语义桌面环境下的一种索引排序方法*)

李 胜[1] 张新明[2] 胡和平[1] 卢正鼎[1]
(华中科技大学计算机科学与技术学院 武汉 430074)[1]
(河南师范大学计算机技术与科学学院 新乡 453007)[2]

摘 要 个人计算机技术的不断发展使得传统的桌面检索和排序方式越来越不能满足用户的需求。本文给出了一种语义桌面环境下的桌面资源索引排序模型,并在PageRank和数据库的权威度传递图理论基础上提出了一种索引排序算法。与传统的检索排序方式相比,本方法能更好地反映查询与检索结果之间的相关程度以及结果的重要程度次序。实验表明,本方法在时间效率和空间占用方面均能适应目前普通个人计算机的处理能力。本方法在语义桌面以及桌面搜索等相关领域有着广泛的应用前景。

关键词 语义桌面,桌面搜索,权威度传递图

A Method for Searching and Ranking in the Semantic Desktop Environment

LI Sheng[1] ZHANG Xin-Min[2] HU He-Ping[1] LU Zheng-Ding[1]
(School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074)[1]
(School of Computer Science and Technology, Henan Normal University, Xinxiang 453007)[2]

Abstract With the development of computer science and the technology of PC, the traditional desktop searching and ranking method could not satisfy the requirement of user. This paper provides a ranking model for the desktop resource searching based on metadata in the environment of semantic desktop, then after the study of PageRank algorithm and the theory of authority transfer graph, a new index ranking algorithm is provided. The experiment proves that this ranking method could effectively reflect the priority of the researching results and the relationship between the query and the results. The time and space efficiency of this method could also meet the capability of today's normal computers. This method could work for semantic Web, desktop searching and other correlative domains.

Keywords Semantic desktop, Desktop searching, Authority transfer graph

1 引言

随着PC技术的不断普及和发展,以及硬盘存储能力的大幅度提高,越来越多的文件被存储到个人计算机当中。然而,用户对这些文件的管理却越来越困难。人们往往难以准确地找到那些自己需要的文件,而这些文件往往是自己曾经使用过或访问过的。相反,Internet上的检索技术却是突飞猛进,我们在Web上搜索文档往往比在自己的PC机上找文档容易得多,这得益于搜索引擎技术和相关排序算法(如Google PageRank算法)的使用和推广。

目前的桌面检索工具,一般都采用建立文件索引的方式来增强桌面的检索效率。然而,即使使用了这样的工具,在个人计算机(相对小的集合)上搜索文档仍然比不上在Web(较大的集合)上搜索文档方便,其根本的原因在于:目前的桌面搜索工具既无法使用PageRank这样的排序机制,也没有充分利用个人计算机的特性,特别是上下文信息。语义桌面技术能够很好地解决上下文信息提取问题。然而现有的语义桌面系统(如Beagle语义桌面)所使用的搜索工具,仍然使用传统的基于关键字的搜索和排序方式。因此,需要把Web上的排序技术运用到语义桌面搜索中,才能有效地改进传统的桌面数据索引方法,提高桌面搜索的质量。

本文通过研究PageRank算法和数据库中的权威度传递理论,在它们的基础上,提出了一种在语义桌面环境下的资源索引排序方法。此法可以大幅度提高桌面搜索工具的性能,使得语义桌面搜索更加贴近Web搜索引擎的功能。

2 PageRank算法和权威度传递图

本节描述关于PageRank和基于权威度传递图的基本原理。

2.1 PageRank算法

假设在图(V,E)中,$V=\{v_1,\cdots,v_n\}$代表图中全部节点的集合,E是全部边的集合。一个Web访问者随机地从节点(网页)v_i开始访问,每当访问一个节点时,都以概率d沿某个超链接去访问下一个页面,或者以概率$1-d$随机地转到其它不相关的网页上去,则v_i的PageRank值是用户在访问v_i这个点的时候的概率$r(v_i)$。如果我们用r来表示向量$[r(v_1),\cdots,r(v_i),\cdots,r(v_n)]^T$,此时有

$$r=dAr+\frac{(1-d)}{V}e \tag{1}$$

其中,A是一个不可约和非周期$m\times n$的矩阵:若存在一条边

*)河南省自然科学基金"粒逻辑语义推理和语法推理研究"(No.0611055200)。李 胜 博士研究生,研究方向为语义Web、信息安全;张新明 副教授,研究方向为人工智能、数字图像处理;胡和平 教授,副博士生导师,研究方向为软件工程、智能决策支持系统;卢正鼎 教授,博士生导师,研究方向为计算机辅助软件工程、智能信息系统。

