Web搜索中的数据挖掘技术研究

耿 桦 李 媛 朱 炜 潘金贵

(南京大学计算机软件新技术国家重点实验室 南京大学多媒体技术研究所 南京210093)

摘 要 WWW已经成为世界上最大的分布式信息系统,如何快速有效地搜索用户所需的资源一直是研究热点。Web挖掘也已经成为数据挖掘中相对成熟的一个分支。本文针对Web资源搜索中利用的相关Web挖掘技术做一个综述。文章首先对目前流行的Web内容挖掘方面的常用技术进行了研究分析,然后着重研究了Web结构挖掘技术,介绍并评价了多种算法模型。接着介绍了用户使用的挖掘,并提出了Web内容挖掘技术,结构挖掘技术和用户使用挖掘相结合,应用于开发智能型搜索引擎的趋势。

关键词 Web挖掘,超链,超文本,PageRank,HITS,搜索引擎

A Research of Data Mining Technologies on Web Search

GENG Hua LI Yuan ZHU Wei PAN Jin-Gui

(State Key Laboratory for Novel Software Technology of Nanjing University, Multimedia Technology Institute of Nanjing University, Nanjing 210093)

Abstract WWW is now the largest distributed information system in the world, and how to find useful information is always a hot topic for researchers. Web mining has become an important branch of data mining. This paper mainly discusses mining technologies used in Web searching. The paper begins with talking about popular technologies in Web content mining, and then focuses on algorithms and models on Web structure mining. Then Web usage mining is briefly discussed. In the end the author advances that the technologies in Web content mining, Web structure mining and Web usage mining will be combined to develop intelligent search engines.

Keywords Web mining, Hyperlink, Hypertext, PageRank, HITS, Search engine

1 引言

随着互联网的日益盛行,Web搜索成了研究和应用的热点。Web搜索工具有两类:分类目录和搜索引擎。采用分类目录的搜索系统,比如Yahoo,利用树结构组织网页文档。文档预先通过手工分类,每个文档都对应树的节点。这种分类目录使用方便,效果也很好。但采用人工操作,更新慢,且覆盖面小。传统的搜索引擎比如Lycos和AltaVista也取得了一定的成功,它们采用信息检索领域的文本分析技术,利用用户提交的关键字去计算数据库中网页的相关度,然后返回相关网页。但它们仍会返回大量网页,而用户想得到的仅仅是少数相关性最好,最权威的网页,他们没有耐心也不可能去遍历搜索引擎返回的成百上千个网页。基于内容的分析方法还有一个明显的缺陷:很多查询主题上的权威网页本身并不包含查询关键字[6,7]。比如对于查询search engine,权威站点Yahoo和Infoseek的主页上找不到这两个关键词,故Yahoo和Infoseek就不会出现在搜索结果中。因此有关学者又提出了超链分析技术以优化搜索结果,其中最具代表性的就是PageRank[17]和HITS[15]算法。

随着网络信息资源的急剧增长,人们对搜索引擎的要求也越来越高。进行有关用户行为的挖掘有利于发现用户的兴趣[10],使得搜索结果更具针对性。利用数据挖掘中的关联规则挖掘还能发现相关网页。这些技术的综合使用,使得搜索引擎往智能化方向发展。

本文拟就Web搜索领域运用的Web挖掘技术进行研究分析,并做出总结。文章的第2节分析超文本环境中应用广泛的内容分析技术;第3节对超文本结构挖掘的主流技术HITS和PageRank算法进行比较分析,并研究一些改进模型;第4节简述用户行为的挖掘和关联规则挖掘技术在这方面的应用;最后总结上述技术在构建新型智能搜索引擎方面的应用。

2 Web内容挖掘

2.1 文本模型和超文本模型

Web内容挖掘源于传统的文本内容挖掘。在传统的信息检索领域,文档用向量空间模型(VSM)来表示[18]。对于一篇普通文档,首先利用一些句法规则和分词技术从文档中提取出多个token,接着对这些token进行词干还原,转换成原型。最后得到的每个项(term)就是欧式空间的一个坐标轴,而一个文档d就对应该空间的一个向量$V(d)$,该向量在每个坐标轴的分量就定义成文档中出现该term的频率W。用如下公式表示:

$$V(d)=(T1,W1(d);\cdots;Ti,Wi(d);Tn,Wn(d)) \quad (1)$$

其中Ti为词条项,$Wi(d)$为Ti在d中的权值(即出现频率)。为了节省开销和提高运行效率,还可以对向量进行降维处理,忽略考虑一些无用和不重要的词条项。可以通过特征选择技术来降维。比如去除标点,去除无意义的连词和权值低的

