文章编号:1000-582X(2012)06-117-08

视觉与标签信息的 Deep Web 查询页面内容提取

冯 永[a,b],唐 黎[a,b]

(重庆大学 a.计算机学院;b.信息物理社会可信服务计算教育部重点实验室,重庆 400044)

摘 要:提出了一种结合页面视觉信息和标签信息来提取页面内容结构的方法——DVS。DVS 首先通过分析页面的 CSS 样式信息、DOM 树以获得页面的视觉信息和标签信息,初步得到页面的视觉树;然后利用树的路径相似算法,既考虑标签信息又考虑视觉信息来计算树中模块的相似性,对模块进行聚类,最终得到页面的视觉树,即页面的内容结构。DVS 主要的特色在于从视觉信息和标签信息两方面来提取页面的内容结构;采用树形结构表示视觉信息,将分析视觉信息转换成分析"视觉属性"树。实验采用 UIUC 的 TEL 数据集,分别与 WTS 算法、VIPS 算法进行了比较,文中算法可以获得更高的准确性。

关键词:深层网; 内容提取; DOM 树; CSS 样式;视觉树

中图分类号:TP311 **文献标志码**:A

Combining vision information and tag information to extract Deep Web result pages content

FENG Yong [a,b], *TANG Li* [a,b]

(a. College of Computer Science; b. Key Laboratory of Dependable Service Computing in Cyber Physical Society, Ministry of Education, Chongqing University, Chongqing 400044, P. R. China)

Abstract: Extracting content from deep web pages is a challenging problem due to the underlying intricate structures of such pages. A vision and tags based approach (DVS) is proposed. It primarily utilizes the vision information and tag information on the Deep Web result pages to extract the content structure of pages. This approach includes two steps as follows: First, the vision information and tag information are produced by analyzing the Cascading Style Sheet and the DOM Tree to generate an initial visual-tree of the Deep Web result page. And then, the Path Shingle (PS) algorithm is employed, by considering both of the vision and the tag information, and the blocks in the visual-tree are clustered according to the similarity computing result of them to produce the final visual-tree, i. e., the content structure of pages. The innovations of DVS are that it utilizes the vision information and tag information on the Deep Web pages to extract the content structure; and stores the vision information as a tree to tansform the analysis of the vision information to a vision-attribute tree. Experiments are conducted with a large set of Web databases called UIUC's TEL. The experimental results show that the vision and tag based approach has high precision compared with the WTS algorithm and the VIPS algorithm.

Key words: deep web; content extraction; dom tree; cascading style sheet; visual tree

收稿日期:2012-01-09

基金项目:国家自然科学基金资助项目(61103114);重庆市高等教育教学改革研究重点资助项目(112023);中央高校基本科研业务基金资助项目(CDJXS11181164);"211 工程"三期建设资助项目(S-10218)

作者简介:冯永(1977-),男,重庆大学副教授,主要从事互联网信息处理方向研究,(E-mail)fengyong@cqu.edu.cn。

网络中的信息量已超过十万 TB 数量级，而且仍以很快的速度增长[1]，最近几年在万维网中，存在着越来越多地可以通过查询接口检索到的 Web 数据库，其数量达到了 25 000 000[2]。Web 中的信息主要通过网页的形式发布，根据蕴含信息的“深度”可以将整个 Web 划分成 Surface Web 和 Deep Web。Surface Web 一般是指可以通过传统搜索引擎搜索的页面集合；Deep Web 则是指不能被传统搜索引擎搜索到的页面集合，其页面主要通过查询接口动态从数据库中生成，所有 Web 数据库组成了 Deep Web 数据源。Deep Web 的内容质量比 Surface Web 高，而且相对于传统的 Surface Web 蕴含着海量的、异构的、面向领域应用的、并且不断动态变化的有用信息和数据，因此，针对 Deep Web 的研究是很有必要的。图 1 就是一个典型的来自于 shopping. yahoo. com 的 Deep Web 页面。

在大多数信息检索系统中，都把一个 Web 页面作为最小检索单元，然而，从页面内容角度来说，把一个页面当成一个最小的单元却并不恰当。如图 1 所示，一个页面中往往包含多种内容：导航信息、目录、广告、查询接口、查询结果信息、网站说明信息等，它们分布在页面的不同区域，代表着不同的主题。因此，发现并识别页面的内容结构是相当有必要的，它可以有效的提高页面信息检索的效率。

