Web 链接结构信息研究综述*)

李 剑 金蓓弘
(中科院软件所 软件工程技术中心 北京100080)

A Survey of Web Link Structure Information Research

LI Jian JIN Bei-Hong
(Technology Center of Software Engineering, Institute of Software, Chinese Academy of Sciences, Beijing 100080)
(lijian@otcaix.iscas.ac.cn)

Abstract As the size of WWW is growing at an incredible rate, there is some limitation in the methods that only analyzes the Web pages' information. This paper presents a basic model of Web link structure. Then it classifies the algorithms that analyze the Web link structure information and their applications. At last, it presents the practical approach of analyzing Web link structure information.

Keywords Web link structure, Clustering, Rank page, Topic distillation

万维网(World Wide Web)是由大量的网页组成的,网页之间由超链接(HyperLink)相互连接。在传统上,人们对网络信息的分析和获取是依靠对网页内容的分析和处理来进行的。例如,传统的网络搜索引擎对网页上文本信息进行分析、索引,并将处理后的信息存储在数据库中,然后根据用户查询输入进行分析,获得查询结果。

然而单一依据网页文档信息进行信息分析和获取具有其局限性。首先,网络的规模正在迅速增长,网络信息量随着网页数目的急剧增加而迅速增大,而现在对网络信息进行分析获取的程序一般是中央化的(不是分布式的),其只具有有限的处理能力和存储能力,能否满足网络信息量迅速增大的需要,这是一个问题。其次,网页的文本信息一般是由自然语言组成,对其的分析处理也不可能十分精确。

近年来,人们试图分析 Web 上的链接结构信息,并以此辅助对网络信息的分析和获取。本文首先给出表示 Web 链接结构的基本模型,然后对 Web 链接结构信息的各种分析算法及其应用进行了分类和综述,最后指出了 Web 链接结构信息分析的实用途径。

1 Web 链接结构模型

在 Web 上,网页可以看作一个结点,而网页上的超链接从一个网页指向另一个网页,它可以被看作从一个网页结点到另一个网页结点的有向边或这两个网页结点之间的无向边。因此,整个 Web 链接结构可以建模为一个由结点和结点之间的边组成的有向图或无向图,我们称之为 Web 链接结构图,参见定义1。从网络结构图中可以发掘出许多有用的信息,称之为 Web 链接结构信息。

定义1 设 V 是一个网页集合,则可以依此生成一个图 $G=(V,E)$,其中 V 表示网页结点集。若 $p,q\in V$,它们为两个网页,在网页 p 中存在一条超链接(HyperLink)指向网页 q,那么在图中存在一条边 $e=(p,q)\in E$。

这一结构图便称之为 Web 链接结构图。根据边的类别分为有向边和无向边,其 Web 链接结构图可以分为有向图和无向图两类。根据图论的知识和方法可以在 Web 链接结构图中分析和发掘出的知识和信息,称之为 Web 链接结构信息。

定义2 给定图 $G=(V,E)$,其中 $V=\{p_1,p_2,\cdots,p_n\}$,矩阵 W 是图 G 的相邻矩阵,其中:如果 (p_i,p_j) 是 G 中的边,则 $W_{ij}=1$,否则 $W_{ij}=0$。

Web 链接结构图的相邻矩阵反映了 Web 链接结构图的连接信息。

2 Web 链接结构分析算法

基于上述 Web 链接结构模型,近年来提出了多种 Web 链接结构的分析算法,我们将它们分别归类于网页聚簇性分析算法、页面权值分析算法等。

2.1 网页聚簇性

人们在对网络结构的研究中,很早就发现:如果网页之间的相互链接关系愈密切,其内容相关性也愈大。网页总是趋向于连接到与自己内容相关的网络结点上。因此,很早就有人根据图论中的算法将 Web 链接结构图划分为不同粒度的聚簇,每一个聚簇中的结点网页内容关系紧密,它们趋向于属于同一主题。

Batafogo 是最早研究网络聚簇性的人之一,他在1991年就提出了两个经典的聚簇算法[1]。他首先划分出两种类型的网页结点:索引(Index)结点和引用(Reference)结点。索引结点是那些扇出值较大的结点,也就是它的输出链接数较多。引用结点是那些扇入值较大的结点,它们的输入链接数较多。Batafogo 根据设定的阈值来确定这两类结点。