耿 桦 李 媛 朱 炜 硕士研究生,主要研究方向为多媒体,Web信息检索和Web挖掘;潘金贵 教授,博士生导师,主要研究方向为中间件,Agent技术,多媒体远程教育。

词条。更复杂的方法还包括计算信息增益，交叉熵以及潜在语义索引(LSI)。

向量空间模型的缺陷在于：每个词条(term)同等对待，没有考虑可能有些词比其他词都要关键和重要。基于此，TFIDF分析被引入向量空间模型。如果词条 t 在 N 篇文档中的 N_t 篇中出现，N_t/N 就表示词汇的欢迎度即重要度。定义 $IDF(t) = 1+\log(Nt/N)$，式(1)中的坐标分量改成：$Wi(d) * IDF(t)$。此外，还可以对文档向量其进行规范化，使每个向量的长度为1[5]。这种模型利用从训练文档中得到的单个词条作为特征，并只考虑词条出现与否或者出现频率，忽略其出现顺序[18]，又称为词袋模型。

对于超文本文档，则可以当作普通的文本再加上超链，最直观的方法是把超文本集合看作一个有向图，图中节点代表页面，有向边代表超链。而节点文档之间的连接情况往往暗示了一些语义联系。超文本内容本身也有区别于普通文本的地方，HTML 文档的各种 tag 提供了有效的信息。比如〈title〉和〈/title〉之间的文字就比普通文本重要得多。另外还可以考虑锚文本、URL、字体大小等等。Google 就充分利用了这一信息来分析网页文档的相关性[4]。

文[20]提出了评价网络资源的四种基于文本分析的算法模型：Boolean Spreading Activation，Most-cited，TFIDF 和 Vector Spreading Activation。前两种基于查询词在文档及其附近文档中的出现情况，后两种算法则引入了 TFIDF 算法分析网页和查询的相关度。

2.2 文档相似度计算

文档相似度在文档分类和聚类以及分析网页相关度中有重要的应用。最常用的文档相似度算法是余弦计算法。对于两个文档向量 $V1,V2$。其相似度定义为：

$$Similarity(V1,V2) = (V1 \cdot V2)/(|V1| * |V2|)$$

其中 $V1 \cdot V2$ 是标准向量内积，$|V1|$ 是向量 $V1$ 的长度。Bharat 的 HITS 改进算法[2]利用了文档和查询项的相似度计算，来过滤相关度低的网页。

2.3 潜在语义索引(LSI)

向量空间模型的文档相似度是通过考察文档之间词条的相似情况来实现的，但当两个语义相关的文档没有共同词条时，如何在它们之间建立语义联系？LSI(Latent Semantic Indexing)[12]利用了线性代数里的奇异值分解(SVD)，把词频矩阵(行对应词条，列对应文档，矩阵每个元素代表某词条在某文档中的出现频率或者 TFIDF 权值等)转换成奇异矩阵。对于词频矩阵 A，其奇异值就是 AA^T 的特征值。利用 SVD 把 A 分解成 $A = USV^T$。U 和 V 是列正交矩阵，S 是奇异值的对角矩阵。算法只保留 k 个最大的特征值及 U,V 中对应的 k 列。这样，一个词条映射成一个 k 维向量。最后用转换后的文档向量来计算相似度。

2.4 网页的自动分类

如果向 Yahoo 的分类目录中引入自动分类技术，就可以解决更新速度和覆盖率的问题。最常用的分类方法基于词频分析。首先预设好一些类别，和一些已经分好类的网页作为训练集。然后利用训练集，为每个类别中页面的所有词条生成一个词频向量。对于一个新的待分类的网页文档，先计算它的词频向量，然后和每个类别的词频向量作相似度比较，最后把最接近的类别作为新网页的所属类别。为了减少计算量，同样可以对词频向量降维。该分类方法的精确性依赖于训练集和预设类别的质量，以及词频向量的计算。

文[9]提出了把网页自动分类技术和基于关键字的查询相结合。文中采用的分类方法最后返回和待分类网页最接近的类别列表(多个类别，按相似程度先后排列)。用户查询时提交关键字和期望类别，系统先根据关键字检索网页，然后对每个结果网页采用如下过滤技术：用户所选类别是否出现在该结果网页的类别列表的前 k 位(k 比较小)。如果不存在，则该网页被舍弃。这种改进的好处在于，算法结果不过分依赖网页自动分类的精确性。