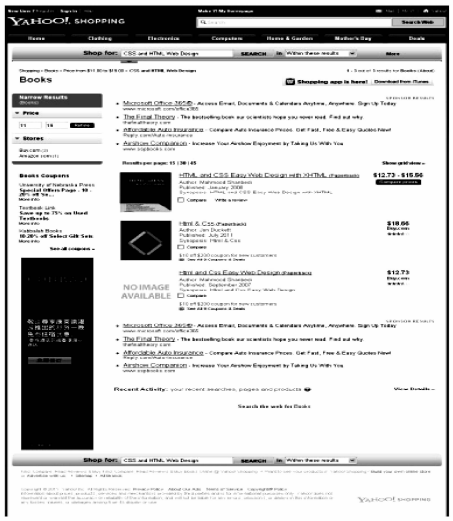


图 1 Deep Web 页面例子

另外，对于 Deep Web 查询结果页面，它比 Surface Web 页面更具有结构性，为了便于用户浏览和展现网站的特色，从同一个 Web 数据库中生成的页面往往具有相同的页面结构：导航栏、目录、广告等装饰性信息，以及页面核心内容，它们在页面上呈现的区域都是相似的，具有固定的页面内容结构，因此，对 Deep Web 结果页面进行内容结构提取能大大提高对 Deep Web 查询结果页面的信息检索，对于语义 Deep Web、Deep Web 的数据提取都有很大的意义。关于 Deep Web 的研究，主要集中在 Deep Web 数据源相关性、可信性分析、语义 Deep Web 、Deep Web 数据提取、Deep Web 查询接口地识别、提取以及查询接口地分类[4-10]。很多研究者利用视觉特征来分析页面，例如在文献[11]中，作者提出了一种利用 HTML-tags 所表现出来的视觉效果来比较不同页面之间的相似度；文献[6] 通过从视觉块树中得到数据区域，提取出数据区域，然后利用位置、布局、外观等视觉特征对数据区域的数据记录和数据项进行分析。至于对页面的视觉树是如何生成的，在此文献中并没有提及。文献[12]的作者提出了一种完全不依赖 HTML-tags 的视觉特征页面内容结构提取算法，把整个页面看成一个视觉块，通过发现视觉间隔符递归地对视觉块进行分块，得到视觉树，最终得到整个页面的内容结构。

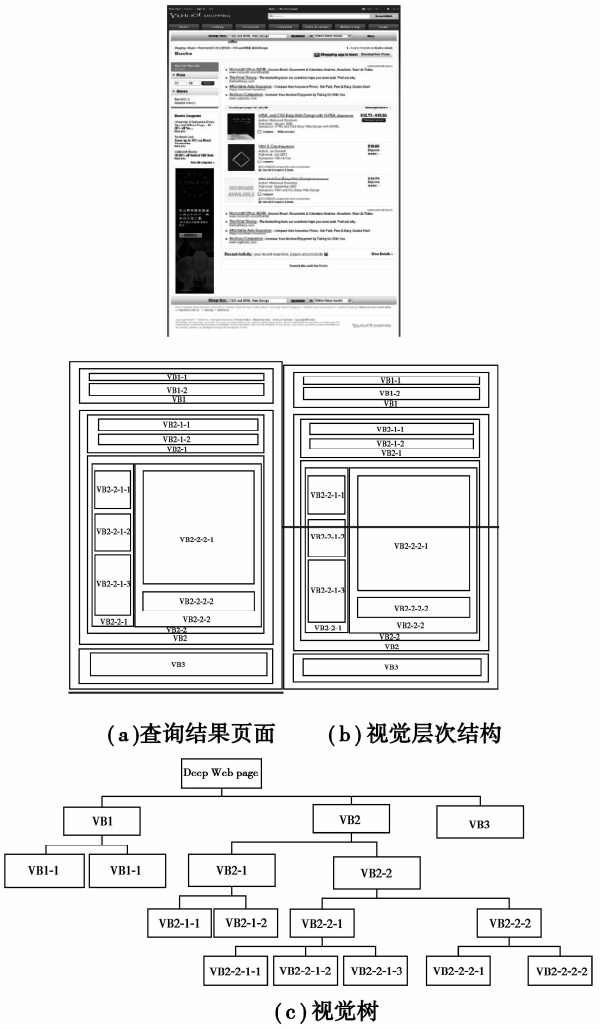
采用 DOM 树和视觉特征相结合的方法来对 Deep Web 结果页面进行分析，从主观和客观 2 方面考虑来分析页面。通过分析 HTML 标签来得到 DOM 树。论文通过分析 CSS 样式来提取页面的视觉特征。这样既从客观又从主观上分析页面，更能有效地提取出 Deep Web 查询结果页面的内容结构。

