ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
Available online at www.ijarbest.com

International Journal of Advanced Research in Biology Ecology Science and Technology (IJARBEST)
Vol. I, Special Issue I, August 2015 in association with VEL TECH HIGH TECH DR. RANGARAJAN DR. SAKUNTHALA ENGINEERING COLLEGE, CHENNAI
National Conference on Recent Technologies for Sustainable Development 2015 [RECHZIG'15] - 28th August 2015

The Novel of Web Mining Based On VTD-XML Technology

[1]Mrs.S.Shanmuga Priya, [2]S.Arun Raj, [3]K.V.Karthick, [4]B.Marudhu Pandiyan

[1]ASSISTANT PROFESSOR,[234]UG SCHOLAR, DEPARTMENT OF COMPUTERSCIENCE & ENGINEERING

VEL TECH HIGH TECH Dr.RANGARAJAN Dr.SAKUNTHALA ENGINEERING COLLEGE

Abstract
The rapid development of the Web technology makes the Web mining become the focus of the current data mining, and the XML technology also becomes the standard of the data exchange on the Web. This paper introduces the Web mining technology, implements a frame of the Web mining based on the XML technology according to the advantage of XML in the data description, describes the implementation process of specific Web mining, and puts forward a promoting scheme on solving XML documents with VTD which solves the difficult mining problem on the Web caused by the most of the non-structure information.

Keywords-XML ; Web mining;VTD; non-structure

INTRODUCTION

Along with the rapid development of the Internet, more and more database and information systems Join into the network, network exists large amounts of data. Facing so many complicated Web space, How to find the required information that have become an important problem in the vast network. Although users can rely on search engines quickly, efficiently and accurately to find the related information, to find the user need information is still very difficult. Recent years web data mining is an effective method to solve this problem. Web mining is the application of data mining techniques to extract knowledge from web data, including web documents, hyperlinks between documents, us-age logs of web sites, etc.

EXISTING SYSTEM

The focus of this paper is how to extract data structures based on XML technology from the web pages. Much of the information appeared on the current Web in Hyper Text Markup Language HTML document, Users through the browser to obtain information of these HTML document. HTML document may be written by manual or using HTML tool. Because the HTML document does not aim to automatically extract, but for expressing the information content. Therefore many of the HTML document on the Web is not standardized format, and extracting data is more difficult from the substandard document than the structured document.

Extracting structured data from Web sites is not a trivial task. Most of the information on the Web today is in the form of Hypertext Markup Language (HTML) documents which are viewed by humans with a browser. HTML documents are sometimes written by hand, sometimes with the aid of HTML tools. Given that the format of HTML documents is designed for presentation purposes, not automated extraction, and the fact that most of the HTML content on the Web is ill-formed ("broken"), extracting data from such documents can be compared to the task of extracting structure from unstructured documents.

XML documents with VTD which solves the difficult mining problem on the Web caused by the most of the non-structure information. The Extensible Markup Language (XML) is released by the World Wide Web Consortium (W3C) in Feb,1998 XML over comed the shortcomings of HTML ,standardized the documents on the Internet, gave mark a certain meaning, and reserved the advantages of the HTML- concise, suitable for transmission and browsing.XML Set the advantages of SGML and HTML in a whole, and become the core of the next generation of the Internet. XML have the advantages of scalability, structural, platform independence, self-describing, flexibility, standardability and simplicity