2.5 Google 中的 IR 技术

在 Google 系统中，人们利用 PageRank 算法和 IR 技术相结合来计算网页文档的相关度权值[4]。比如，对于一个单词汇查询，首先分析该词汇在网页文档数据库中的命中列表。每种命中类型(指查询词出现位置比如标题、锚文本、URL、普通文本，另外还考虑字体大小等)都有类型权重，组合到一起形成一个类型向量，接着根据命中数目得到一个计数权重组成的向量(向量的每个分量和对应命中数呈线性关系)，最后把两个向量点乘作为该文档的 IR 分值。IR 分值和 PageRank 值组合得到网页文档的最终相关度级别。

3 超链结构挖掘

自1998年 PageRank 算法[17]和 HITS 算法[15]提出以来，很多学者致力于网络超链结构的分析和研究上，他们在研究 PageRank 和 HITS 算法的基础上，提出了很多改进算法和模型，并且成功应用于一些搜索系统。

3.1 PageRank 算法

算法反映了用户的一种浏览倾向，rank 值高的网页，用户浏览的概率大。rank 值最高的网页并不一定是包含了查询关键字信息最多的网页，但很可能是用户最感兴趣的。算法假设了一个用户随机浏览的模型：用户首先浏览某网页，然后继续浏览时，他随机点击该网页的一个超链进入其他网页，如果该网页没有超链指向其他网页，或者用户厌倦了点击超链的浏览方式，他可以在地址栏随机输入一个网址。其递归定义公式如下：

$$PR(u) = p/N + (1-p)\sum_{v \to u}[PR(v)/Outdeg(v)] \quad (2)$$

式中的 $PR(u)$ 代表网页 u 的 PageRank 值，N 是总的网页数目，$1-p$ 是点击超链进行浏览的概率，p 是随机输入网址的概率，$Outdeg(v)$ 是网页 v 的出度(出度在图论中指节点的出边个数，此处即被该网页指向的网页个数。出边在图论中指某节点的引出边，在本文中用来表示某网页的引出超链。同样，某节点的入边原意是指向该节点的边，本文中表示指向该网页的超链)，$v \to u$ 表示网页 v 有超链指向 u。算法假设随机输入网址时每个网址被输入的概率都是 $1/N$。

L. Page 在原文中给出的公式[17]如下：

$$PR(u) = cE(u) + c\sum_{v \to u}[PR(v)/Outdeg(v)] \quad (3)$$

这里 c 是规范因子，引入向量 $E(u)$ 是为了防止沉积现象：当一组网页节点构成循环，且只有入边没有出边时，随着迭代进行，该组网页的 PageRank 值不断积累而永不收敛。它模拟了如没有出边，则随机选择其他页面浏览。两个公式的基本思想是一致的。

3.1.1 利用高斯迭代加快 PageRank 的收敛

PageRank 算法是计算整个 Web 上所有页面的 rank 值的，因此计算量大。且由于 Web 的更新速度很快，所以 PageRank 也要经常更新。A. Arasu 等人通过改写迭代公式，利用高斯-塞德尔迭

代取代了雅可比迭代，大大加快了算法收敛的速度[1]。

由于 PageRank 算法写成矩阵形式为：$x = (1-p)e + pAx$，因此原来的迭代公式为：

$$x_i^{(k+1)} = (1-p) + p\sum_{(j,i)\in N} a_{ij} * x_j^{(k)}$$

改进后每步迭代都利用了最新计算的 PageRank 值，从而加快了算法的收敛。改进公式如下：

$$x_i^{(k+1)} = (1-p) + p\sum_{j<i} a_{ij} * x_j^{(k+1)} + p\sum_{j>i} a_{ij} * x_j^{(k)}$$

3.1.2 Adaptive PageRank 算法及其改进 斯坦福大学的 S. Kamvar 等人发现，Web 上的大部分网页收敛比较快，但少数 PageRank 值高的网页，往往收敛速度很慢。他们提出了 Adaptive PageRank 算法[13]。设 C 为给定步骤内收敛的网页集合，N 为尚未收敛的网页集合。定义：$P' = cP + (1-c)E$，$A = (P')^T$（P_{ij} 定义为 $1/Outdegree(i)$，如果网页 i 有链接指向 j；或者 $1/n$，如果网页 i 上没有超链），则 PageRank 算法表示成 $x^{(k+1)} = Ax^{(k)}$。