$v_j \to v_i$，则 $A_{i,j}=\frac{1}{OutDeg(v_j)}$；否则 $A_{i,j}=0$。$OutDeg(v_j)$是节点 v_j 的出度，同时有 $e=[1,\cdots,1]^T$。

以上的 PageRank 公式是一种典型的预处理方式。在查询之前，服务器提供一个全局的与关键字无关的页面排序。在该算法中，用户不需要使用全体节点的集合 V 作为基础集合，而只需要使用任意一个节点子集 S。这样做，可以增加那些与 S 相邻节点的权威度。具体来说，我们定义了一个向量 $s=[s_0,\cdots,s_i,\cdots,s_n]^T$。其中，如果 $v_i \in s$，则 $s_i=1$，否则 $s_i=0$。最后的 PageRank 公式是

$$r=dAr+\frac{(1-d)}{|S|}s \tag{2}$$

反复使用以上公式，PageRank 算法可以计算出全部的 r 值。其中第$(k+1)$个 r 的值可以由以下公式计算得到：

$$r^{(k+1)}=dAr^{(k)}+\frac{(1-d)}{|S|}s \tag{3}$$

当 r 聚合于一点的时候，算法停止。基础集 S 的概念在文[10]中做了介绍，它是实现个性化排序的一种方法，我们通过设置 S 来为用户设置书签。

2.2 权威传递图

把数据库看作是一个标记图，这样更易于我们从中获取关系和 XML 数据。图 $D(V_D,E_D)$是一个标记有向图，图中每个节点 v 都拥有一个标记 $\lambda(v)$和一组关键字。每个节点表示数据库中的一个对象，可能有自己的子结构。为了不失一般性，本文假定每个节点拥有一对属性名和属性值组合。其中，属性值中的关键字包含与节点相关的关键字集合。通过加入元数据，可以描述关键字的语义。

用任务 $\lambda(e)$来标记从节点 u 指向节点 v 的边 e，用以表示 u 和 v 之间的关系。当任务很清晰且不会产生歧义时，可以省略标记，用"$u \to v$"来表示从节点 u 指向节点 v 的边 e。为简化问题，本文假定图中不存在两条完全相同的边。

模式图 $G(V_G,E_G)$是一种有向图（如图 1 所示，我们以 DBLP 文献数据库为例），用来描述 D 的结构。每个节点都有一个与之相关联的标记；每条边用一个任务来标记，有时标记可以忽略。一个数据图 $D(V_D,E_D)$适应于一个模式图 $G(V_G,E_G)$，如果存在一个唯一指定的 μ，满足：

(1)对于每一个节点 $v \in V_D$，存在一个节点 $\mu(v) \in V_G$，使得 $\lambda(v)=\lambda(\mu(v))$；

2. 对于每一条从节点 u 到节点 v 的边 $e \in E_D$，存在一条边 $\mu(e) \in E_G$，由 $\mu(u)$指向 $\mu(v)$，使得 $\lambda(e)=\lambda(\mu(e))$。

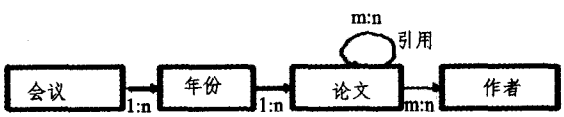

图 1 DBLP 数据库的模式图

2.2.1 权威度传递模式图

通过模式图 $G(V_G,E_G)$可以创建权威度传递模式图 $G^A(V_G,E^A)$，该图可以反映通过边传递权威度的情况。对于 E_G 中的每一个边 $e_G=(u \to v)$，可以创建两条权威度传递边 $e_G^f=(u \to v)$和 $e_G^b=(u \to v)$，这两条边携带模式图中边的标记。此外，每条边还有一个相应的权威度传递率 $\alpha(e_G^f)$和 $\alpha(e_G^b)$注释。我们说一个数据图适应于一个权威度传递模式图，如果它适应于相应的模式图。

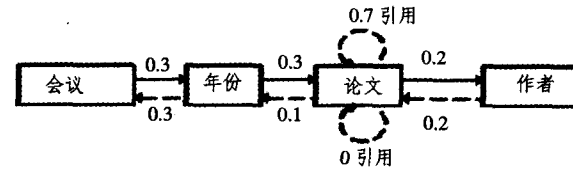

图 2 DBLP 数据库的权威度传递模式图

图 2 是由图 1 所示的模式图生成的权威度传递模式图。之所以定义双向边，主要是因为权威度双向传递，而且，两个方向的权威度传递率是不同的。例如：引用别人的文章和被别人的文章引用，其权威度传递率就截然不同。注意，如果从某节点出发边的权威度传递率之和小于 1，说明该接点传递出的权威度很小。