METHODS

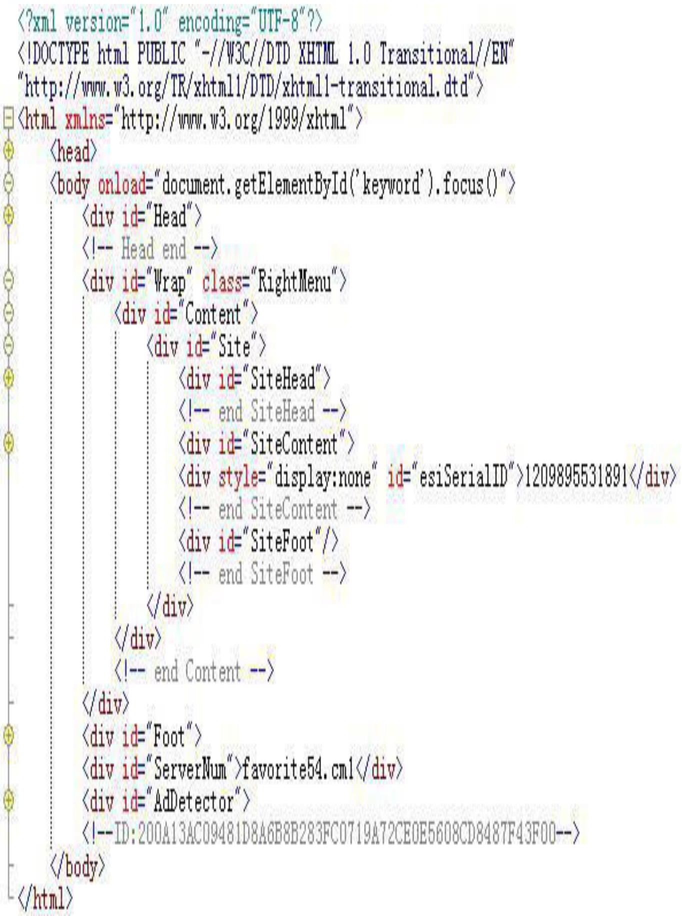
Virtual Token Descriptor for eXtensible Markup Language (VTD-XML) refers to a collection of cross-platform XML processing technologies centered on a non-extractive XML, "document-centric" parsing technique called Virtual Token Descriptor (VTD). VTD-XML is developed by XimpleWare and dual-licensed under GPL and proprietary license. It is originally written in Java, but is now available in C, C++ and C#.The entire framework consists of three modules: data acquisition module, data pre-processing module and data mining module. In data acquisition module, the Web pages are identified through the meta-search engine. In the data pre-processing module, the data can be converted into their corresponding HTML format through the URL in Java, and then converted into XHTML format by JTidy and converted into the corresponding XML documents through XSLT.Then these converted XML documents are integrated. Finally the data extracted from the integrated XML documents are stored to the database through the technology of VTD. In data mining module, the data in the database are pre-processed again to standardized data sets. Then data mining is carried out on them in order to extract useful knowledge

A. Data Acquisition Module

The main task of data acquisition module is to obtain the data source. The source of the data is mainly local data, as well as data on the network. It has a variety of manifestations, which may be text data, as well as the data in the database. Here we can grasp appropriate Web pages as the source of Web mining from Internet through meta-search engine. The example this paper selects is the related information of Shenzhou Pad extracted from Taobao, including trade names, dispensers, price, and the number of collections.

B. Data Preprocessing Module

1) Data Cleaning. The Web page can be converted into HTML format through the URL in Java. This is because HTML is a kind of language that can hardly be handled by program and the main part of Web page is in the layout of the haphazard. For example, the label does not match, the property is lost, and the elements of the nested are bad. All of them are due to the lack of widespread adoption of standards. Therefore, a mechanism must be used to convert HTML pages to well-structured XHTML format so that you can use tools to conduct analysis and processing. JTidy can automatically carry out the necessary changes to make HTML documents' code consistent with the requirements of XHTML, which provides a syntax checker and labeling compensator to repair a mess of HTML to conform to XHTML standards. One class called HTML to XHTML to achieve the conversion from html to XHTML is designed, the key function of which is the procedure of convert (). In this procedure, a file is created to display the error message in the process of conversion by the function tidy. setErrout (new Print (new File Writer (errOutFileName), true)).In the conversion process of this experiment, we mainly solve the problem of garbage as well as the format. JTidy can only deal with English page, but the page is in Chinese, so Chinese characters will be garbled. This problem is due to the non-uniform of the conversion among byte streams. In addition for a simple html page, the conversion is good.