Adaptive PageRank 算法把 A 分为两个子矩阵：A_N 和 A_C。A_N 是 $m * n$ 阶对应于尚未收敛的 m 个网页的入边的子矩阵，A_C 则是 $(n-m) * n$ 阶对应于已收敛的 $n-m$ 个网页的入边的子矩阵。把 A 和 $x^{(k)}$ 可以表示成：$A = (A_N, A_C)^T$ 和 $x^{(k)} = (x_N^{(k)}, x_c^{(k)})^T$。由于 $x_c^{(k)}$ 已经收敛，所以迭代过程简化成两块，为：

$$x_N^{(k+1)} = A_N * x^{(k)} \text{ 和 } x_c^{(k+1)} = x_c^{(k)}$$

注意到 A_C 不再在迭代过程中使用，故 A 还可以进一步改写成：$A' = (A_N, 0)^T$。如定义 $x_c'^{(k)} = (0, x_c^{(k)})^T$，则上述迭代式改成：$x^{(k+1)} = A'x^{(k)} + x_c'^{(k)}$。由于矩阵 A' 非常稀疏，故开销减少。

3.2 HITS (Hyperlink Induced Topic Search)算法

J. Kleinberg 在文[15]中提出了权威(authority)网页和中心(hub)网页的概念。互联网上每个主题都有大量相关网页，如果某网页被相当数量的其他网页指引，也就是被大量的网页作者认可，则说明该网页在该主题上有很高的权威度，该网页就称作权威网页。同时，某些网页可能本身不够权威，但却指向大量权威网页，这种网页称作中心网页。权威网页和中心网页之间，形成了一种互相促进互相增强的关系：一个好的中心网页应该指向很多好的权威网页，而一个好的权威网页则应该被很多好的中心网页所指向。

HITS 算法从根集 S 开始。S 这样定义：首先把查询交给某搜索引擎，对于参数 k，取返回结果中排在前 k 位的网页，构成根集。然后算法扩展该根集形成集合 T，方法是加入引用了 S 中网页的网页，和被 S 中网页引用的网页。为防止 T 过度膨胀，可以对网页入度（即指向该网页的网页个数）加上限 d（文[15]中把 k 取为200，d 为50）。然后以 T 中的所有网页为节点，网页之间的超链为边（删除同一站点内部的链接），构造有向图 G。分别以 h_u 和 a_u 代表文档节点 u 的 *hub* 值和 *authority* 值并全部初始化为1，算法对这两个值进行迭代计算：

$$a_v = \sum_{u\to v} h_u \quad \text{和} \quad h_u = \sum_{u\to v} a_v$$

每步迭代都要规范化。最后得到每个网页的中心度和权威度。Kleinberg 还给出了迭代收敛的严格证明，并指出如果定义 A 为 G 的邻接矩阵，则最后结果权威度向量和中心度向量分别就是 A^TA 和 AA^T 的主特征向量（模最大的特征值对应的特征向量）。通常取权威度最高和中心度最高的数个网页作为结果。算法最后得到的网页集合，对应于主特征向量，我们称这个集合为主社区。实验证明，HITS 算法对于许多查询有很好的搜索结果。

3.2.1 BHITS 和 WBHITS 算法 HITS 算法最容易产生的问题是主题泛化和漂移现象。算法中所有超链同等对待，很容易产生主题漂移。Bharat 等人针对该问题作了改进，提出 BHITS 算法[2]。首先，从 T 中剔除无关网页。他们通过引入传统的文本分析的方法，如下定义了文档 D 和主题 Q 的相似度，并设置恰当的阈值来剔除相关度低的网页。其中 $U_{iq} = FREQ_{iq} * IDF_i$，$U_{ij} = FREQ_{ij} * IDF_i$，$FREQ_{iq}$ = 项 i 在查询 Q 中的频度，$FREQ_{ij}$ = 项 i 在文档 Dj 中的频度。

$$Similarity(Q,Dj) = \Sigma(U_{iq}, U_{ij})/[(\Sigma U_{iq}^2) * (\Sigma U_{ij}^2)]^{1/2}$$

其次，引入权重。如果站点 A 有 k 个网页指向站点 B 的某个网页，则 A 上的 k 个网页对 B 的权威值的贡献总和为1，每个网页贡献 $1/k$。同样，A 的某个网页指向 B 的 l 个网页，则 B 的 l 个网页对 A 中该网页的中心值的贡献总和为1，每个网页贡献 $1/l$。BHITS 算法对主体漂移问题的解决有明显的效果。

