中有少数与查询主题无关的网页, 但是它们是紧密链接的, HITS 算法的结果可能就是这些网页, 因为 HITS 只能发现主社区, 从而偏离了原来的查询主题^[3-6]。

2 结构挖掘算法的改进

一个完整的网络结构挖掘系统的运行机制如下: 自动搜索软件使用高速的分布式爬行器系统中的漫游遍历器定时地遍历网页, 将遍历到的网页送到存储服务器中; 存储服务器使用压缩软件将这些网页进行无损压缩处理后存入数据库 Repository 中, Repository 获得了每个网页的完全 Html 代码后, 对其压缩后的网页

及 URL 进行分析, 记录下网页长度、URL、URL 长度和网页内容, 并赋予每个网页一个文档号 (docID), 以便当系统出现故障的时候, 可以及时完整地进行网页的数据恢复; 索引器从 Repository 中读取数据, 以后做以下工作:

1) 将读取的数据解压缩后进行分析, 它将网页中每个有意义的词进行统计后, 转化为关键词 (wordID) 的若干索引项 (Hits), 生成索引项列表, 该列表包括关键词、关键词的位置、关键词的大小和大小写状态等。索引项列表被存入到数据桶 (Barrels) 中, 并生成以文档号 (docID) 部分排序的顺排档索引。

2) 索引器除了对网页中有意义的词进行分析外, 还分析网页的所有超文本链接, 将其 Anchor Text、URL 指向等关键信息存入到 Anchor 文档库中。

3) URL 分析器 (URL Resolver) 读取 Anchor 文档中的信息然后做将其锚文本 (Anchor Text) 所指向的 URL 转换成网页的 docID; 再将该 docID 与原网页的 docID 形成“链接对”, 存入 Link 数据库中; 最后将 Anchor Text 指向的网页的 docID 与顺排档特殊索引项 Anchor Hits 相连接。

4) 数据库 Link 记录了网页的链接关系, 用来计算网页的 PageRank 值。排序器 (Sorter) 对数据桶 (Barrels) 的顺排档索引重新进行排序, 生成以关键词 (wordID) 为索引的倒排档索引。将生成的倒排档索引与先前由索引器产生的索引词表相连接产生一个新的索引词表供搜索器使用。搜索器的功能是由网页服务器实现的, 根据新产生的索引词表结合上述的文档索引和 Link 数据库计算的网页 PageRank 值来匹配检索。

对该关键词排在前列的页面输出给用户。网络结构挖掘系统在此基础之上充分利用了页面间的链接关系, 并挖掘出这些链接关系的潜在意义^[4,7,8], 网络结构挖掘系统首先要取得根集, 设定根集为 Google 搜索引擎搜索结果的前 200 个页面, 得到根集后, 再将根集扩展成为基本集, 这个扩展过程将重复进行, 在扩展过程中还将对页面进行筛选, 在这一步执行时, 必须先消除超链接信息中的噪音, 在网络有向图的链接中并不是每一个超链接都具有注解性, 有些链接是为其它目的而创建的, 如为了导航或为了付费广告等等。也就是说在网络有向图中那些入度较小的点不可能是权威 (Authority), 那些出度较小的点也不可能是目录型 (Hub)。因此, 这些点的所有导出链接和导入链接在求权威和目录型时都是无用的, 将它们称为超链接信息中的噪音。在执行算法前应该把它们从 G 中删除掉 [G 被清理]。清理结果中会有新节点有较低的出度或入度, 这一过程重复进行, 直到达到一种稳定的状

态。为此可以按照下列算法执行^[4,7]:

首先,设定网络有向图中顶点的入度(indegree)和出度(outdegree)的最小值分别为 X_i 和 X_o ;然后重复做下列动作:

1) 对于任意网络有向图中顶点 V ,若 V 的入度 $< X_i$,则删去 V 的所有的导入边。

2) 对于任意网络有向图中顶点 V ,若 V 的出度 $< X_o$,则删去 V 的所有的导出边。

直到达到一种稳定的状态。这种算法实现其实比较简单,只需要利用图的遍历算法即可。

基本集扩展完成后,对基本集的页面间的相互链接关系充分发掘,形成链接信息库,该链接信息库包括基本集页面间的相互链接关系、链接文本信息等,得到链接信息库后再对每一链接进行综合权重计算,在计算超链权值 W 时,需要将链接文本中的语义信息进行量化,这样才能使链接文本中的语义信息具有可计算性,这个 W 本文使用的是与查询关键字 Q 在链接文本中出现的次数 $n(q)$ 和查询关键字 Q 在链接文本附近出现的次数 $m(q)$ 有关,这个 W 随查询关键字 Q 在链接文本中出现的次数增多而增大。实现时可按下面公式计算:

$$W = 1 + n(q) + m(q) \quad (1)$$

其中(1)式中的 $m(q)$ 在计算时应该考虑链接的类别。显然这个 W 在迭代过程中会不断增大,因此,为了把结果控制在一定范围内,可以在每次迭代后标准化。计算出来的综合权重保存在链接信息库中,形成含权重的链接信息库。