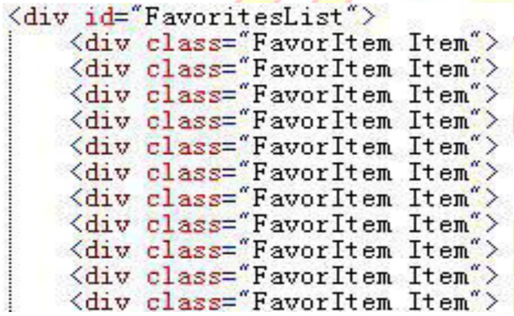


```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
    <head>
    <body onload="document.getElementById('keyword').focus()">
        <div id="Head">
        <!-- Head end -->
        <div id="Wrap" class="RightMenu">
            <div id="Content">
                <div id="Site">
                    <div id="SiteHead">
                    <!-- end SiteHead -->
                    <div id="SiteContent">
                    <div style="display:none" id="esiSerialID">1209895531891</div>
                    <!-- end SiteContent -->
                    <div id="SiteFoot"/>
                    <!-- end SiteFoot -->
                </div>
            </div>
            <!-- end Content -->
        </div>
        <div id="Foot">
        <div id="ServerNum">favorite54.cml</div>
        <div id="AdDetector">
        <!--ID:200A13AC09481D8A6B8B283FC0719A72CE0E5608CD8487F43F00-->
    </body>
</html>
```

However, due to the very large page, the corresponding html file is very complex, so there are many problems in the format of the XML output, such as the "&nbsp", "&reg". So pre-process must be taken before the conversion to XHTML, which converts such marks into the appropriate type of nonentity reference marks. We can use Microsoft's Internet Explorer or Altova's XML Spy to view the effect of conversion, as well as code.

2) Data Conversion

At this time the norm of the XHTML document has been obtained. Then we use the XSL (eXtensible Style sheet Language) to achieve data conversion. The purpose of data conversion is available because of the poor structural XHTML documents. It has been based on the XML syntax structure, but it contains a lot of HTML vocabulary, so we can generate a more structured XML document by XSL. Next, we will extract the related information from the web page of notebook. After a closer analyze on the page, we can find the information in the bottom of the page, that is the label of <div id = "SiteContent"> in below fig. Then expand the label and it will be easy to find data included in <div class = "FavorItem Item"> below <div id ="FavoritesList"> in below fig. The tag of div consists of 40 lines, each of which represents a kind of information ofnotebook.



```
<div id="FavoritesList">
    <div class="FavorItem Item">
    <div class="FavorItem Item">
    <div class="FavorItem Item">
    <div class="FavorItem Item">
    <div class="FavorItem Item">
    <div class="FavorItem Item">
    <div class="FavorItem Item">
    <div class="FavorItem Item">
    <div class="FavorItem Item">
    <div class="FavorItem Item">
    <div class="FavorItem Item">
```

After finding the location of the notebook information, we can use XSLT processor to convert the XHTML document, and output the document in accordance with the specified style sheet. The following task is to create a style sheet in accordance with the XHTML document in order to achieve a smooth conversion. We make the sub-Label as a reference point, and establish a path from the root node to the preference node. Only the elements on the path will be dealt with, then this sub-label can be found and needless information can be filtered out. You can use the following general expression: html / body / div [2] / div / div [2] / div [1] / div / div [5], which decides the path from the root <html> to the parent node of the reference node. Then you can do the matching through the principle of node matching template. <xsl:valueof select="…"> can be used in the result tree to create multiple text nodes, which are the relevant information of notebook, including trade names, price, etc. So that the original XHTML document information is extracted from the notebook, and the other unrelated information is filtered out to generate a more structured XML document.

3) Data Integration.