BHITS 只对相同站点之间超链的权值做了修改。当 T 包含入度很小出度很大的网页时，这些网页拥有很大的 *hub* 值，因此被这些网页所指向，往往很多不相关的网页会得到很高的权威值。WBHITS(Weighted BHITS)算法[16]对超链权值进一步做了调整。迭代方程如下：

$$A[i] = \sum_{(i,j)\in N} H[j] * auth\_weight(j,i) \tag{4}$$

$$H[i] = \sum_{(j,i)\in N} A[j] * hub\_weight(j,i) \tag{5}$$

式(5)中的 $hub\_weight(j,i)$ 均设为1。而式(4)中 $auth\_weight(j,i)$ 的设定如下：先看 T 是否包含这样的网页：在所有网页中是入度最小的三个网页中的一个，同时又是出度最大的三个网页中的一个。如果存在，则所有超链的 $auth\_weight(j,i)$ 值设为4。否则 $auth\_weight(j,i)$ 均设为1。然后迭代一步。这时再看是否出现这样的网页：在所有网页中是权威度最低的三个网页中的一个，同时又是中心度最高的三个网页中的一个。如果存在，则所有超链的 $auth\_weight(j,i)$ 设为4。

3.2.2 Clever 系统对 HITS 的改进 IBM 研究中心的 Clever 工程组通过文本分析来调整超链权值，以优化 HITS 算法。他们提出的 ARC 算法[8]的基本思想是：如果超链的锚文本或者其附近文本出现了关键字，则说明该超链指向的目标网页很可能和主题相关，且出现关键字的次数越多，则相关性越好。其权值公式为：

$$W(p,q) := 1 + n(t)$$

其中 $W(p,q)$ 表示 p 指向 q 的超链权值，$n(t)$ 表示查询词 t 在超链对应的锚文本及其前后各 B 个字节中出现的次数。文[8]中经过实验把 B 取为50。

3.3 HITS 算法的改进方向

纵观几种基于 HITS 的改进算法，采取的办法往往是修改超链的权值来避免主题漂移。而 HITS 主题漂移的重要原因，就在于它最后得到的主社区虽然结构紧密，但相关性有时候并不好，而且由于结构的紧密性导致主题漂移通常是整体性的，这是致命的。而修改超链权值，就会使得一些和其他网页联系尽管不是很密切，但由于超链权值大，而排名提前。这就导致了原来的主社区被打破，最后的结果网页并非再同属一个社区。但是结果网页的质量却较理想。

另一种改进方向，可以从其它非主社区（对应非主特征向量的社区）着手。实践表明，往往一些非主社区和主题的相关

性较好。此时可以引入内容分析方法来确定相关性比较好的社区。引入非主社区的好处还体现在可以解决一词多意现象，这是搜索引擎遇到的一个经典难题。普通搜索引擎无法按照多义词的不同意义来区分结果网页，而只是根据计算出的相关度对网页排列，然后返回给用户，显然这并不理想。对应不同特征值的多个社区恰好提供了这种多义词的自动分类。由于每个社区整体结构性良好，因此其结果往往针对性更好。

4 Web 用户使用挖掘

4.1 使用挖掘概述

网页质量的高低最终还是要通过用户来评价，因此仅仅通过对网页文档的静态分析来评价网页的相关性仍是不够的。通过用户行为的分析，挖掘出用户感兴趣的网页也是一个有趣的课题。和上面的挖掘技术略有不同，用户使用挖掘（或行为挖掘）的数据往往不是原始的超文本数据，而是用户在浏览网页过程中记录和抽取出来的。这主要包括用户的客户端 Web 访问记录和服务器端，代理服务器端的日志。

关于用户数据，还有其他类型和获得途径。Alexa[22]的作法是，通过让用户安装浏览器插件来把用户的浏览情况反馈给服务器。服务器再进行用户浏览的行为分析。最后服务器利用分析结果，提供一些用户感兴趣的资源链接给用户。

Web 使用挖掘一般分三个步骤：数据预处理，模式发现和模式分析[19]。而预处理工作是使用挖掘中难度最大的。文[11]对几种数据预处理方法作了详细的比较。

模式发现过程就是利用统计分析，数据挖掘等技术的过程。最简单的方法就是对用户行为进行统计分析。例如对搜索引擎返回的网页进行使用分析可以优化搜索结果，把浏览次数多，比较流行的网页排在前面。此外还包括对用户群按其访问特点进行分类或聚类，对网页按其内容相似度分类或聚类等。最后的模式分析主要是去除一些无用规则和模式。