定义3 在无向图中,如果存在这样一个结点 a,并存在另外两个结点 v 和 w,在 v 和 w 之间的任意路径都通过 a 结点,则结点 a 称为连接(articulation)结点。

定义4 不包含连接结点的连接图称为互连接(biconnected)图。

定义5 在有向图 G 中,对任意两个结点 a 和 b,如果有

*)本文研究得到国家重点基础研究发展规划973资助项目(G1999035807)和国家自然科学重点基金(69833030)的资助。李 剑 博士生,主要研究方向为基于内容的网络信息处理。金蓓弘 博士,副研究员,主要研究方向为分布式计算,数据库实现技术。

一条从结点 a 到结点 b 的路径，就存在一条从 b 到 a 的路径，那么这一有向图 G 称为强互连接(strongly biconnected)图。

Batafogo 提出的第一个算法首先找到图 G 中的索引和引用结点，如果没有找到，则其算法只执行一次，然后对索引结点删除输出边，并对引用结点删除输入边，接着将图 G 的连接看作是无向边，删除图 G 中的连接结点，从而形成多个互连接子图，此后对每个互连接子图递归执行这一算法。此算法最后生成一系列子图，子图结点间的聚簇性较大。

Batafogo 的另一个算法和这一算法略有不同，其改动在于：在寻找图中的索引和引用结点时，如果没有找到，则在算法执行一次后，再将每个互连接子图变换成有向图，并将其分解为强互连接子图。

Batafogo 注意到这些生成的子图中结点的内容相关性比较大，因此他使用这种方法来对文档进行聚簇，将聚簇后的文档合成大文档。后人对其研究进行了扩展，将其主要应用在主题划分等领域内。

这种聚簇方法可以看作为单纯依据 Web 链接结构信息的聚簇。如果两个网页结点之间的连接路径越多，连接路径越短，则两个网页结点之间的"距离"就越近，那么它们属于同一聚簇的可能性就越大。

2.2 页面权值

根据 Web 链接结构图，可以使用多种方法计算出网页结点不同类别的重要性，可以将其称之为页面权值，也是一种 Web 链接结构信息。

2.2.1 PageRank 值　它在一定程度上表示了网页的重要性。其定义为：$R(u)=\sum_{v\in Bu}R(v)/Nv$，其中 u 表示当前网页，R(u)表示 u 网页的 PageRank 值。F_v 表示 v 页指向的页面集合，Bu 表示指向 u 页的页面集合，$N_v=|F_v|$，它表示对应页面集合的网页个数。其基本思想是：如果一个页面被很多的重要页面指向，则这个页面也很重要。但是有的页面可能指向很多的页面，其输出边的数目也应该被考虑，因此将每个输入页面的权值除以其输出链接数，然后求和。这一算法是递归执行的。

在实际的网络结构中，肯定会出现有循环回路的链接路径，例如网页 a 指向网页 b，同时网页 b 也指向网页 a。这样，在一次递归计算中，网页 a 的 PageRank 值增加会使网页 b 的 PageRank 值增加，反过来网页 b 的 PageRank 值增加也会使网页 a 的 PageRank 值增加。因此，如果采用这种方法计算会导致这一循环回路中的页面 PageRank 值不断增长。Brin 等[2]对此算法作了一定的修正，保证了其算法的收敛性，其新的计算方式为：$R'(u)=c\sum_{v\in Bu}R'(v)/Nv+cE(u)$，其中 E(u)是对各结点权值的一个固化设置值，c 是可变的变量，它需要使 $\|R'\|_1=1$(R'表示各网页结点的 PageRank 值构成的向量，$\|R'\|_1$表示向量的各分量之和)。Brin 等证明了这一计算是收敛的。其计算网页结点 PageRank 值的算法如下：

W 是 Web 链接结构图的相邻矩阵，R 表示各网页结点的 PageRank 值向量。在每一次递归计算中，$R_{i+1}\leftarrow WR_i$(R_i 表示第 i 次计算得出的 R 的值)，并且计算出来的新的 R_{i+1}需要加上一个偏差值($R_{i+1}\leftarrow R_{i+1}+dE$)，其中 d 为此次计算前后的偏差度，$d=\|R_i\|_1-\|R_{i+1}\|_1$，E 为预先设定的偏差向量。算法在前后两次计算结果的差值($\|R_{i+1}-R_i\|_1$)小于一个设定的微小值 ε 后结束。