1 基于 DOM 树和视觉特征的 Deep Web 页面内容结构

页面是承载信息的载体，对于想要提取的信息一定是可视化的。Deep Web 数据库中的信息是以查询结果页面的形式，通过浏览器呈现给用户。对于用户来说，通常首先排除页面的导航栏、目录、广告等装饰信息部分，直接阅读页面的中心内容部分。相对于 Surface Web 页面，Deep Web 页面更具有结构特征。类似于文献[12]，把 DOM 树不可再分的叶子节点叫为基本对象，把视觉结构中的每一块称为视觉块，一个视觉块由一个或多个基本对象组成。

用一棵树来表示一个页面，如图 2 所示，对于 Deep Web 查询结果页面图 2(a)其视觉层次结构如图 2(b)，将其转换成视觉树结构如图 2(c)所示，视

觉树的一个节点表示一个内容块,树中节点的关系表示页面中内容块之间的关系,其中树中节点的兄弟关系表示内容块的平行关系,树中节点的父子关系表示内容块之间的包含关系。



(a)查询结果页面　　(b)视觉层次结构

(c)视觉树

图 2　页面的层次结构和视觉树

2　Deep Web 页面内容结构的构建

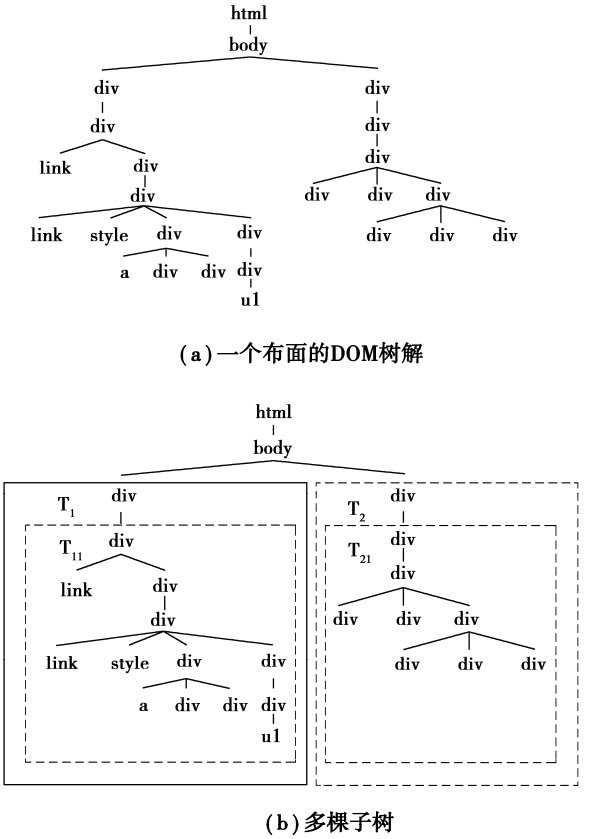
对于 Deep Web 查询结果页面的内容结构提取主要包含以下 3 步,DOM 树的分析、CSS 样式信息的分析、结合 DOM 和 CSS 样式信息提取页面的内容结构。下面将详细对每一步进行介绍。

2.1　DOM 树

对于每个 Deep Web 查询结果页面都对应着一棵 DOM 树,在此将这棵树定义为 Tag-Tree,一个 Tag-Tree 由一个或多个标签组成。

定义 1:一个 Deep Web 查询结果页面 Ω,可以由一棵标签树 T 表示,Ω: $T=\{T_1, T_2, T_3\cdots, T_n\}$,其中 $T_1, T_2, T_3\cdots, T_n$ 叫作 T 的子树,它们分别也是 Tag-Tree,分别又由它们的子树组成。

下面举例说明定义 1,如图 3 所示。图 3(a)为一个页面的 DOM 树 T,这棵 DOM 树又有多棵子树如图 3(b),$T=\{T_1, T_2\}$。



(a)一个布面的DOM树解

(b)多棵子树

图 3　Tag-Tree 的例子

通常情况下,并不是 DOM Tree 的每个标签都代表一个视觉块,在此,根据标签在浏览器上是否可视化,将其分为 Visible-Tag 和 Invisible-Tag。

定义 2:Visible-Tag:标签的效果可以在浏览器上表现出来的标签,如:<tr>,<td>, ,<h_1>,<h_2>…<h_6>等;

Invisible-Tag:标签的效果不可以在浏览器上表现出来的标签,如:<! -- -->,<script>等。

在文中首先对 Deep Web 查询结果页面中的所有标签进行了处理,去除了 Invisible-Tag,其次,也去除了页面中的空文本标签。

对于 Visible-Tag,它们自身就代表了页面的视觉效果,表 1 中给出了详细的 HTML 标签和 Visual 标签的对应关系。其中 grp 表示由多个元素组成的块元素,row 表示行元素,col 表示列元素,test 表示页面中包含的具体数据。

