Available online at www.elixirpublishers.com (Elixir International Journal)

Network Engineering

Elixir Network Engg. 67 (2014) 21576-21578

ISSN: 2229-712X

Web extraction based on the XPath expanding technique

Song Mei Mei and Li HaiXia

Department of Information and Engineering, Taishan Medical College Taian, 271016 China.

ARTICLE INFO

Article history:
Received: 26 December 2013;
Received in revised form:
24 January 2014;
Accepted: 8 February 2014;

Keywords
Search engine,
Web extraction,
Xpath expansion.

ABSTRACT

Research and Realization of a Web Information Extraction based on Xpath expansion. This algorithm marks importance weight of semantic blocks and obtains a model of block importance. This information extraction method based on Xpath expansion forms extraction rule corresponding to web site. Lastly, this algorithm extracts the text information blocks of theme-based web pages. It uses a case to confirm that this algorithm can accurately complete the extraction task of text information blocks in the theme-based web pages.

Introduction

Information technology not only offers abundant information sources, but also puts forward a challenge on how to quickly and accurately obtain information at the same time. How to identify information the user needed becomes a difficult problem in information extraction. Information extraction technology firstly used by MUC,(Message understanding confrenc) [1,2].Research on Web information extraction started relatively late in our nation, Institute of Data and Knowledge Engineering, renmin university of china schema-guided wrappers in the xwis [3] , Firstly, the user defines a schema, then provides sample mappings between the schema and the HTML page,last the system will induce the mapping rules and generate a wapper. DOM-based information extraction for the web source was presented[4], the algorithm designed a novel approach to semi-automatically rules. Theses methods will generate redundant information and a low precision ratio.

Vision-based Page Segmentation (VIPS)

（VIPS） [5]is also named Vision-based Page Segmentation, Utilizing the visual property of page this technology is applied to sepator page, it combines the technology of information extraction. The algorithm of VIPS is used to extract the visual block which is sticked on a label, the sepator of these node is found. By means of sepator, the page is segmented the block of visual information, then designs content tree structure on the web. This algorithm reflects the content structure of page fairly well, but it has beginning to reach its limits by the font background.

Improvement the algorithm of information extraction based on VIPS

Step 1:According to the algorithm of VIPS, the page of the trained data is splited, all of the content structures are extracted on the web, and the leaf nodes are saved data base.

Step 2: filly use the important information tags and the tag structures on HTML pages, so that can embody the important of the semantic structure , The detailed computing process of the is as follows：

①Analysis of content blocks. In the text content of page blocks,if there is the important information in the content blocks, the importance of the content blocks and the nested blocks will be relatively high. Each content block is weighted according to the conclusion, the transition rule of the weight is put forward. A one-to-one correspondence between the content blocks and the tag tree is established, each tag tree is assigned to one, if the leaf tag is the main content, then accumulate the impact factor, get the impact factor of leaf node, other leaf node is still assigned to one. When the impact factor of leaf node increase more than one, the weight of the leaf node is λ times of the current weight, father node and other nodes become $\sqrt{\lambda}$ times of current weight ,then it sprawls from the father node outwards. The process is shown in Figure 1:

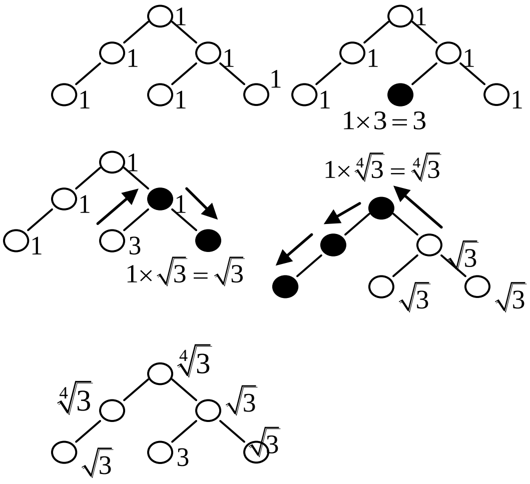

Figure 1 The Process Of Weight Transition

②Attritute Weighting[6]

$$w_i = \frac{\sum_{j=1}^{N} B_j \times Bf_{ij}}{\sqrt{\sum_{i=1}^{n}(\sum_{j=1}^{N} B_j \times Bf_{ij})^2}} \qquad (1)$$