2.2.2 hub 结点和 authority 结点　Kleinberg 在文[3]中总结出了有向图中两类特殊的结点：集中(hub)结点和权信(authority)结点。它们与 Index 和 Reference 结点有些相似。所谓集中结点是指指向很多重要结点的结点。而权信结点是被很多重要结点指向的结点。指向很多好的权信结点的结点是好的集中结点，而被很多好的集中结点指向的结点是好的权信结点。Kleinberg 提出的计算集中结点和权信结点的方法如下：

N 为网页结点集，$n\in N$，h(n)为 n 结点的 hub 值，a(n)为 n 结点的 authority 值，a 和 h 分别是这些值组成的 N 维向量。首先对每个结点的 hub 值和 authority 值进行初始化。然后反复计算各结点的 hub 值和 authority 值：$h[n]:=\Sigma_{n'}a(n')$，(n'为被 n 指向的结点)；$a[n]:=\Sigma_{n'}h(n')$(n'为指向 n 的结点)，并在每次计算后对 a 和 h 向量正规化(normalized)，在算法结束时，对于用户设定的 c 值，hub 值最大的 c 个结点是集中结点，authority 值最大的 c 个结点是权信结点。

假设这一算法第 i 次执行时计算出 hub 向量和 authority 向量分别是 h_i 和 a_i。Kleinberg 证明了：

定理1　序列 $h_1,h_2,h_3,\cdots$ 和序列 $a_1,a_2,a_3,\cdots$ 分别收敛到特定的值 h^* 和 a^* 上。

定理2　h^* 是矩阵 WW^T 的主特征向量，a^* 是矩阵 W^TW 的主特征向量。

定理2说明了 hub 向量和 authority 向量最后的收敛值，在实际计算中，可以直接用求主特征向量的方法来计算各个结点的 hub 值和 authority 值。

Lempel 和 Moran[4]根据网络随机行走的特点提出了计算 hub 值和 authority 值的 SALSA 方法。所谓网络随机行走是指随机地选择一个网页结点作为当前网页结点，然后在与当前网页结点有链接关联的网页中随机地选择一个作为下一当前结点，如此重复。其有两种选择方式：(a)在有链接指向当前网页的网页结点中随机选择；(b)在当前网页结点链接指向的网页中随机选择。经常由 a 类选择方式访问到的网页结点的 hub 值大；经常由 b 类选择方式访问到的网页结点的 authority 值大。其算法如下：

设 B(i)是有链接指向 i 网页结点的网页结点集，F(i)是 i 网页结点链接指向的网页结点集，|S|表示网页结点集 S 中的结点数。则有：

authority 矩阵 A，其中 $a_{i,j}=\sum_{k,k\in B(i)\cap B(j)}\frac{1}{|B(i)|}\frac{1}{|F(k)|}$

hub 矩阵 H，其中 $h_{i,j}=\sum_{k,k\in F(i)\cap F(j)}\frac{1}{|F(i)|}\frac{1}{|B(k)|}$

网页结点的 hub 值和 authority 值向量 h 和 a 分别是矩阵 H 和矩阵 A 的主特征向量。根据 Ergodic 定理[5]，这样计算出的 hub 值和 authority 值高的结点正是分别容易被 a 类和 b 类随机访问方式访问到的结点。

在 Kleinberg 的计算方法中，结点之间 hub 值和 authority 值的相互影响很大。两个距离很远的结点之间也能通过链接路径相互影响它们的 hub 值和 authority 值，这种现象被称之为相互增强效应(mutually reinforcing)。Lempel 分析：在 SALSA 方法中，结点 hub 值和 authority 值只取决于局域性的网络结构，因此它不会产生相互增强效应。

Kleinberg 的计算方法是通过整个图结构计算网页结点的权值，而 SALSA 方法只根据相邻结点的特性来计算页面权值[6]。其计算的结果也具有一定的差异，使用 Kleinberg 的计算方法的结果中，其页面权值最高的结点趋向于属于一个比较紧密结合的页面结点聚簇内，而使用 SALSA 方法计算

