第16卷 第7期
2006年7月

计算机技术与发展
COMPUTER TECHNOLOGY AND DEVELOPMENT

Vol.16 No.7
Jul. 2006

Google核心——PageRank算法探讨

冯振明
(河海大学 计算机及信息工程学院,江苏 南京 210098)

摘 要:搜索引擎技术的发展是随着电子技术不断进步而形成的信息数字化和数据网络化的必然产物。一个出色的搜索引擎能够及时向用户提供所需要的信息,而要做到这点就需要一个快速、优质、高效的搜索算法予以支持。Google搜索引擎依靠其PageRank机制及收敛算法一直处于该领域的领先地位。文中介绍了这个搜索引擎的核心:PageRank算法。PageRank算法通过计算网页的重要性值——PageRank值来确定网页排序的优先级,而网页的PageRank值则是通过累加指向该网页的其他网页的PageRank值得到的。因此Google的搜索结果是高效的、客观正确的。

关键词:PageRank;网络图;PageRank特征向量;收敛算法
中图分类号:TP301.6 **文献标识码**:A **文章编号**:1673-629X(2006)07-0082-03

Google's Core: Discussion about PageRank Algorithm

FENG Zhen-ming
(College of Computers & Information Engineering, Hohai University, Nanjing 210098, China)

Abstract: Search engine technology is coming out while the development of electronic technology which make the information become numeral and data become webside. A famous search engine has to provide the information which user need immediately. If it can do so, a quick, excellent, usable search algorithm is needed. Google search engine keep on top by its PageRank and convergence algorithm. We will make a general introduction about its core: PageRank algorithm. PageRank algorithm makes the preference of the web by its PageRank value and web's PageRank value is accumulated by another webs' PageRank value which are pointing to it. So, Google's search result is effective, impersonal and realistic.

Key words: PageRank; web graph; PageRank eigenvector; convergence algorithm

0 引 言

随着计算机技术的普及和Internet网络技术的发展,信息数字化和数据网络化已成为现代经济社会发展的客观要求和必然趋势。当今世界上已经拥有超过10亿的Internet用户和近百万个不同级别的网络服务器。同时,包括政治、经济、科学、文化、艺术等各个不同的社会领域也都不同程度地实现了其资源信息的数字化和共享化。Internet网络已名副其实地成为世界最大的信息中心。

作为一个Internet用户,自然希望能够最大程度地使用如此庞大而全面的信息资源,但是Internet网又是分散的,单靠浏览一个或几个网页所能得到的信息对于整个网络中的可用信息而言可谓是沧海一粟,浅薄得很。用户自然希望能够得到更多、更全面的信息,在这种需求下网络搜索引擎技术应运而生。只要用户键入关键字,搜索引擎就能从网络上找出与关键字相匹配的信息返回给用户。目前,国际和国内使用较多的搜索引擎有Google、百度、新浪、搜狐等等。其使用频率已达每分钟上万次。

收稿日期:2005-10-10

作者简介:冯振明(1981-),男,江苏苏州人,硕士研究生,研究方向为数据挖掘;导师:王志坚,教授,博士生导师,研究方向为软件自动化、面向对象和构件技术等。

1 搜索引擎的技术要求

但是,并不是只要进行简单的字符匹配就能满足Internet用户的搜索需要的。

首先,用户所给的关键字未必语义清晰;用户可能输入不同的文字,或者是多个单词或词组的无序组合等等,这就要求搜索引擎拥有语意识别及字符串的模糊查询等功能。

其次,网络中的数据格式并不单一,网上的数据包含文本、多媒体、图片、压缩包等。单靠关键字匹配,未必能把这些不同格式的数据都找出来,因此搜索引擎也需要拥有数据类型转换或提供相关链接的功能。

再次,也是用户最关心的内容:搜索引擎的效率如何?搜索引擎的效率包括搜索时间是否最短,搜索的内容是否全面,搜索的内容是否正确,所搜出的内容的排序是否合理(正确性高、内容全面的排在最前)等。