表1 html标签和visual标签的对应关系

HTML标签	Visual标签
html,body,table,div,p,h_1,h_2,h_3,h_4,h_5,h_6, ul,ol	grp
tr,li	row
td	col
其他	text

2.2 CSS样式信息

CSS(cascading style sheet)是用于控制网页样式并允许将样式信息与网页内容分离的一种标记性语言,它决定浏览器如何描述HTML元素的表现形式。CSS可以灵活地控制具体的页面外观,从精确的布局定位到特定的字体和样式。因此,页面的具体外观视觉信息都可以从CSS中得到,在文中用到的视觉信息包括页面布局、颜色、字体。

1)页面布局

根据浏览器显示页面的原理,页面上的信息一般包括文本信息(纯文本和超文本)和图像信息(包括图片、视频、动画等)。对于这些信息都是按照一个特定的像素坐标和大小显示在浏览器上,这就是页面布局。对于整个页面来说,页面布局有其自身的特征,下面具体列出其特征。

布局特征:

①页面导航信息通常处于整个页面的顶端;

②网站版权、联系方式等信息一般位于整个页面的顶部;

③页面"rich-content area"一般位于整个页面的水平中心位置;

④页面"rich-content area"的大小通常要比页面的其他区域要大。

页面设计者在设计页面时,一般把页面的主要内容放在整个页面的中心,比较明显的位置,以吸引用户的注意力。通过对大量Deep Web页面地观察和分析,发现在页面布局方面,存在以上归纳的几点特征。图4给出了一个Deep Web页面布局的例子说明。从图中可以看出,在整个页面布局中VB2-2-2-1不管是在位置方面还是在大小方面都比页面其他部分明显,因此,可以判断出,对于这个页面的"核心内容域"应该就是VB2-2-2-1部分。

2)颜色

颜色特征是一个重要的页面视觉特征。为了吸引人们的注意以及更好的表现网页内容,网页设计者通常使用不同的颜色来表现,主要表现在以下2个方面。

颜色特征:

①对于整个页面,不同视觉块具有不同的背景颜色;

②对于某个视觉块,通常采用的某些明显的颜色以突出重要信息;

如图4所示,对于整个页面,VB1和VB2就具有不同的背景色。对于VB2-2-2-1中的每条数据记录,设计者采用了不同的颜色,用蓝色来突出书名和单价这些重要信息。

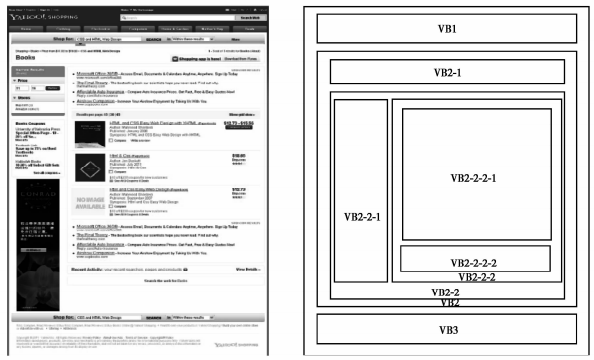


图4 Deep Web页面布局例子

2.3 内容结构提取

内容结构是指Deep Web结果页面的整体组织结构。内容结构提取的目的是从Deep Web结果页面中识别并提取内容块(视觉块)并将所有的内容块(视觉块)在页面的组织结构也提取出来。以树型结构来表现各内容块之间的关系,即视觉树Visual-Tree。图5给出了内容结构提取的整体流程,主要包括2个阶段。

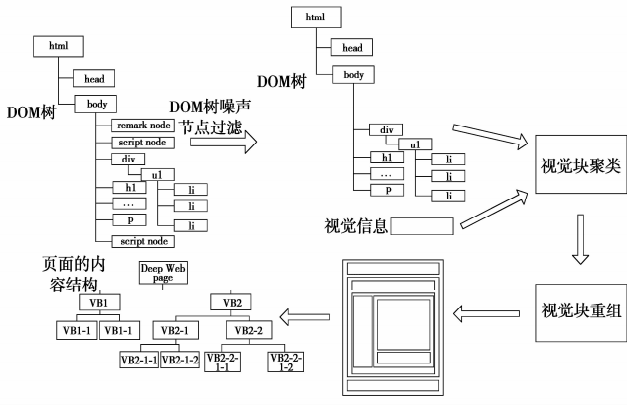


图5 内容结构提取的整体流程

第一阶段,DOM树噪声节点的过滤,将过滤后

的 DOM 树和视觉信息相结合,形成最初的视觉树;