权值高的结点更趋向于分布在不同的页面结点聚簇内。因此，Kleinberg 方法的计算结果更具有主题代表性，而 SALSA 方法的计算结果更具有广泛性。可以根据实际应用的需要来选择不同的方法。

2.2.3 基于所链接网页的重要性评价 Marchiori 在文[7]中介绍了一种利用网络链接及所链接网页的重要性来评价网页重要性的方法。他认为网页的重要性包括两个方面：网页内容的重要性和其链接网页的重要性，即 Information(A)=TextInfo(A)+HyperInfo(A)。$Information_{[k]}(A)$表示从 A 出发 k 距离内的信息重要性(包括 A)。首先考虑在单路径链接中的情况，设 $A_0 \rightarrow A_1 \rightarrow A_2 \rightarrow \cdots \rightarrow A_k$，$A_0, \cdots, A_k$ 都是网页结点，链接重要性 $HyperInfo_{[k]}(A_i)$ 为 F 和 $Information_{[k-1]}(A_{i+1})$ 的乘积，其中，F 是消退因子，$0<F<1$，它表示在链接路径中，下一个结点重要性对上一结点重要性的影响。于是：$Information_{[k]}(A_0)=TextInfo(A_0)+F(TextInfo(A_1)+F(TextInfo(A_2)+F(\cdots TextInfo(A_k))))=TextInfo(A_0)+F\ TextInfo(A_1)+F^2\ TextInfo(A_2)+\cdots+F^k TextInfo(A_k)$。这是因为距离 A 结点越远的结点对 A 结点重要性的影响越小。

如果结点 A_0 同时链接到多个结点 $A_1, A_2, \cdots, A_k$，可以这样考虑，用户一般会先点击比较重要的网页结点，因此各个结点的信息重要性对 A 结点的影响是不同的。假设 $TextInfo(A_1) \geqslant TextInfo(A_2) \geqslant \cdots \geqslant TextInfo(A_k)$，则 $Information_{[1]}(A_0)=TextInfo(A_0)+F\ TextInfo(A_1)+F^2\ TextInfo(A_2)+\cdots+F^k TextInfo(A_k)$。

在实际的 Web 链接结构中，可以结合这两种方法来计算网页重要性。例如：在图1中所示的 Web 链接结构中，假设 $TextInfo(B) \geqslant TextInfo(C)$，$TextInfo(E) \geqslant TextInfo(D)$，则选择序列为 A,B,C,E,D，$Information_{[2]}(A)=TextInfo(A)+F\ TextInfo(B)+F^2\ TextInfo(C)+F^3\ TextInfo(E)+F^4\ TextInfo(D)$。

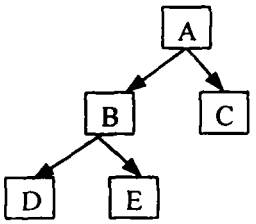

图1

2.3 其他结构信息算法

用户在浏览因特网是通过点击网页上的超链接来访问下一结点的。在 Web 中，网络的链接结构十分复杂，这就是著名的“Lost in Hyperspace”问题。因此人们试图通过网页的链接信息来计算网页的上下文(context)，以帮助用户浏览网络。

Mukherjea 和 Foley 提出了一种利用 Web 链接结构计算网页上下文的方法[8]。其方法找出网络中的某些重要的路标(Landmark)结点，其他网页结点可以通过从某路标经过的链接来标识。路标结点是根据 importance 值决定的，importance 值的计算方法如下：

$$importance = (I+O) * wt_1 + (SOC+BSOC) * wt_2$$

I，O 分别是结点扇入链接数和扇出链接数(InLink 和 OutLink)，SOC 是从本结点出发两步内可以达到的结点数，BSOC 是在两步内可以达到本结点的结点数。wt_1 和 wt_2 是权值，$wt_1+wt_2=1$。为了计算结点的上下文，他采用了一个算法将每一个结点的路标设立在其附近具有最大 importance 值的路标结点上，而这个结点的上下文就是从此路标结点到本结点的路径。