B_j means the weight of the content j, N means total contents, Bf_{ij} means that how many times would keywords i appear on the content j. InnerTexts of the page block express as

Tele:
E-mail addresses: mmsong.shuxue@163.com

the eigen vector. The eigen vector D express as $D=(w_1,w_2,...,w_N)\cdot$

③Computing the degree of similarity between the text of two page blocks. the equation to calculating is[6]:

$$Sim(w_i,w_j)=\frac{\sum_{k=1}^{N}w_{ik}\times w_{jk}}{\sqrt{(\sum_{k=1}^{N}w_{ik}^2)(\sum_{k=1}^{N}w_{jk}^2)}} \quad (2)$$

w_i means the ith weight of the page block, w_j means the jth weight of the page block,N means demention of the eigen vector.

④When the value of Sim is over the value of Doc,we think that the semantics of two page blocks is identical.

⑤Computing the relative position of page blocks

A rectangle is redirect each semantic structure to page , Spatial aspects of each semantic structure include four aspects: {Block-CenterX, BlockCenterY, BlockRectWidth, BlockRec-tHeight}。

⑥Computing the weight of the page block[6]。

Assuming that the page of block B occurs n times,N is the number of page , Btl is the length of text in the page block, Ptl is the overall length of page block text in page block B, （X，Y）is the relative position of page block B, Ltl is the length of anchor text in page block B, the equation to computing the weight is:

$$W(B)=\frac{w_1(1-\frac{n}{N})+w_2(\frac{Btl}{Ptl})+w_3(1-\frac{\sqrt{(X-0.5)^2+(Y-0.5)^2}}{\sqrt{2}/2})+w_4(1-\frac{Ltl}{Btl})}{\sum_{i=1}^{4}w_i} \quad (3)$$

Web extraction based on the XPath expanding technique

Each of page has a main information area,the main information area has many of subdata blocks,each of the subdata block has many of data item.First the data item gain an expression of XPath

Info(p)={B1,B2,...,Bn},BI={ I1,I2,...,Im };

All{Bi(I[R1], I[R2],..., I[Rt])};// [R1,R2,..., Rt];

the FLCS(the forward longest common substring) compairs with Xpath of subdata item,the public part is taken as the Xpath corresponding to the subdata,chosing arbitrarily the Xpath of data,we can gain the Xpath corresponding to the data,which is expressed P,that is the extraction rule R.

Put in :element node of *HTML*

Put out: a string that uniquely identifies a specified node of Xpath expression,

Pseudocode of algorithm:

```
XERG(HTML, I1,I2,…,Im)
PageSource=GetPage(HTML)
      Dom=Tidy(PageSource)
      Pip=GetHtmlElementXpath(Dom,Iip)
      Piq= GetHtmlElementXpath(Dom,Iiq,)
      Pjp= GetHtmlElementXpath(Dom,Ijp)
      Pjq= GetHtmlElementXpath(Dom,Ijq)
      Pi=FLCS(Pip,Piq)
Pj=FLCS(Pjp,Pjq)
AddxPath(string $P_i$)
AddxPath(string $P_2$)
      Int[] diffset=GetDiffrenttable($P_i$,$P_j$)
      For each index in diffset
         Replace index with“*”in $P_i$
      R=$P_i$
      Return string  $P_i$
```

The main information block on the page is regarded as being planar-table,each of subdata block is regarded as being a row,the subdata block of its item is regarded as being a column,the extraction rule is expressed node of Xpath expression based on the row and column,that can extract information. The experimental datas are ten classifications of book web which is from http://www.amazon.cn[7],the results of the XPath as followed:

```
(P11RowXPath)
/html[1]/body[1]/table[5]/tr[3]/td[2]/table[1]/tr[1]/td[2]/a[1]
<Columns>
<Column>
(XPath) table[1]/ tr[1]/A[1]/text()[1] </XPath>
</Column>
<Column>
(XPath) table[1]/ tr[1]/ td[2]/text()[1] </XPath>
<Column>
<XPath> table[1]/ tr[1]/ td[2]/text()[1] </XPath>
</Column>
<Columns>

(P12RowXPath)html[1]/body[1]/table[5]/tr[3]/td[2]/table[1]/tr[2]/td[1]
<Columns>
<Column>
(XPath) table[1]/tr[2]/A[1]/text()[1] </XPath>
</Column>
<Column>
(XPath) table[1]/tr[2]/ td[2]/text()[1] </XPath>

<Column>
<XPath> table[1]/tr[2]/ td[2]/text()[1] </XPath>
</Column>
<Columns>
```