第二阶段,通过计算视觉块之间的相识度对视觉块进行聚类处理,形成最终的视觉树,得到页面的内容结构。

1)DOM 树噪声节点过滤

使用 html parser 来得到 Deep Web 结果页面的标签,通过对这些标签进行处理,得到页面的 DOM 树。在一棵 DOM 树中并不是所有的节点都是有用的,如注释信息(remark node)、JavaScript 信息(script node)、空节点(empty node)等,这些信息对于提取视觉块无用,在此将这些信息称为噪声节点。对于这些噪声节点,在解析 html 时,首先判定出这些节点,然后去掉这些节点,形成一棵去除噪声节点以后的 DOM 树。

2)视觉块聚类

通常情况下,并不是每个 DOM 树标签都代表一个视觉块,对于有些标签,需要多个标签组合才表示一个视觉块。因此,将过滤后的 DOM 树作为形成视觉树的雏形,结合视觉信息对 DOM 树中的节点进行聚类处理,采用块之间的相似性来对其进行聚类处理。

通过对大量的页面统计分析发现①不同的 DOM 树标签可以表现同样的视觉效果,如图 6 所示,图中(a)和(b)表现的是同样的视觉效果,但是它们的 html 代码分别如(c)(d)所示;②页面设计者在表示同类信息时,使用同样的标签表现同一视觉效果。如图 7,对于书目(1)(2)(3)它们都包含书名、作者、出版时间等信息,对比它们的 html 代码发现它们采用了同样的 html 代码。

(a)页面1

(b)页面2

```
<html>
 <head>
  <title>books1</title>
 </head>
 <body>
  <tr><td>
   <table border="0"><tr><td>
    <img src="E:\2.jpg"/>HTML Parser<br/>
    <img src="E:\2.jpg"/>HTML&&CSS<br/>
    <img src="E:\2.jpg"/>CSS Design<br/>
    </td></tr></table>
  </td></tr>
  </body>
</html>
```

(c)页面1的html代码

```
<html>
  <head>
   <title>books</title>
  </head>
  <body>
   <ul>
    <li>HTML Parser</li>
    <li>HTML&&CSS</li>
    <li>CSS Design</li>
   </ul>
  </body>
</html>
```

(d)页面2的html代码

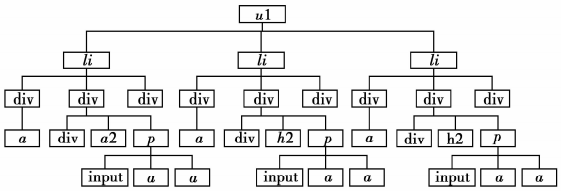
图 6　相似页面的例子

(1)书目1

(2)书目2

(3)书目3

(a)书目的html代码



(b)相似代码

图 7　相似 html 代码例子

基于上面普遍存在的 2 种情况,在计算模块 b_1 和块 b_2 的相似度时既考虑标签信息又考虑视觉信息,在此命名为 DVS 算法,具体公式如下

$$\mathrm{sim}(b_1,b_2)=\omega_t \mathrm{sim}T(b_1,b_2)+\omega_v \mathrm{sim}V(b_1,b_2), \tag{1}$$

其中:$\mathrm{sim}T(b_1,b_2)$,$\mathrm{sim}V(b_1,b_2)$分别表示模块 b_1 和 b_2 块之间 html 代码的相似度,外观相似度;ω_1,ω_2 分别表示它们的权值。

采用路径相似算法(PS)[13] 公式(2)来计算 $\mathrm{sim}T(b_1,b_2)$,并与标签权重相似算法[14] 公式(3)进行了比较。

$$\mathrm{sim}T(b_1,b_2)=\frac{|S(b_1,\omega)\cap S(b_2,\omega)|}{|S(b_1,\omega)\cup S(b_2,\omega)|}, \tag{2}$$

s_1 是一串标签序列,它表示模块 b_1 中,根节点到任意节点的路径;ω 表示滑动窗口的大小;$S(b_1,\omega)$表示子标签序列集合;

例如下面 s_1, s_2 标签序列,

$$s_1=(\mathrm{div},\mathrm{table},tr,td,h1,tr,p),$$

$$s_2=(\mathrm{div},\mathrm{table},ul,li,tr,td,h1,tr,p),$$

$\omega=1$,其子序列集合为

$$S(b_1,1)=\{(\mathrm{div}),(\mathrm{table}),(tr),(td),(h1),(tr),(p)\},$$

$$S(b_2,1)=\{(\mathrm{div}),(\mathrm{table}),(ul),(li),(tr),(td),(h1),(tr),(p)\}$$

s_1 和 s_2 的相似度为77.8%。

$\omega=2$,其子序列集合为

$S(b_1,2)=\{(\text{div},\text{table}),(\text{table},tr),(tr,td),(td,h1),(h1,tr),(tr,p)\}$;