Pirolli 等在文[9]中使用了网页结点的扇入链接和扇出链接数以及其他文档信息来计算文档的类别。他将网页文档分为几种类别。对每个网页计算其网页大小、扇入链接数、扇出链接数、用户点击频度等几个特征，并将其组织成这一网页的特征向量 v。不同的网页文档种类对不同的网页特征有不同的权值 w_{ij}，例如，索引类型的网页的扇出链接数比较多，因此这一对应权值比较高。这些权值组成矩阵 W。算法计算网页的类型向量 $c=W*v$，并可以根据 c 来判断网页的类型。一个网页可以同时属于多种类型。Pirolli 使用这种方法来从 Web 链接结构信息中发掘出网页的隐含信息。

3 Web 链接结构信息应用

从 Web 链接结构图中发掘出的 Web 链接结构信息可以应用在页面评价、主题划分、主题提取以及相关网页寻找等领域。

3.1 网络搜索结果的页面评价

目前，人们对网络信息的查找一般是依靠网络搜索引擎进行的。用户在搜索引擎中输入自己感兴趣内容的关键词，网络搜索引擎在自己的索引数据库中搜索相关的网页，将符合条件的网页返回给用户。随着 Web 规模的不断增长，网络搜索引擎所索引网页的数量不断增加。而调查显示，用户趋向于输入简单的关键词进行查询，这就导致查询结果的数目往往十分巨大，用户还需在众多的结果中查找自己所需要的内容。因此，如果能对这些查询结果进行排序，将重要的网页首先显示，那么将提高网络搜索引擎的查询性能。

Google 搜索引擎[10]由斯坦福大学开发，它是被广泛应用的著名搜索引擎。其中就使用 PageRank 值对网页根据其重要性进行排序(通常称为网页评价：Rank Page)。其搜索引擎在下载网页、进行分析索引的同时，还提取网页中的链接结构信息，计算网页的 PageRank 值。在查询时，根据网页的 PageRank 值对查询结果进行排序。这是 Google 搜索引擎的一个重要的特点。

此外，还可以通过计算网页结点的 hub 值和 authority 值来评价网页结点的重要性。与 PageRank 值不同，其反映了两类不同的重要性。hub 值高表示此网页是很好的索引网页，可以在此找到重要网页的链接；authority 值高表示此网页的内容比较重要。因此，这两种评价值可以满足用户两方面的需求。

3.2 主题划分

彼此联系紧密的网页结点趋向于属于同一主题，因此可以根据网络信息结构，在彼此联系紧密的网页中获取它们的主题信息。

HyPursuit 搜索引擎[11]是 Weiss 研究开发的层次性网络搜索引擎。HyPursuit 的体系结构为一个树型结构，页结点是存储文档的信息库，而其他结点可以看作为路由模块，其被称之为 Content router，一个 Content router 表示一个网页结点聚簇，这些网页结点相关性较大，因此它们趋向于属于同一主题。这些 Content router 组成层次结构，上层的 Content router 的主题包含其子孙 Content router 的主题。HyPursuit 采用聚簇的方法来计算所索引的网页属于哪些 Content router。

HyPursuit 将网络结构信息和页面内容信息结合，以此来计算页的相似性，并根据这一相似性来计算页面的聚簇。其算法使用的链接相似性信息包括三个方面：S^{spl}，S^{anc}，S^{dec}。$S^{spl}(i,j)$表示在网页结点 i 和 j 之间的距离特性，两个结点之间的距离（两个网页结点之间的最短路径长度）越短其相似性越大。$S^{anc}(i,j)$，$S^{dec}(i,j)$分别表示其 i，j 结点的共同祖先结点和共同后代结点的特性，其共同祖先和共同后代的数目越多，其值越大。其链接相似性 $S^{links}=W_d*S^{spl}+W_a*S^{anc}+W_a*S^{dec}$。$W_d$、$W_a$ 和 W_a 表示各自的权值。其算法还使用了对网页文档内容的相似性比较算法来计算两个文档之间的相似度 S^{term}。然后将这两种相似性结合起来计算网页结点之间的相似度。这一相似度就相当于网页文档之间的“距离”，系统根据所有文档之间的“距离”信息来对文档进行聚簇，彼此之间距离短的网页文档趋向于属于同一聚簇。

3.3 主题提取

在利用网络搜索进行查询时，用户趋向于使用简单的关键词进行查询。这样返回的查询结果中可能包含多个主题，因此需要寻找用户感兴趣主题所包含的相关网页。这种寻找特定查询主题相关网页文档的过程，我们称之为主题提取（topic distillation）。