4.2 网页相似度挖掘

文本挖掘利用向量计算文档相似度，超链挖掘利用网页之间的引用关系发现相关性。而对网络用户的行为进行关联挖掘，可以发现网页之间的潜在语义联系。

关联规则挖掘是传统数据挖掘领域一个重要的研究和应用方向。关联规则挖掘通常包括两步：找出频繁项目集，和从频繁项目集中挖掘出关联规则。后面一步是容易做到的。关于第一步，流行的 Aprior[14]算法是解决方法之一。算法的大致思想如下：如果定义 C_i 为候选频繁 i-项集（包含 i 个项目的集合），L_i 为频繁 i-项集。则首先从 C_1 开始，算法要设定一个最小支持度 min_sup 作为阈值来确定频繁集。每次从 C_i 中选取在用户数据库中出现频率大于 min_sup 的项集，作为 L_i。实现的伪代码如下：

```
L1= 用户数据库中的频繁1-项集;
for (k = 1; Lk不为空; k++) {
    Ck+1=从频繁 k-项集生成候选频繁 k+1-项集;
for 用户数据库中的每条记录 t {
    Ct= subset (Ck+1, t);//候选集 Ck+1中 t 的子集
for each candidate c ∈Ct
    c. count ++;}
Lk+1= {c∈Ck+1| c. count >= min_sup};
}
return L = {Ln | n = 1,2,…,k};
```

利用关联规则挖掘从大量用户浏览记录中挖掘出用户同时浏览的一些站点和网页，这些网页之间往往有语义联系或者多少反映了用户的共同兴趣。关联规则还可以用作启发式原则，让服务器预先从其它站点下载相关文档，以提升系统效率。

4.3 兴趣挖掘和行为预测

对特定用户的行为进行跟踪分析，可以发现他们的兴趣，预测他们的行为，使搜索引擎的服务更具针对性。通过学习用户的行为偏好，站点还可以自动对其结构和组织进行优化。

文[10]中开发了一种自动记录用户日志数据的工具，它用 XML 格式记录用户的五种行为：浏览，点击，查询，保存，关闭。文中提出利用简单贝叶斯分类器和贝叶斯信念网络对用户的行为分类并建模，并预测用户的兴趣和行为。

文[3]则提出了重构网络（restructure hypertext network）的概念，并构建了 Adaptive Hypertext 系统。系统的目的是把网络改造成各类用户群喜欢的链接模型。系统使用反馈功能来重构超链，而基本上不需要网络设计者的干预。最初超链被赋予初始权值，随着用户的访问情况，训练出一组学习规则，利用规则对超链的权值进行相应的调整，最后网页中的超链按照新权值重新排列，实现网络的重构，并反馈给用户。

结论 从以上分析可以看到，内容挖掘，结构挖掘和使用挖掘都已经有了一些成形的算法，也有其各自的优势和不足。内容挖掘利用了网页文档的语法信息，结构挖掘实际上是利用了网页设计者自身对网页的评价信息，而使用挖掘则是利用用户的评价信息。随着互联网信息的急剧增长和用户多元化的需求，上述技术的结合将能极大地优化搜索引擎的功能和效果。而最近 Google 又推出个性化服务和 Email 功能，新一代智能化搜索引擎已是形势所趋。

随着用户对信息的多样化需求和网络挖掘研究的不断深入，智能化将是目前搜索引擎发展的目标和方向，Web 领域的搜索技术研究也将会变得非常有趣且具有挑战性。

参考文献

1 Arasu A, Novak J, Tomkins A, Tomlin J. PageRank Computation and the Structure of the Web: Experiments and Algorithms. In: 11th Intl. World Wide WEB Conf. 2002

2 Bharat B, Henzinger M R. Improved algorithms for topic distillation in a hyperlinked environment. In: ACM Conf. on Research and Develop. in Info. Retrieval(SIGIR'98), 1998

3 Bollen J, Heylighen F. A system to restructure hypertext networks into valid user models. In: The New Review of Hypermedia and Multimedia, 1998

4 Brin S, Page L. The anatomy of a large-scale hypertextual web search engine. In: Proc. of the 7th World-Wide Web Conf. (WWW7), 1998