$S(b_2,2)=\{(\text{div},\text{table}),(\text{table},ul),(ul,li),(li,tr),(tr,td),(td,h1),(h1,tr),(tr,p)\}$,

s_1 和 s_2 的相似度为55.6%。

$$\text{sim}T(b_1,b_2)=\frac{\sum_{k=1}^{n}2\cdot\min(\omega_{1k},\omega_{2k})}{\sum_{k=1}^{n}(\omega_{1k}+\omega_{2k})},\quad(3)$$

式中:T_1 表示模块 b_1 的所有标签集合;T_2 表示模块 b_2 的所有标签集合。t_{1k} 表示集合 T_1 中的一个元素,ω_{lk} 表示标签 t_{1k} 在模块 b_1 中出现的次数;t_{2k} 表示集合 T_2 中的一个元素,ω_{lk} 表示标签 t_{2k} 在模块 b_2 中出现的次数;n 表示集合 T_1、T_2 中出现的不同标签总数。

通过对多个页面的视觉信息分析发现,页面的各个部分的视觉效果并不是完全独立的,它们之间存在着继承和兄弟的关系,因此在采用树型的结构来表示整个页面的视觉信息,将这个树叫为 Visual-Attribute 树(VA 树)。因此,可以用公式(2)来计算 $\text{sim}V(b_1,b_2)$ 的值。与 DOM 树的每个节点只存在标签名不同,该树的每个节点存在多个信息,如 position、width、height、color、font 等视觉信息。

3 实验及结果分析

首先介绍实验所使用的数据集;然后评估所用的算法 PS,选择 WTS[14] 与研究的相似性算法比较;最后,选择 VIPS 算法[12] 与所提出的内容提取算法像比较。

笔者采用 UIUC 中的 TEL 数据集,(http://metaquerier.cs.uiuc.edu/repository/),该数据集包含 447 个 Deep Web 数据库,涉及 8 个领域。选择了其中 200 个 Deep Web 数据库,对于每个数据库提交了 10 个查询条件,相应的得到了 10 个 Deep Web 查询结果页面。这些 Deep Web 查询结果页面全部都可以在使用的浏览器上正确显示。

3.1 树型相似性算法的实验结果

在 WTS 中有一个相似性阀值 T_s,默认情况下 $T_s=0.8$,就采用默认值,另外,也把 PS 的相似性阀值也设为 0.8。在这部分实验,各自采用了 1 000 对大小从 2 到 200 不等的 DOM 树和 VS 树作为数据集,实验结果如表 2 所示,每个页面产生 DOM 树和 VS 树以及得到 2 个算法的相似性结果的时间不到在 1 s。

从表 2 的实验结果可以得出以下 3 点。第一,PS 不管是在以 DOM 树为对象还是以 VS 树为对象,准确率都要高于 WTS;第二,PS 在不同对象上的稳定性好于 WTS。PS 在 2 种对象上的准确率相差 0.97%,而 WTS 在 2 个对象上的准确率相差 8.93%;第三,PS 和 WTS 在 DOM 树上的准确率差距远小于其在 VS 树上的准确率差距。以 DOM 树为对象时,其准确率相差 1.47%,然而在以 VS 树为对象时,其准确率相差 11.37%。对于上面的实验现象,分析如下:PS 采用路径的方式,即记录了标签信息又记录了树的结构特征,而 WTS 采用标签方式,只记录了标签信息,忽略了树本身的结构特征;对于 DOM 树的节点(标签)类型不多,一般一棵 DOM 树中包含几十个不同的节点(标签),然而对于 VS 树,其节点类型远多于 DOM 树,例如一个以 5 个 CSS 属性组成一个节点的 VS 树,每个属性的不同属性值就是一个不同类型的 VS 节点,VS 树可以有上千不同类型的节点,其环境复杂度远远高于 DOM 树。

表 2 PS 和 WTS 之间的结果比较 %

算法	对象	准确率
PS	DOM 树	96.10
	VS 树	97.07
WTS	DOM 树	94.63
	VS 树	85.70

3.2 内容结构提取的实验结果

首先评估在对不同权值 w_t 和 w_v 下,公式(1)的相似性算法的准确率,然后评估提取出来地页面内容结构的结果,并与 VIPS 算法比较。在本部分实验中,模块相似性阀值仍然设为通用阀值 0.8。图 8 给出了在不同权值 (w_t,w_v) 下,模块聚类准确率的结果。

从图 8 中可以得到一下几点:第一,w_t 和 w_v 在 0.5 附近的时候,其准确率最高并稳定;第二,w_t 和 w_v 相差大于等于 0.2 时,准确率都将降低。单方面的依赖标签信息或者视觉信息,实验结果都不理想,因此从该实验结果更加证明了,采用标签和 CSS 来提取 Deep Web 页面内容结构思路的正确性。基于上面的实验结果,分别选定,$w_t=0.5$,$w_v=0.5$。