Kleinberg 在文[3]中提出了一种主题提取方法，首先根据查询结果以及这些网页的相邻网页生成网络结构图，然后计算每一个网页结点的 hub 值和 authority 值，同时被重要的 hub 结点指向的网页结点趋向于属于同一主题，而同时指向重要的 authority 结点的网页结点也趋向于属于同一主题。这样就可以根据网页结点的 hub 值和 authority 值进行主题提取。

K. Bharat 和 M. R. Henziger 利用网络结构信息进行主题提取[16]，其中也采用了计算 hub 结点和 authority 结点的方法。不过他们提到了原算法在实际应用中的缺陷。其中最重要的一个缺陷是在由链接组成的回路内，其计算的数值会不断相互影响。因此改进的算法在递归计算时引入权值，计算其加权和，其算法是收敛的。在这一主题提取的方法中，Bharat 等还辅助使用了文本内容的相似性来计算结点之间的关系，同时删除一些在网络结构图中非相关的结点。

Chakrabarti 等对 Kleinberg 的方法进行了扩展，设计开发出了应用于主题提取的 CLEVER 系统[12]，其中他针对实践中遇到的问题对算法作了特殊的改进。

3.4 相关网页寻找

传统的网络信息搜索方法是由用户向网络搜索引擎发送关键词来进行搜索，此外还有另一种搜索方法，即用户提供一个关于其感兴趣主题的网页，搜索程序在网络上找到这一网页的相关网页（related pages），这些相关网页和原网页属于同一主题，但是它们并不一定在内容上一致。例如：给定某种报纸的网页，系统据此找出其他一些报纸的网页地址。

J. Dean 和 M. R. Henziger 利用网络结构信息来寻找相关网页[13]。他们设计了两个算法：Companion 算法和 Cocitation 算法。Companion 算法利用计算 hub 结点和 authority 结点的方法在网络中寻找相关的网页。这一算法首先加入从输入网页出发 k 步可以到达的网页以及 m 步可以链接到输入网页的网页，生成网络结构图；然后删除重复的网页结点（两个网页之间95%的链接相同）；接着为每条边设置权限；最后计算网页结点的 hub 值和 authority 值。这一计算方法和 Kleinberg 的方法相似，只不过在计算时加入了边的权值。最后算法返回指定一个 authority 值最高的网页结点作为输入网页的相关网页。Cocitation 算法考虑结点之间拥有共同父结点的多少（如果结点 i 有一条链接指向结点 j，则结点 i 称为结点 j 的父结点）。如果两个结点有共同的父结点，则称之为 co-cited，其共同父结点的个数称之为 co-citation 度数。Cocitation 算法在输入网页的相邻网页结点图中寻找和输入结点 co-citation 度数大的网页结点作为相关网页。

在浏览器软件 Netscape Communicator V4.06中有一项“寻找相关”（“what's related”）功能，其可以寻找相关网页，其算法综合利用了链接结构信息，网页使用信息，网页内容信息这几类信息来寻找相关网页。

总结 在对网络信息的获取和发掘的研究中，人们发现单纯依靠对网页文档信息的分析是不够的。这是因为对网页内容的分析计算是很耗费计算机处理时间的，而 WWW 上的网页量又十分巨大。并且由于网页上的内容一般是自然语言，因此对其处理结果也不是十分精确。

Web 链接结构是指由网页结点和网页之间超链接边构成的图结构，利用 Web 链接结构可以发掘出许多有用的信息。例如：网页结点的聚簇性信息以及网页结点的重要性信息。同时对 Web 链接结构图获取和计算的耗费[14]也相对较小。因此人们在网页评价、主题提取、相关网页寻找等领域都引入了使用链接结构的分析方法。

但是，Web 链接结构信息也是非精确性的。在实际情况中，在一个聚簇内的网页结点并不一定就是属于同一主题的；它们也可能同属于一个大范围的主题，而各自属于相对小范围内的主题。在链接中也可能出现一些不相干的页面链接，例如在网页上可能会有商业广告链接，而这些链接与网页内容完全不相干，我们称之为“噪音”信息。另外，在一个好的 hub 结点指向的结点之间也很有可能内容不相关，它们可能分属于不同的主题。因此，单纯地对 Web 链接结构信息进行分析具有很大的偏差。