According to the algorithm of FLCS:the Xpath of the first data block is p1, the Xpath of the second data block is p2,by comparing the subscript sequences,we can konow that the fourth subscript sequences is different,we change it as “*”,by this way ,we can gain all blocks Xpath

```
Xpath:/html[1]/body[1]/table[5]/tr[*]/td[1]/table[1]/tr[1]/td[2].
<Columns>
<Column>
(XPath) table[1]/tr[1]/A[1]/text()[1] </XPath>
</Column>
<Column>
(XPath) table[1]/tr[1]/ td[2]/text()[1] </XPath>
<Column>
<XPath> table[1]/tr[1]/ td[2]/text()[1] </XPath>
</Column>
<Columns>
```

Those are the extraction rules R.The experiment result is shown in table:

This paper testified the precision and recall result by the experiment, It can be concluded from above results that the precision rate of data block is prosperously 100%,it was because we use the simple pages. This algorithm uses method of exchanging the data with users, Mining the user extration information, we can obtain an expression of XPath, by the XPath expanding technique, we can get the extraction rule. The experiments indicate the XPath expanding technique are far better than vision-based page segmentation algorithm.

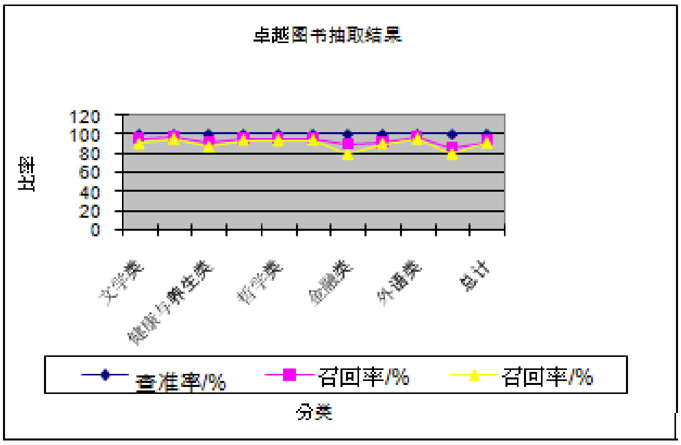

Table 1. The extraction result of amazon books

References

[1] Naney A.Chinchor. Overview of MUC-7/MET-2. In: Proeeedings of the Seventh Message understanding Conferenee, 1998.

[2] Marsh，E.，Perzanowski，D. MUC-7 EVALUATION OF IE TECHNOLOGY: Overview of results. In: Proeeedings of the Seventh Message Understanding Conferenee，1998

[3] MENG Xiao-feng, WANGHa-i yan, GUMing-zhe, WANG Jing. Schema-Guided Wrappers In The Xwis [J]. Computer Applications 2001,21（9）:2-7

[4] LI Xiao-Dong GU Yu-Qing DOM-based Information Extraction for the Web Sources [J]. CHINESE J.COMPUTERS 2002,25(5):526-533

[5] CAI Deng, YU Shipeng, WEN Jirong. VIPS: a vision-based page segmentation algorithm [EB/OL]. [2003-11-01]. http://research. microsoft.com/~ jrwen/jrwen-files/publications/VIPS_Technical 20% Report .PDF.

[6] A Study of Algorithms about Web resource,Page.Cleannig and classifye [D].Zhejiang:The Thesis Of Zhejiang Gongshang University

[7] http://www.amazon.cn/

[8] LIU Wei, YAN Hua-Liang, XIAO Jian-Guo, ZENG Jian-Xun Solution for Automatic Web Review Extraction[J]. Journal of Software 2010,21（12）3231-3235