在标签信息权值 w_t 和视觉信息权值 w_v 分别为 0.5 的情况下,实验比较了公式(1)分别采用 PS 算

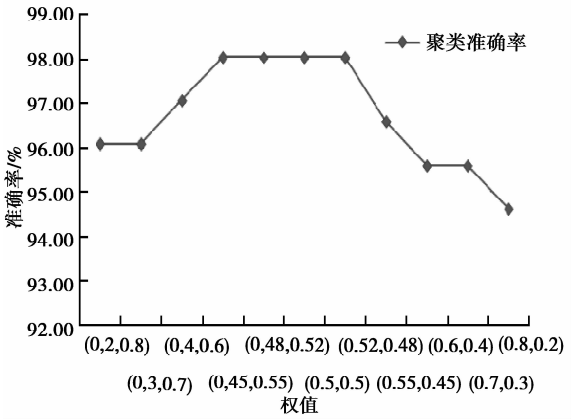


图 8　不同权值比较

法和 WTS 算法计算模块相似性,其对模块聚类准确率的影响。

从表 3 中可以看出,利用 PS 算法来计算模块相似性的准确率高于使用 WTS 算法计算模块相似性的准确率。由于 WTS 算法在环境比较复杂情况下的局限性,导致了其模块聚类准确性不高。PS 算法就不存在此局限性,因此其聚类的准确性远高于 WTS。

表 3　PS 和 WTS 之间聚类结果比较　%

($w_t=0.5, w_v=0.5$)	准确性
PS	98.23
WTS	89.04

下面将实验页面内容结构提取的结果和 VIPS 算法相比较。表 4 给出了实验的比较结果。对于每个 Deep Web 结果页面,实验大概需要 1 s 的时间生成页面的内容结构。

表 4　内容结构准确率比较　%

算法	准确率
研究算法	98.23
VIPS 算法	97.00

VIPS 算法完全不依赖标签,只依赖页面的视觉特征,存在很大的主观性,而且对于不同的浏览器显示有异。然而论文的算法,即从客观又从主观 2 方面分析页面,因此,研究算法的准确性优于 VIPS 算法。

4　结　论

随着 Deep Web 的飞速发展,用户有更多的机会可以从中获得丰富的资源。每一个页面都有其内容结构,提取这些内容结构对语义 Deep Web、Deep Web 的数据提取、信息检索等都有很大的意义。研究结合页面的标签信息和视觉信息来提取页面的内容结构,此外,在实验的过程中发现了页面视觉信息存在的继承和兄弟的关系,基于这种联系,采用树形的结构存储视觉信息,将分析视觉信息转换成分析 Visual-Attribute 树。由于有些网页的设计者并不是按照标准的 CSS 书写标准书写的,对于那些 CSS 书写不规则的页面,有一定的处理难度。因此,下一步工作就是提高 CSS 信息处理的容错性。

参考文献:

[1] FETTERLY D, MANASSE M, NAJORK M, et al. A large-scale study of the evolution of Web pages [J]. Software, Practice and Experience, 2004, 34(2): 213-237.

[2] MADHAVAN J, COHEN S, DONG X L, et al. Web-scale date intergration: you can only afford to pay as you go[C]//Proceedings of the 3rd Biennial Conference on Innovative Data Systems Research(CIDR), January 7-10, 2007. Asilomar, CA, USA: [s. n.], 2007, 7: 342-350.

[3] EGLIN V, BRES S. Document page similarity based on layout visual saliency: application to query by example and document classification [C]//Proceedings of the Seventh International Conference on Document Analysis and Recognition, Aug. 3-6, 2003, Edinburgh, Scotland, UK. Washington, DC, USA: IEEE Computer Society, 2003, 2: 1208-1212.

[4] BALAKRISHNAN R, KAMBHAMPATI S. SourceRank: relevance and trust assessment for deep web sources based on inter-source agreement [C]// Proceedings of the 20th international conference on World Wide Web, March 28-April1, 2011. Hyderabad, India: [s. n.], 2011: 227-236..

[5] HONG J L, SIEW E G, EGERTON S. WMS-extracting multiple sections data records from search engine results pages [C]//Proceedings of the 2010 ACM Symposium on Applied Coputing, March 22-29, 2010. Sierre, Switzerland: ACM, 2010: 1696-1701.

[6] LIU W, MENG X F, MENG W Y. ViDE: a vision-based approach for deep web data extraction [J]. IEEE Transactions on Knowledge and Data Engineering, 2010, 22(3): 447-460.