在实际应用中，提高对网络信息分析精度有两种方式：(1)在网页信息分析时加入 Web 链接结构信息分析，例如：Chakrabarti 等在文[15]中就结合网页的超链接信息和文本信息对网页进行分类。(2)在主要依靠 Web 链接结构信息分析的应用领域中，对 Web 链接结构信息的发掘必须在某种程度上加入对特定网页信息内容的分析，以提高其分析的精确度。例如：在分析中引入网页文档的标题信息和超链接上的标题信息。Bharat 等在进行主题提取时对 Kleinberg 的方法进行了扩展[16]，在其中利用到了网页内容信息。他将此方法与 Kleinberg 方法执行的结果进行了比较，证明其结果的精确度确实提高了。

我们认为，综合利用 Web 链接结构信息和网页信息，是目前为止提高 Web 信息分析精度和速度的有效途径，也是值得研究的课题方向。

参考文献

1 Botafogo R A, Shneiderman B. Identifying Aggregates in Hypertext Structures. In: Third ACM Conf. on HyperText (1991. San Antonio, TX), ACM, 1991

2 Page L, Brin S, Motwani R, Winograd T. The PageRank Citation Ranking: Bringing Order to the Web, Manuscript in progress, 1998

3 Kleinberg. Authoritative sources in a hyperlinked environment. In: Proc. of 9th ACM-SIAM Symposium on Discrete Algorithms, 1997

（下转第138页）

意一个支持向量，$\vec{x}^*(-1)$为相应的属于第二类的任意一个支持向量。

(3)检查 $m_k \leqslant m$ 和 $F_k(\xi) \leqslant F(\xi)$是否成立。其中，$m_k$ 为第 k 次学习中错误分类的训练样本数目。$F(\xi)$为第 k 学习中得到的最小值。

(4)如果步(3)中的两个不等式均成立，则停止学习过程。如果任意一个不等式不成立，则标记出分类错误的点，寻找出各个子区域的紧互对，并对错误分类的点比较多的子区域计算出分段节点。对划分后的各个子区域 $k=k+1$，转到(2)。

由上面的算法可知，利用该算法可以很好地控制经验风险，下面将证明该算法也很好地控制了结构风险。从而也就很好地控制了期望风险限。

证明：根据定理1：

$$h \leqslant \min([\frac{R^2}{\Delta^2}], V)+1,$$

成立。对任意 $R_n \geqslant 0, n=1,\cdots,N_k$，又有不等式：

$$\sum_{n=1}^{N_k} R_n^2 \leqslant (\sum_{n=1}^{N_k} R_n)^2 \approx R^2$$

成立。这里 $R_n \geqslant 0, n=1,\cdots,N_k$，表示各个子区域的半径。$N_k$ 为第 k 次学习中子区域的数目。又因为，对所有的子区域 $\Delta \leqslant \Delta_n$，所以 $h \geqslant \sum_{n=1}^{N_k} h_n$，因此这个算法的期望风险得到了有效控制。

从算法及其证明可知，这种算法相对于 Δ-间隔分类超平面方法而言，可以提高识别正确率。既降低经验风险，相对于核函数方法而言，可以降低结构风险又能降低模式识别系统的复杂度，也避免了核函数方法对多样本系统不能有效处理的问题。所以采用本文提出的方法，我们可以比较好地限制期望风险，从而构造出好的学习系统。

结论 本文结合支持向量和邻域法思想提出了基于支持向量的分段线性的学习算法。这种算法克服了传统的分段线性的模式识别方法由于不能控制学习系统的结构复杂性，而可能出现的过学习现象，也避免了现有的基于支持向量的学习算法不能较好控制经验风险或较难确定好的核函数的缺点。另外，即使对于本质非线性系统也有助于降低对所引入的核函数的要求和降低学习系统的结构复杂性，并通过理论证明这种算法是正确的。

本文提出的算法，在构造性控制子区域的数量和合理性方面还有待于进一步改进，以适应复杂样本分布的情况。

参考文献

1 边肇祺，张学工. 模式识别. 清华大学出版社，2000

2 Vapnik V N. The Nature of Statistical Learning Theory. New York: Springer-Verlag, 1999

3 Vapnik V N. An Overview of Statistical Learning Theory. IEEE Translations on Neural Networks, 1999, 10(5): 988～999

4 Scholkopf B. Input Space Versus Feature Space in Kernel-Based Methods. IEEE Translations on Neural Networks, 1999, 10(5): 1000～1016