假设最终基础集合中包含 N 个网页,那么对于改进 HITS 算法来说,输入数据就是一个 $N \times N$ 的相邻矩阵 A ,所有链接权值 W 计算完成后,现在就可以改进传统 HITS 算法中求解 Authority 网页和 HUB 网页中迭代公式中矩阵:

若网页 P 链向 Q 则迭代矩阵 $A(p, q) = W$,否则为 0,其它不变,算法为每个网页 i 分配两个度量值:中心度 h_i 和权威度 a_i 。设向量 $a = (a_1, a_2, \dots, a_N)$ 代表所有基础集合中网页的权威度,而向量 $h = (h_1, h_2, \dots, h_N)$ 则代表所有的中心度。最初,将这两个向量均置为 $u = (1, 1, \dots, 1)$ 。操作 $In(a)$ 使向量 $a = A^T h$,而操作 $Out(h)$ 使向量 $h = Aa$ 。经过若干次迭代后,以保证其数值不会使计算溢出,根据数学知识结果向量将收敛于矩阵 $A^T A$ 的特征向量,也就是得到一组 X, Y 值最大的 Authority 网页和 HUB 网页。通过以上过程可以看出,基础集合中网页的中心度和权威度从根本上是由基础集合中的链接关系所决定的,更具体地说,是由矩阵 $A^T A$ 和 AA^T 所决定。

通过以上的改进,在单纯的页面结构链接关系的基础上,加入了页面在内容上的关联,使得算法最后的结果更能令人满意。对于 HITS 算法而言,还有其它的方式可以进一步改进它的精度,比如网络中的页面通常都不是关于一个主题的等等。具体实现则作为文中的进一步研究工作。

为此建立一个原型系统来验证所提出的解决方案的有效性,为改进算法提供实践证明。该系统并不需要独立完成搜索工作,只需和现有搜索引擎建立接口进行资源共享。也就是说根集的获取和链入链出页面的获取将利用现有搜索引擎的功能,再按设计方法计算页面权威权重和 Hub 权重。这个平台的实现具体需要以下几个模块:查询模块、数据接收模块、URL 提取模块、页面遍历模块(附带计算上文中 M)、PR 值计算模块、结果处理模块(显示、存储)。

3 实验结果分析

本系统部分功能的实现是在 WINDOWS XP 环境下,用可视开发工具 Visual C++ 6.0 作为主要的界面和功能模块编程工具。其特点是使用灵活并且方便,提供了很多功能强大的函数和系统调用。

由于结果处理数据一般为几千个,所以使用小型数据库 Access2002,其特点是操作方便、界面友好,是处理小规模数据常用数据库。

搜索引擎本系统使用的是 PageRank 处理较好的 Google 系统。表 1 是以“IT 公司”为搜索关键字返回结果,其中最右边一列为其网页的 PageRank 值,显然系统返回结果中 90% 以上 PageRank 值都在 5 以上,而百度输出结果只有 20% 的在 5 以上。但似乎更接近关键字,这是因为这仅是本系统在基本集选取中进行的实验的结果,作为基本集的选取要远远高于百度输出结果,当然这中间有些是与搜索关键字无关的网站,但大多数都是很好的 HUB 网站如 www.microsoft.com/china/ 等等,根据本系统的后面算法显然能得到较好的结果。表 2 给出几个不同的搜索关键字进行实验分析如图 1 所示。

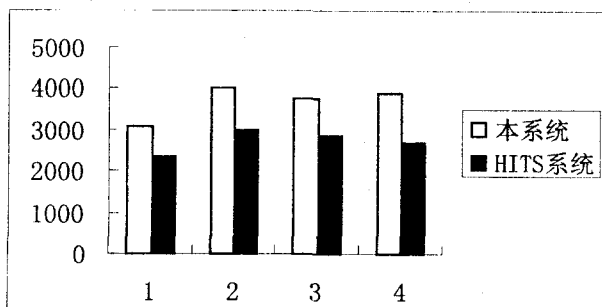


图 1 本系统与 HITS 系统对比图

表 1 返回结果

本系统部分完成输出结果	1	中国电信	www.chinatelecom.com.cn	7
	2	中国海洋石油公司	www.cncooc.com.cn	6
	3	网易	www.163.com	7
	4	微软中国有限公司	www.microsoft.com/china/	7
	5	海尔集团	www.haier.com	5
	6	金山在线	www.kingsoft.com	6
	7	瑞星反病毒查询网	www.rising.com.cn	6
	8	中国上市公司查询网	www.cnlist.com	4
	9	中国移动公司	www.chinamobile.com	7
	10	中国国际股份有限公司	www.airchina.com.cn	6
百度输出结果	1	IT 公司查询网	www.seeitoo.com	4
	2	全国 IT 公司速查手册	www.iteer.net/modules/xdirectory/	5
	3	中国 IT 公司猎头公司	www.hrm.net.cn	5
	4	互联网实验室	www.chinalabs.com/lisi	2
	5	IT 公司速查手册	www.newwww.com	2
	6	IT 世界网	It.com.cn	5
	7	IT 公司	www.chinassb.com/nv/nv2501	0
	8	翻译公司	www.translation-company.net.cn/gzff.htm	0
	9	3691 网	www.3691.com/computer/105.htm	0
	10	IT 公司	www.texindex.com.cn/company/netail	0