The purpose of data integration is to organize the data in the system as a whole according to certain rules through a certain means of technique and allow users to operate on data effectively. Since XML documents are so many, a merging method based on XSLT for several XML documents is given. First, a document merge.xml is added manually, the sub-elements of which contain the name of the XML document to be merged. All of these are as follows:

<merge>

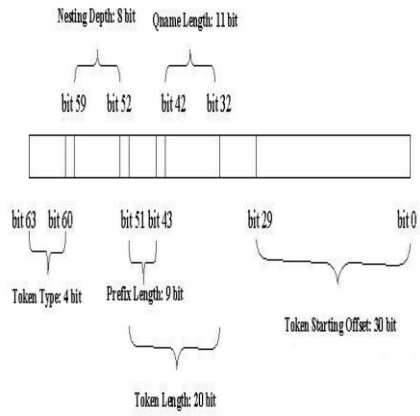
<file>first.xml</file>

<file>second.xml</file>
……
<file>xx.xml</file>
</merge>

Next according to the principle of template matching we define a style sheet for the documents and the corresponding XML document named as test.xsl, and then apply it to the document merge.xml so that it is easy to merge several XML documents. This method can avoid the excessive use of document () and improve the speed of the system. The most important thing is that the combined structure of XML documents is not destroyed, which provides the convenience for the following data mining.

4) Data Extraction.

The purpose of the data extraction is to resolve the contents of the combined XML documents and then store them into database. The technology of VTDXML is used to analyze the XML document. This technology is an XML processing technology without extraction, which describe the markings with the starting offset and length. VTD record is a basic data type (can be integer multiple of 32bits), which records the following parameters information of token: Starting offset, length, Nesting depth and Token type as shown in Fig.



A VTD record uses 64 bits as the basis of numerical type (b63 ~ b0), whose format is as follows [5]:

a) Start Offset: 30 bits (b29 ~ b0), the maximum value: 230-1 = 1G - 1

b) Length: 20 bits (b51 ~ b32), the maximum value: 220-1 = 1M - 1

c) Nested depth: 8 bits (b59 ~ b52), maximum value: 28 - 1 = 255

d) Token type: 4 bits (b63 ~ b60).

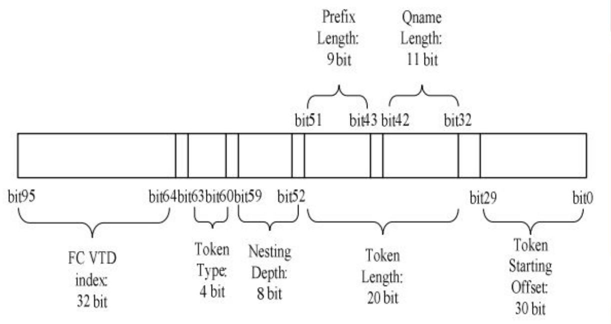
e) retain the bit: 2 bits (b31: b30)

The structure of LC (Location Cache) is used for the traverse of VTD-XML and it's simply that VTD is used to build a tree-like table according to its depth as a standard. The entry of LC is also 64bits long of numerical type, the former 32bits of which are on behalf of a VTD index and the rest 32bits denotes the LC index of VTD's first sub child. Any node can be accessed with the information. There are mainly some drawbacks of this traverse as follows: a)too much memory consumption b) The speed is too slow. In view of the shortcomings above, this paper proposes an improved strategy with VTD records: 32bits is added in VTD record, which represents the VTD index of the first child. And if there is no child,

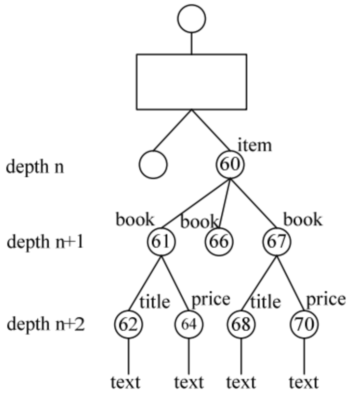
ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
Available online at www.ijarbest.com