5 Muller K-R. An Introduction to Kernel-Based Learning Algorithms. IEEE Translations on Neural Networks, 2001, 12(2): 181～201

(上接第98页)

4 Lempel R, Moran S. The Stochastic Approach for Link-Structure Analysis (SALSA) and the TKC Effect. In: Proc. of the 9th Intl. World Wide Web Conf. 2000

5 Gallager R G. Discrete Stochastic Processes, Kluwer Academic Publishers, 1996

6 Borodin A, Roberts G O, Rosenthal J S, Tsaparas P. Finding Authorities and Hubs From Link Structure on the World Wide Web. In: Proc. of the 9th Intl. World Wide Web Conf. 2000

7 Marchiori M. The Quest for Correct Information on the Web: Hyper Search Engines. In: The Sixth Intl. WWW Conf. (WWW97), Santa Clara, USA, 1997

8 Mukherjea S, Foley J D. Showing the Context of Nodes in the World Wide Web. In: Proc. of ACM CHI'95 Conf. on Human Factors in Computing Systems, volum 2 of short papers: Web Browsing, 1995

9 Pirolli P, Pitkow J, Rao R. Silk from a Sow's Ear: Extracting Usable Structures from the Web. In: Proc. of 1996 Conf. on Human Factors in Computing Systems (CHI96), Vancouver, British Columbia, Canada, 1996

10 Brin S, Page L. The Anatomy of Large-Scale Hypertextual Web Search Engine. In: Proc. of 7th Intl. World Wide Web Conf. 1998

11 Weiss R, Velez B, Sheldon M A. Hypursuit: A Hierarchical Network Search Engine that Exploits Content-Link Hypertext Clustering. In: Proc. of the 7th ACM Conf. on Hypertext, New York, 1996

12 Chakrabarti S, et al. Experiments in Topic Distillation. In: Proc. of 8th Intl. World Wide Web Conf. 1999

13 Dean J, Henzinger M R. Finding Related Pages in the World Wide Web. In: Proc. of the 8th Intl. World Wide Web Conf. Toronto, Canada, Elsevier Science, 1999.5

14 Bharat K, et al. The Connectivity Server: fast access to linkage information on the Web. In: Proc. of 7th Intl. World Wide Web Conf. 1998

15 Chakrabarti S, Dom B, Indyk P. Enhanced hypertext categorization using hyperlinks. In: Proc. of the 1998 ACM Intl. Conf. on Management of Data (SIGMOD'98), 1998

16 Bharat K, Henzinger M R. Improved Algorithms for Topic Distillation in a Hyperlinked Environment. In: Proc. of the 21st Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval, 1998

17 Zhang D, Dong Y. An Efficient Algorithm to Rank Web Resources. In: Proc. of 9th Intl. World Wide Web Conf. Amsterdam, 2000

18 Rafiei D. What is this Page Known for? Computing Web Page Reputations. In: Proc. of 9th Intl. World Wide Web Conf. Amsterdam, 2000

19 Dwork C, Kumar R, Naor M, Sivakumar D. Rank Aggregation Methods for the Web. In: Proc. of 10th Intl. World Wide Web Conf. Hong Kong, 2001

20 Chakrabarti S, Joshi M, Tawde V. Enhanced Topic Distillation using Text, Markup tags, and Hyperlinks, ACM SIGIR 2001, New Orleans, 2001

21 Bharat K. When Experts Agree: Using Non-affiliated Experts to Rank Popular Topics. In: Proc. of 10th Intl. World Wide Web Conf. Hong Kong, 2001

22 Chakrabarti S, et al. Mining the Link Structure of the World Wide Web, IEEE Computer, 1999.8

23 Mukherjea S. WTMS: A System for Collecting and Analyzing Topic-Specific Web Information. Computer Networks, 2000, 33

24 Spertus E. ParaSite: Mining Structural Information on the Web. In: Proc. of 6th Intl. World Wide Web Conf. Santa Clara, CA, 1997

25 chakrabarti S, Dom B, Raghavan P, Rajagopalan S. Automatic Resource Compilation by analyzing Hyperlink Structure and Associated Text. In: Proc. of the 7th Intl. World Wide Web Conf. 1998