5 Chakrabarti S. Data mining for hypertext: A tutorial survey. ACM SIGKDD, Jan 2000,1(2)

6 Chakrabarti S, Dom B, Kumar S R, et al. Mning the Web's Link Structure. In 1999 IEEE, 1999. 60～67

7 Chakrabarti S, Dom B, Gibson D, et al. Mining the Link Structure of the World Wide Web. IEEE Computer, 1999

8 Chakrabarti S, Dom B, Gibson D, et al. Automatic resource compilation by analyzing hyperlink structure and associated text. Computer Network and ISDN Systems, 1998

9 Chekuri C, Goldwasser M H, Raghavan P, Upfal E. Web Search Using Automatic Classification. In: Proc. of WWW-96, 6th Intl. Conf. on the World Wide Web, 1996

10 Chen Z, Lin F, Liu H, et al. User Intention Modeling in Web Applications Using Data Mining. In: Internet and Web Information Systems, 2002. 181～191

11 Cooley R, Mobasher B, Srivastava J. Data preparation for mining world wide web browsing patterns. In: Knowledge and Information Systems, 1999

12 Deerwester S, Dumais S T, Landauer T K, et al. Indexing by latent semantic analysis. Journal of the Society for Information Sci-

ence, 1990. 391～407
13 Kamvar S D, Haveliwala T H, Golub G H. Adaptive Methods for the Computation of PageRank. In:Linear Algebra and its Applications, Special Issue on the Numerical Solution of Markov Chains, 2003
14 Han J, Kamber M. Data Mining: Concepts and Techniques. Morgan Kaufmann, 2001
15 Kleinberg J. Authoritative sources in a hyperlinked environment. In:ACM-SIAM Symposium on Discrete Algorithms, 1998
16 Li L Z, Shang Y, Zhang W. Improvement of HITS-based algorithms. on web documents. In:Proc. of the eleventh intl. conf. on World Wide Web, 2002. 527～535
17 Page L, Brin S, Motwani R, Winograd T. The PageRank citation ranking: Bringing order to the web. Stanford Digital Libraries Working Paper, 1998
18 Salton G, McGill M J. Introduction to Modern Information Retrieval. McGraw-Hill, 1983
19 Srivastava J, Cooley R, Deshpande M, et al. Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data. In 2000 ACM SIGKDD, Jan. 2000
20 Yuwono B, Lee D L. Search and Ranking Algorithms for Locating Resources on the World Wide Web. In:Proc. of the Twelfth Intl. Conf. on Data Engineering, 1996. 164～171
21 朱炜，王超，李俊，潘金贵. Web超链分析的算法研究. 计算机科学, 2003,30(9)
22 Alexa Internet, Inc. http://www.alexa.com, 1996-2004

(上接第10页)

会议 VLDB 和 ACM SIGMOD 等都将数据流作为一个热点问题来讨论，而我国在这方面的研究才刚刚起步。本文简要分析了目前三个典型原型系统的基本实现技术，并将这个领域目前的研究热点和今后的发展趋势介绍给广大计算机工作者，希望我国有更多的感兴趣的同行加入到这一领域的研究行列中来，以提高我国在这一领域的整体研究水平。本文也给出了一个基于硬件预处理的并行数据流管理原型系统的体系结构。