其中,提高搜索引擎效率的方法是搜索引擎的核心技术所在,也是代表该搜索引擎技术水平的主要指标。各个

不同的搜索引擎都有其自己的算法和数据结构。下面将以 Google 搜索引擎的 PageRank 算法加以介绍。

2 PageRank 算法的准备工作

2.1 网络图模型思想

对于海量的无规则的网上数据页面,要从其中得到尽可能多的有用信息是很困难的,因此,需要对它们进行组织,从而建立起一个便于进行处理的网络图。

2.2 网络图模型的建立

对于 Internet 用户,当他们要查询某条信息时,往往会先浏览那些访问次数较为频繁的网页,从中找出与目的信息相关的链接。基于这种现象,可以建立起以某个主题为中心的"地区密集型"子图。其过程为:

(1)收集含有相关主题的离散型网页。

(2)当这些网页的数量到达某个临界值时,用某种算法以一定的图形式的数据结构组织起来,所形成的便是"地区密集型"子图。

(3)在多个"地区密集型"子图之间再通过某些链接连接起来。这些链接可以是本来就存在的,也可以是建模者另行建立的。

2.3 网络图模型的技术要求

网络模型图的结构就是将相互关联的网页结点连接起来构成密集型子图,然后再以这些子图为结点构成更高一级的子图[1]。在实际应用中,根据结点数量和使用目的的不同可以扩展为更多层次的结构,这样也可便于逐层进行查询。网络图中同一层次的结点间应满足高聚合的原则,而不同层次之间应满足低耦合的原则。至于构建多层网络图的具体方法及规范可以参阅与空间数据库相关的资料,文中不再加以详述。

3 PageRank 算法的思想及数学实现

3.1 算法思想

在一个网络图中,从结点 u 到结点 v 的一条有向边(即从网页 u 到网页 v 的一个链接)可以看成结点 v 对于结点 u 而言是一个"重要"结点,而网络图中任意一个结点有多"重要"便可参看指向该结点的有向边的数量有多少。同时若指向 v 的结点 u 本身便有许多指向它的有向边,即结点 u 就是一个很"重要的"结点,那么可以认为结点 v 也是"重要的"[2]。由此,可以建立一个由两两结点间的相互关系所组成的 n 维矩阵,再通过数学方法来进行数据的处理。

3.2 数学实现

设 $u \to v$ 表示在图 G 中,存在一条从 u 到 v 的有向边,$\deg(u)$ 表示图 G 中 u 的出度。在一个随机的时间内从 u 出发指向另一个结点,那么正好指向 v 的概率为 $1/\deg(u)$。因此图 G 中所有两两结点 i,j 间的连接相对于出发点 i 的概率为 $P_{ij} = 1/\deg(i)$。由此定义图 G 的随机过渡矩阵 $\boldsymbol{P}$ 定义为 $P_{ij} = 1/\deg(i)$,表示图 G 的所有连接变换。若 i 到 j 没有连接,则 $P_{ij} = 0$。在图 G 中可能会存在没有出度的结点,即该点所在的一行都为 0。这样的杂点不利于之后的处理,故需要排除。因此可以通过以下变换得到矩阵 $\boldsymbol{P}'$:

$$d_i = \begin{cases} 1 & \text{if } \deg(i) = 0 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

设 $\boldsymbol{v}$ 为 n 维列向量,它是所有结点的概率值,n 为结点的个数。$\boldsymbol{v} = [1/n]_{n+1}$,设 $\boldsymbol{d}$ 为结点有无出度的 n 维列向量,则 $\boldsymbol{D} = \boldsymbol{d} * \boldsymbol{v}$。变换矩阵 $\boldsymbol{P}' = \boldsymbol{P} + \boldsymbol{D}$。$\boldsymbol{D}$ 的作用是转移概率值,表示冲浪者在浏览一个没有出度的网页时,在下一个时段按照 v 所给的分布状态随机地跳到下一个页面。