表 2 搜索关键字列表

搜索关键字	IT 公司	数据挖掘	MFC 基类	CAR
编号	1	2	3	4

实验中关键技术主要使用了 VC++ 6.0 中几个 MFC 类: HTTP 类、ChttpConnection 类、ChttpFile 类等。HTTP 类是在查询模块中应用, 利用 ChttpConnection 类中的方法 OpenRequest 发送一个 GET 命令, 此时 ChttpFile 类被创建, 然后利用 ChttpFile 类的几个成员函数就可以和搜索引擎响应了。为了获得 Url 信息,

可以利用 CString 类的成员函数 find 读取文件中的字符串。最后就可生成网络图, 主要包括两张表一是节点表, 另一是有向边表。

4 结束语

在分析网络结构的基础上, 介绍了网络结构挖掘中 HITS 算法和 PageRank 算法模型, 并针对其弱点提出了改进方案。主要是对每个网页定义这三个参数: PageRank、Authority、Hub, 并进行分析与优化, 为此建立一个原型系统来验证所提出的解决方案的有效性, 为改进算法提供实践证明, 在今后的工作中, 可以考虑根据这些思想进一步改进算法。

参考文献:

- [1] Kosala R, Blockeel H. Web Mining Research: A Survey[J]. SIGKDD Explorations, 2000, 2(1): 1-15.
- [2] 杨 鲲, 孟 波. 一种基于 XML 的 Web 数据挖掘方法[J]. 计算机应用, 2003, 23(6): 160-164.
- [3] 陈 莉, 焦李成. Internet/Web 数据挖掘研究现状及最新进展[J]. 西安电子科技大学学报, 2001, 28(1): 114-119.
- [4] 邓 英, 李 明. Web 数据挖掘技术及工具研究[J]. 计算机工程与应用, 2001(20): 92-94.
- [5] 冉 丽, 何毅舟, 许龙飞. 基于 Web 结构挖掘的搜索引擎作弊检测方法[J]. 计算机应用, 2004(10): 160-162.
- [6] 宋建康, 张礼平. Web 结构挖掘算法探讨[J]. 华东理工大学学报, 2003(10): 537-540.
- [7] 刘丽珍. 网络结构挖掘的关键分析[J]. 计算机应用研究, 2003(5): 116-118.
- [8] Spertus E. Parasite: Mining Structural Information on the Web[C]// The Sixth International WWW Conference. Santa Clara, USA: [s. n.], 1997.

(上接第 40 页)

- [4] Tan Pang-Ning, Steinbach M, Kumar V. 数据挖掘导论[M]. 范 明, 等译. 北京: 人民邮电出版社, 2006.
- [5] Inokuchi A, Washio T, Okada T, et al. Applying the apriori-based graph mining method to mutagenesis data analysis[J]. Comput Aided Chem, 2001(2): 87-92.
- [6] Kuramochi M, Karypis G. Frequent subgraph discovery[C]// In: IEEE Intl Conf Data Mining. California, USA: [s. n.], 2001: 313-320.
- [7] Yan X, Han J. gSpan: Graph-based substructure pattern mining[C]// In: IEEE Intl. Conf. Data Mining (ICDM'02). Maebashi City, Japan: [s. n.], 2002: 721-724.
- [8] 王艳辉, 吴 斌, 王 柏. 频繁子图挖掘算法综述[J]. 计算机科学, 2005, 32(10): 193-196.
- [9] 李玉华, 罗汉果, 孙小林. 一种基于 Apriori 思想的频繁子图

- 发现算法[J]. 计算机工程与科学, 2007, 29(4): 84-87.
- [10] 唐德权, 夏幼明, 张丽英. 基于图的数据挖掘算法研究[J]. 云南师范大学学报, 2007, 27(5): 30-34.
- [11] Koyutürk M, Grama A, Szpankowski W. An efficient algorithm for detecting frequent subgraphs in biological networks[J]. Bioinformatics, 2004, 20(1): 200-207.
- [12] 李先通, 李建中, 高 宏. 一种高效频繁子图挖掘算法[J]. 软件学报, 2007, 18(10): 2469-2480.
- [13] Han J W, Pei J, Yin Y W. Mining frequent patterns without candidate generation[C]// In: ACM-SIGMOD. Dallas, TX, New York: ACM Press, 2000: 1-12.
- [14] Krishnamurthy L, Nadeau J, Özsoyoglu G, et al. Pathways database system: an integrated system for biological pathways[J]. Bioinformatics, 2003, 19(8): 930-937.