2.2.2 权威度传递数据图

给定一个数据图 $G(V_G,E_G)$适应于权威度传递模式图 $G^A(V_G,E^A)$，对于每一条边 $e=(u \to v) \in E_D$，权威度传递数据图都有两条与之对应的边 $e^f=(u \to v)$和 $e^b=(u \to v)$。边 e^f 和 e^b 分别用权威度传递率 $\alpha(e^f)$和 $\alpha(e^b)$加以注释。假设 e^f 的类型是 e_G^f，则有

$$\alpha(e^f)=\begin{cases}\dfrac{\alpha(e_G^f)}{OutDeg(u,e_G^f)}, & \text{if } OutDeg(u,e_G^f)>0\\ 0, & \text{if } OutDeg(u,e_G^f)=0\end{cases} \tag{4}$$

其中 $OutDeg(u,e_G^f)$是节点 u 的出度，类型是 e_G^f。权威度传递率 $\alpha(e^b)$的定义和 $\alpha(e^f)$类似。

2.3 权威传递图中的排序权值

给出一个关键字查询 w，用以下公式为每一个节点 $v_i \in V_D$ 设置一个对象排序权值：

$$r^w=dAr^w+\frac{(1-d)}{|S(w)|}s \tag{5}$$

其中，当 E_D^A 中包含有边 $e=(v_j \to v_i)$时，$A_{ij}=\alpha(e)$；否则 $A_{ij}=0$。d 用于调节基础集的重要性，它决定了从一个节点转移到相邻节点时，对象排序权值的传递比率。文[10]对 d 做了介绍，并设置 $d=0.85$，本文也采用了这个值。$s=[s_0,\cdots,s_i,\cdots,s_n]^T$ 是 $S(w)$的基础集向量。若 $v_i \in S(w)$，$s_i=1$；否则 $s_i=0$。

3 语义桌面索引排序算法

3.1 体系结构

本文提出一种三层体系结构来创建和提取元数据以强化桌面资源。体系结构的底层是个人计算机中已有的物理资源，即各种文件。我们将这些文件分为结构化文档、非结构化文档、Email、离线 Web 页面，其它文件和文件目录等。底层的物理资源为桌面搜索提供了部分基本信息，但是丢失了大量的上下文信，例如邮件的作者，网站的浏览路径等。我们利用 OWL 元数据记录和存储这些附加信息，这是体系结构的第二层，也就是语义桌面中的上下文工具提供的功能；最上层实现了资源的排序机制。我们为每个桌面资源都计算出一个"重要程度"，从而加强桌面检索程序的排序功能。整个体系结构如图 3 所示。

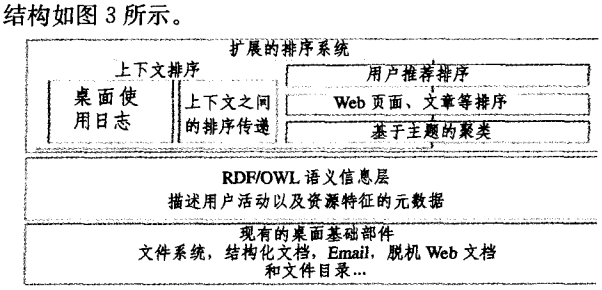

图 3 桌面排序系统体系结构

本文利用 PageRank 公式和权威度传递模式图的基本原理，计算得到每个资源的索引排序权值。以下是本文提出的排序算法。

3.2 索引排序算法

本算法多次访问权威度传递数据图 D^A，当 D^A 很大的时候，该算法的执行时间很长。但是通常情况下 D^A 并不大，因为它只存储了对象的 ID 和一组边的集合，足以适应大多数应用环境。下面给出创建资源索引的算法步骤：

1) 初始化参数，包括关键字列表 *keywordsList*，阈值 *epsilon* 和 *threshold* 等；

2) 对于关键字列表中的每一个关键字 w_i，做以下操作：

2.1) 若 $\exists v, |r^{(k-1)}(v) - r^k(v)| > epsilon$，则按照公式(2)，逐个计算每个 $r(v)$ 值，否则退出算法；

2.2) 对于满足 $r(v_i) < threshold$ 的 $r(v_i)$ 值，将二元组 $\langle id(i), r(v_i)\rangle$ 存储到索引列表中，并按降序排列；

3) 算法结束。

3.3 算法的优化

在实际应用中，很多事物的权威转移图可以抽象为有向无环图(DAG)。例如一个科技文献数据库，每篇论文只能引用在它之前出版的论文，这样的数据库就可以看作是一个 DAG。对于这样的数据模型，我们给出一种优化算法，它具有单一的权值传递路径，不需要多余的 epsilon 值，并能够计算出更精确的排序权值。因为本算法利用到公式(5)，所以得到的是精确值，而前面介绍的一般排序算法得到的是近似值。