[7] AN Y J, GELLER J, WU Y T, et al. Semantic deep web: automatic attribute extraction from the deep web data sources[C]//Proceedings of the 22nd Annual ACM Symposium on Applied Computing, March 11-15, 2007. Seoul, Korea: [s. n.], 2007: 1667-1672.

[8] QIANG B H, XI J Q, ZHANG L. An effective schema extraction algorithm on the deep web[C]//Proceedings of the 4th International Conference on Wireless Communications, Networking and Mobile Computing, Oct. 12-14, 2008. Dalian, China: IEEE, 2008: 1-4.

[9] WANG F, AGRAWAL G. Extracting output metadata from scientific deep web data sources[C]//Proceedings of the 9th IEEE International Conference on Data Mining, December 6-9, 2009. Miami, FL, USA: IEEE Computer Society, 2009: 552-561.

[10] LIU W, MENG X F, MENG W Y. Deep web data integration [R]. WAMDM: Technical Report WAMDM-TR-2006-3, 2006.

[11] ALPUENTE M, ROMERO D. A visual technique for web pages comparison [J]. Electronic Notes in Theoretical Computer Science, 2009, 235: 3-18.

[12] CAI D, YU S P, WEN J R, et al. Extracting content structure for web pages based on visual representation [C]//Proceedings of the 5th Asia-Pacific Web Conference on Web Technologies and Applications, April 23-25, 2002. Xi'an, China: [s. n.], 2003: 406-417.

[13] BRODER A Z. On the resemblance and containment of documents[C]//Proceedings of the Compression and Complexity of Sequences, June 11-13, 1997. Positano, Italy: IEEE Computer Society, 1997: 21-29.

[14] BUTTLER D. A short survey of document structure similarity algorithms [C]//Proceedings of the 5th International Conference on Internet Computing, March 5, 2004. Las Vegas, Nevada, USA: [s. n.], 2004: 3-9.

(编辑 侯 湘)

(上接第 84 页)

[10] 史小兴, 金剑. 建筑工程纤维应用技术[M]. 北京: 化学工业出版社, 2008.

[11] 董振英, 李庆斌, 王光纶, 等. 钢纤维混凝土轴拉应力应变试验研究[J]. 水利学报, 2002(5): 47-50.
DONG ZHEN-YING, LI QING-BIN, WANG GUANG-LUN, et al. Experimental study on stress-strain characteristics of steel fiber reinforced concrete under uniaxial tension [J]. Journal of Hydraulic Engineering, 2002(5): 47-50.

[12] 张君, 居贤春, 郭自力. PVA 纤维直径对水泥基复合材料抗拉性能的影响[J]. 建筑材料学报, 2009, 12(6): 706-712.
ZHANG JUN, JU XIAN-CHUN, GUO ZI-LI. Tensile properties of fiber reinforced cement composite with different PVA fibers[J]. Journal of Building Materials, 2009, 12(6): 706-712.

[13] 过镇海, 时旭东. 钢筋混凝土原理和分析[M]. 北京: 清华大学出版社, 2003: 24-33.

[14] 杨萌, 黄承逵. 钢纤维高强混凝土轴拉性能试验研究[J]. 土木工程学报, 2006, 39(3): 55-61.
YANG MENG, HUANG CHENG-KUI. Study on stress-strain curve of high strength steel fiber reinforced concrete under uniaxial tension [J]. China Civil Engineering Journal, 2006, 39(3): 55-61.

[15] 高丹盈, 赵 军, 汤寄予. 掺有纤维的高强混凝土劈拉性能试验研究[J]. 土木工程学报, 2005, 38(7): 21-27.
GAO DAN-YING, ZHAO JUN, TANG JI-YU. An experimental study on the behavior of fiber reinforced high strength concrete under splitting tension[J]. China Civil Engineering Journal, 2005, 38(7): 21-27.

[16] 王起帆, 李洁, 田强, 等. 塑钢混杂纤维轻骨料混凝土力学性能研究[J]. 四川建筑科学研究, 2010, 36(4): 204-208.
WANG QI-FAN, LI JIE, TIAN QIANG, et al. The mechanical properties research of HPP fiber lightweight aggregate concrete[J]. Sichuan Building Science, 2010, 36(4): 204-208.

[17] 王晓飞, 丁一宁. 聚丙烯粗纤维混凝土轴拉性能的试验研究[J]. 混凝土, 2011, 255(1): 81-84.
WANG XIAO-FEI, DING YI-NING. Experimental study on uniaxial tensile properties of macro-PP fiber reinforced concrete [J]. Concrete, 2011, 255(1): 81-84.

(编辑 陈移峰)