根据马尔可夫链的各态历经性定理,当 $\boldsymbol{P}'$ 是非周期和不可回归的情况下其马尔可夫链有惟一固定的值。因此如果图 G 是强连接的,那就不能正确地表示网络图的即时结构(网络图通常是动态变化的)。为了解决这个问题,为每个结点再加上一个数值较小的概率值,形成一个完全变换图。把转换过后的矩阵定义为 $\boldsymbol{P}''$:

$$E = [1]_{n\times 1} \times \boldsymbol{v}^{\mathrm{T}}$$

$$\boldsymbol{P}'' = c\boldsymbol{P}' + (1-c)E \tag{2}$$

其中 c 为某个概率值。E 的作用是,在任一时间,一个冲浪者访问某些结点时,以概率 $1-c$ 跳到一个随机页面。所随机条至的页面的选择同样取决于 $\boldsymbol{v}$ 中所给出的概率分配。

最后是对转置矩阵 $\boldsymbol{A} = (\boldsymbol{P}'')^{\mathrm{T}}$ 进行进一步计算,利用矩阵相乘式:$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$。值得注意的是,从 $\boldsymbol{D}$ 和 E 中人为地引入的边并不需要具体化。因此引入 $\boldsymbol{D}$ 和 E 对于矩阵在计算过程中的使用并不会造成太大影响(其影响可忽略不计)。

设冲浪者在时间 0 时他的位置概率向量为 $\boldsymbol{x}^{(0)}$,那么在时间 k 时向量值则为 $\boldsymbol{x}^{(k)} = \boldsymbol{A}^k\boldsymbol{x}^{(0)}$。当 $k \to \infty$ 时其置为:$\lim_{k\to\infty}\boldsymbol{x}^{(k)}$,亦即:$\lim_{k\to\infty}\boldsymbol{A}^k\boldsymbol{x}^{(0)}$。从式中可以看出最终的向量值与初始向量相关。这个最终向量是矩阵 $\boldsymbol{A} = (\boldsymbol{P}'')^{\mathrm{T}}$ 的主要特征向量,也是所要求得的 PageRank 向量。通过此向量,可以找到所需要的结点。

在计算机实现的过程中,由于机器运算的特性,收敛的过程使用递归运算更为有效,因此使用递规运算式 $\boldsymbol{x}^{(k)} = \boldsymbol{A}\boldsymbol{x}^{(k-1)}$。运算过程如下(算法 1)[3]:

```
function x^(n) = PowerMethod(){
x^(0) = v;
k = 1;
repeat
    x^(k) = A x^(k-1)
    δ = || x^(k) - x^(k-1) ||_1;
k = k + 1;
until δ < ε;
}
```

其中 ε 为设定的阈值,当 δ 小于 ε 时,便认为已达到收敛了。

4 当前的研究重点

虽然可以使用算法(1)来求解 PageRank 向量,但是仍然无法满足实际需要:一方面,网络数据库中的数据正在以几何级数增长,数据容量已达到太(TB)级以上;另一方面,在时间就是金钱的观念下,人们不希望花更多的时间来等待搜索结果。因此,能否优化算法,使其能够加速收敛成为当前的主要议题。一种方法是对矩阵 A 进行压缩,以减少递归运算的计算量。但是,数据压缩本身需占用大量的系统时间和资源,而且压缩过程极易产生数据失真或数据丢失,从而影响计算结果的正确性。因此,人们大多选择改进收敛算法。目前,已有不少人提出了自己的收敛算法。

这些算法虽然侧重点不同,但是都在一定程度上实现了加速收敛的效果。如:

* 斯坦福大学 Sepandar D. Kamvar, Taher H. Haveliwala 提出了艾特肯外推法和二次外推法[1]。

* 美国布鲁克林工艺大学 CIS 部 YenYu Chen, Qingqing Gan, Torsten Suel 提出了基于 I/O 技术的 PageRank 加速算法[4]。