本算法的实质是对象排序权值仅沿着拓扑顺序的方向传递，所以路径是唯一的。该算法不但试用于桌面资源排序，还可应用于数据库，以及其它含有链式结构的系统。具体算法步骤如下：

1) 按照拓扑顺序存储权威度传递模式图 D^A 中的节点；

2) 对于关键字列表中的每一个关键字 w_i，做以下操作：

2.1) 按照公式(5)，逐个计算每个 $r(v)$ 值；

2.2) 对于满足 $r(v_i) < threshold$ 的 $r(v_i)$ 值，将二元组 $\langle id(i), r(v_i)\rangle$ 存储到索引列表中，并按降序排列；

3) 算法结束。

在某些情况下，DAG 的性质可以通过权威度传递模式图 G^A 的结构推导得到，其定理如下：

定理 1 当且仅当权威度传递模式图 G^A 是一个 DAG 时，或者当 G^A 中的每一个子图都是一个 DAG 时，其对应的权威度传递数据图 D^A 是一个 DAG 图。

4 实验及效率评估

本文在 4 台个人计算机上分别选用开放式的语义桌面系统 DeepaMehta 和 Gnowsis 作为实验平台，版本分别为 DeepaMehta 2.0b7 和 Gnowsis beta 0.9.2。操作系统为 Microsoft Windows XP Professional，硬件配置为 P Ⅳ 2.6G 处理器，512M 内存。本文在 JBuilder2005 集成开发环境中，使用 Java 实现以上索引创建算法。

实验过程如下：首先，测试计算所有关键字的对象排序权值以及将它们存储到对象排序索引中的速度和占用空间的大小，本文使用的算法是“索引创建算法”(如 3.3 节所示)。实验反复修改变量 epsilon 和 threshold 的值，并调整排序对象数据集的大小。

表 1 反映的是在可搜索文件总数为 1448651 的个人计算机上，在 DeepaMehta 语义桌面环境下，当 epsilon 和 threshold 取不同的值时，用 3.2 节的算法创建排序索引的各项参数情况。

表 1 排序索引算法执行情况

epsilon	threshold	时间(秒)	关键字	空间(MB)
0.1	0.3	3702	84	2.20
0.1	0.5	3701	67	1.77
0.1	1.0	3702	46	1.26
0.05	0.5	3875	67	1.77
0.3	0.5	3517	67	1.77

从表 1 所示的实验结果中可以看出，当排序对象的阈值 threshold 增加时，索引所占用的存储空间减少，而创建索引算法的执行时间并不会随着 threshold 而变化，因为，threshold 并不参与到循环过程中。索引创建时间随着 epsilon 值的增高而降低，因为当算法所需的精确度要求下降时，算法循环的次数也会随之减少，而存储空间的大小不会随着 epsilon 的变化而改变。

表 2 反映的是在 4 台配置相同的个人计算机上，当 epsilon=0.05，threshold=0.1 时，对于不同的可搜索文件数目，创建索引执行时间和存储空间的比例关系。

表 2 文件数量与时间空间的比例关系

文件数量	时间(秒)	关键字	空间(MB)
11533	10512	21	0.7
37267	38467	65	1.7
124158	202013	316	7.4
410915	3639804	1745	42.5

从表 2 的实验结果可以看出，文件数量的增幅小于创建索引所需时间和占用空间的增幅。原因是，随着文件数量的增多，同样的关键字重复的次数增多。

结论 本文提出了一种在语义桌面环境下检索的排序方法，包括排序系统的体系结构和索引创建算法，并提出了在有向无环条件下的优化算法。通过多组实验验证了该方法的有效性，能够提高语义桌面的文件检索排序能力。

参 考 文 献

1 Balmin A, Hristidis V, Papakonstantinou Y. Objectrank: Authority-based keyword search in databases. In: VLDB, Toronto, Sept. 2004

2 Kleinberg J M. Authoritative sources in a hyperlinked environment. Journal of the ACM, 1999.46

3 Sauermann L. The semantic desktop-a basis for personal knowledge management. In: Maurer H, Calude C, Salomaa A, et al. eds. Proceedings of the I-KNOW 05. 5th International Conference on Knowledge Management, 2005

4 李胜，胡和平，卢正鼎. 语义桌面：个人计算机技术的未来发展方向. 计算机科学，2007.7

5 Google search engine. http://www.google.com

6 Haveliwala T. Topic-Sensitive PageRank. WWW Conference, 2002

7 Gnome beagle desktop search. http://www.gnome.org/projects/beagle/

8 Deepmehta semantic desktop for knowledge management. http://www.deepamehta.de/

9 Gnowsis semantic desktop. http://www.gnowsis.org/

10 Brin S, Page L. The Anatomy of a Large-Scale Hypertextual Web Search Engine. In: WWW Conference, 1998