参 考 文 献

1 Arasu A, Babu S, Widom J. An Abstract Semantics and Concrete Language for Continuous Queries over Streams and Relations: [Technical Report]. Nov. 2002. http://dbpubs. stanford. edu: 8090/pub/2002-57
2 Motwani R, et al. Query Processing, Approximation, and Resource Management in a Data Stream Management System. In: Proc. Conf. on Innovative Data Syst. Res, 2003. 245～256
3 Chandrasekaran S, Franklin M. Streaming Queries over Streaming Data. In: Intl. Conf. on Very Large Databases (VLDB), Hong Kong, 2002
4 Chandrasekaran S, et al. TelegraphCQ: Continuous Dataow Processing for an Uncertain World. In:Proc. Conf. on Innovative Data Syst. Res, 2003. 269～280
5 Carney D, et al. Monitoring streams-A New Class of Data Management Applications. In: Proc. Int. Conf. on Very Large Data Bases, 2002. 215～226
6 Cherniack M, et al. Scalable Distributed Stream Processing. In CIDR, Asilomar, CA. Jan. 2003
7 Carney D, et al. Operator Scheduling in a Data Stream Manager. In: Proc. of the 29th Intl. Conf. on Very Large Data Bases (VLDB'03), Berlin, Germany, Sep. 2003
8 Wang H, Zaniolo C. ATLaS: A Native Extension of SQL for Data Mining and Stream Computations, UCLA CS Dep. 2002
9 Naughton J, DeWitt D, Maier D. The Niagara Internet Query System. University of Wisconsin, 2002
10 Bonnet P, Gehrke J, Seshadri P. Towards Sensor Database Systems. In:Proc. Int. Conf. on Mobile Data Management, 2001. 3～14
11 Cortes C, Fisher K, Pregibon D, Rogers A. Hancock: a language for extracting signatures from data streams. In: ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining, 2000. 9～17
12 Madden S, Franklin M J. Fjording the stream: An architecture for queries over streaming sensor data. In:Proc. of 18th Intl. Conf. on Data Engineering, 2002
13 Terry D, Goldberg D, Nichols D, Oki B. Continuous queries over append-only databases. In:Proc. of the 1992 ACM SIGMOD Intl. Conf. on Management of Data, June 1992. 321～330
14 Sullivan M, Heybey A. Tribeca: A System for Managing Large Databases of Network Traffic. In: Proc. of the USENIX Annual Technical Conf., New Orleans, LA, 1998
15 Madden S, Shah M, Hellerstein J, Raman V. Continuously adaptive continuous queries over streams. In: Proc. ACM SIGMOD Intl. Conf. on Management of Data, Madison, Wisconsin, May 2002. 49～60
16 Shah M, Hellerstein J, Chandrasekaran S, Franklin M. Flux: An Adaptive Partitioning Operator for Continuous Query Systems: [Technical Report CS-02-1205]. U. C. Berke-ley, 2002
17 Zhu Y, Shasha D. StatStream: Statistical Monitoring of Thousands of Data Streams in Real Time. In VLDB, 2002
18 Domingos P, Hulten G. Mining high-speed data streams. In: ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining, 2000. 71～80
19 Guha S, Mishra N, Motwani R, O'Callaghan L. Clustering data streams. In: the Annual Symposium on Foundations of Computer Science, IEEE, 2000
20 Gilbert A C, Kotidis Y, Muthukrishnan S, Strauss M. Surfing wavelets on streams: One-pass summaries for approximate aggregate queries. In VLDB, 2001. 79～88
21 Guha S, Koudas N, Shim K. Data streams and histograms. In: Proc. of Symposium on Theory of Computing, 2001. 471～475
22 Viglas S D, Naughton J F. Rate-based query optimization for streaming information sources. In: Proc. ACM SIGMOD Intl. Conf. on Management of Data, Madison, Wisconsin, May 2002. 37～48
23 Babu S, Widom J. Exploiting k-constraints to reduce memory overhead in continuous queries over data streams: [Technical report]. Stanford University Database Group, Nov. 2002. Available at: http://dbpubs. stanford. edu/pub/2002-52
24 Chen Y, Dong G, Han J, Wah BW, Wang J. Multi-Dimensional Regression Analysis of Time-Series Data Streams. In VLDB, 2002
25 Kang J, Naughton J F, Viglas S D. Evaluating Window Joins over Unbounded Streams. In ICDE, Feb. 2003
26 Arasu A, Babcock B, Babu S, McAlister J, Widom J. Characterizing memory requirements for queries over continuous data streams. In:Proc. 21st ACM SIGACT-SIGMOD-SIGART Symp. on Principles of Database Systems, May 2002. 221～232
27 Babcock B, Babu S, Datar M, Motwani R, Widom J. Models and issues in data stream systems. In:Proc. of the 2002 ACM Symp. on Principles of Database Systems, June 2002. 1～16
28 Dobra A, Gehrke J, Garofalakis M, Rastogi R. Processing complex aggregate queries over data streams. In: Proc. of the 2002 ACM SIGMOD Intl. Conf. on Manage-ment of Data, 2002. 61～72
29 Aanur R, Hellerstein J. Eddies: Continuously Adaptive Query Processing. In SIGMOD, 2000. 261～272
30 Babcock B, Babu S, Datar M, Motwani R. Chain: Operator Scheduling for Memory Minimization in Stream Systems. In:Proc. of the Intl. SIGMOD Conf. San Diego, CA, 2003
31 Manku G S, Motwani R. Approximate Frequency Counts over Streaming Data. In: Proc. of the 28th Intl. Conf. on Very Large Data Bases (VLDB 2002), Aug. 2002
32 Charikar M, Chen K, Farach-Colton M. Finding frequent items in data streams. In:Proc. of 29th Intl. Colloquium on Automata, Languages and Programming, 2002