* 斯坦福大学 Sepandar D. Kamvar, Taher H. Haveliwala, Gene Golub 提出了 PageRank 的适应性算法[5]。

5 展 望

对于 PageRank 的加速算法虽然有不少学者提出自己的新观点或改进了现有算法,但在计算过程中大多仍然是先获取矩阵 A 的值,然后再脱机计算 PageRank 向量。也就是说运算的过程是一个静态过程。但是,网上的数据是不断动态变化的,很有可能在获取矩阵 A 的值过后,网上出现了新的符合要求的数据。如此便造成了信息丢失,对于某个 Internet 用户来说甚至可能是失去了一个商机。因此,在继续优化加速算法的同时,考虑设计适合获取实时信息的动态算法将是未来研究 PageRank 算法的主要趋向。

参考文献:

[1] Kamvar S D, Haveliwala T H. Extrapolation Methods for Accelerating PageRank computations[Z]. CA, USA: Stanford University, 2003.

[2] Ridings C, Shishigin M, Whalen J. PageRank Uncovered[Z]. [s.l.]: [s.n.], 2002.

[3] Arasu A. Pagerank computation and the structure of the web - Experiments and algorithms[Z]. CA, USA: Computer Dept. of Stanford Univ., 2002.

[4] Chen Yenyu, Gan Qingqing, Suel T. IO - efficient Techniques for Computing Pagerank[Z]. 美国: 布鲁克林工艺大学, 2002.

[5] Kamvar S D, Haveliwala T H, Golub G. Adaptive Methods for the Computation of PageRank[Z]. CA, USA: Stanford University, 2003.

(上接第 81 页)

```
  private String description;
  private int inputsNumber;
  private int outputsNumber;
  public MapFunction(String fileName)//从映射函数的 XML 文档生成映射函数抽象类的构造函数
  public abstract Object do();//抽象的映射函数运行方法,在子类中具体实现
  …//get 与 set 方法;
}
```

具体的映射函数类(MapFunction),如加减乘除、数据库操作等,可以通过实现(extends)映射函数抽象类得到。

映射函数工厂类:

```
public class mapFunctionFactory
{
  public static MapFunction creator(Link link) //根据当前映射链生成相应的映射函数
  …
}
```

3 结束语

基于 Web 服务,以 XML 为公用格式的数据交换方案,能够解决数据模式和数据语义上的异构性,通过互联网提供跨系统平台的异构数据库交换。其层次化的交换框架具有松耦合性和高可集成性的特点,客户可以动态地发现服务,加入数据交换体系,同时服务和客户可以保持各自的灵活性。模型化的数据转换过程从实现层面上保证了交换方案的通用性。通过这一方案,网络数据资源可以得到简单而高效的交互和融合,有望在整个互联网内实现应用系统间的数据协作。

参考文献:

[1] Bray T, Paoli J, Sperberg - Mcqueen C M, et al. Extensible Markup Language (XML) 1.0[EB/OL]. http://www.w3.org/TR/REC-XML, 2004.

[2] 方 翔, 李伟生. 关系模式到 XML 模式的影射[J]. 计算机应用研究, 2002(1): 130 - 132.

[3] 彭 竑, 赵曦滨, 雍建平, 等. 基于消息机制的异构系统集成方案[J]. 计算机应用研究, 2005(8): 43 - 46.

[4] 柴晓路. 柴晓路专栏/架构 Web Services 及柴晓路专栏/SOAP 应用模式[EB/OL]. http://www-128.ibm.com/developerworks/cn/xml/theme/indexcxl.html, 2003.

[5] 杨 丹, 周 刚. 基于 UDDI 服务订阅的 Web 服务推荐机制[J]. 华中科技大学学报(自然科学版), 2003, 31(S): 362 - 364.