International Journal of Advanced Research in Biology Ecology Science and Technology (IJARBEST)
Vol. I, Special Issue I, August 2015 in association with VEL TECH HIGH TECH DR. RANGARAJAN DR. SAKUNTHALA ENGINEERING COLLEGE, CHENNAI
National Conference on Recent Technologies for Sustainable Development 2015 [RECHZIG'15] - 28th August 2015

then marked it as -1 (0xffffffff). Then with it you can obtain the information of any node more quickly. As shown in following fig



A tree structure of XML documents is shown in Figure 6



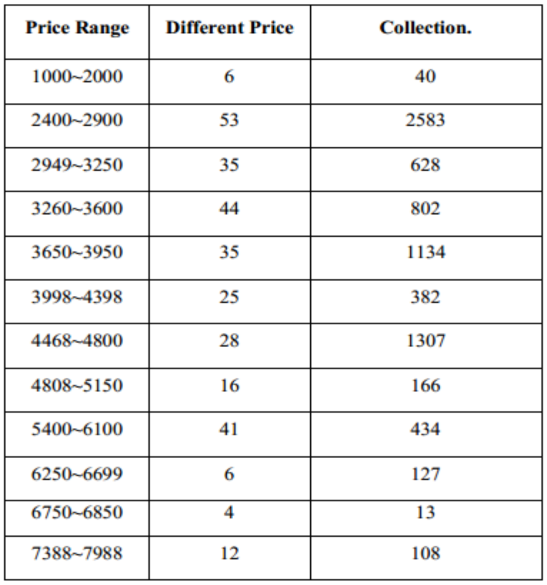
The algorithm for resolving the value of title and price of all the book nodes in accordance with the improved VTDrecord is as follows:

```
a) index[] = findVTDindex("book");
b) depth = findVTDdepth ("book");
// To find VTD index of book and the depth, and then store them to the array index[ ].
c) for(i: 0 —> index.length)
d) { if( index[i].FC != -1) // If index[i] has a child
e) { ilow = find(depth+1, index[i] );
// To find the index of index[i]'s first child in the depth of depth+1
f) ihigh = find(depth+1, index[i+1]);
// To find the index of index[i+1]'s first child in the depthof depth+1
// It is just the value of token of the index ilow —>ihigh-1
g) for(j: ilow —> ihigh-1)
h) if(token(j) == title) print(token(j + 1));
// Output title
i) else if(token(j) == price) print(token(j + 1));
// Output price
j) } //end of if
k) } //end of for
```

Due to the large scale of the XML document, VTD is first used to extract the text corresponding to these labels.Then SQL that is embedded in Java can be used to insert the contents to the database.

C. Data mining module

The main task of data mining module is to mine in the database in order to extract useful knowledge The recourse of this experiment is the 802 data sets extracted from pre-process module, that is to say, clustering of data for 802 samples. The sample has a total of five attributes. Attribute 1 is the sample ID, attribute 2 is trade names, and attribute 3 is treasurer. Since treasurer and trade have so many attribute values and the distribution is so broad, we remove them as redundant attributes. This experiment clusters them by k-means. After the k-means algorithm, the final clustering results are shown in Table 1.



Price Range	Different Price	Collection.
1000~2000	6	40
2400~2900	53	2583
2949~3250	35	628
3260~3600	44	802
3650~3950	35	1134
3998~4398	25	382
4468~4800	28	1307
4808~5150	16	166
5400~6100	41	434
6250~6699	6	127
6750~6850	4	13
7388~7988	12	108

CONCLUSION:

The above paper describes us about Web Mining by XML and VTD. First of all the HTML pages are converted into XML documents. After obtaining XML documents we will then store them in VTD to get some knowledge. Development in XML provides us to achieve greater efficiency in Web Mining.

REFERENCES

[1] Fan Ming, Meng Xiaofeng. The concept and technology of data
mining [M]. Beijing: The publishing company of engineering
industry, 2005.

[2] Xu Jingsong, Yang Bo. The basic tutorial of XML[M]. Beijing: The
publishing company of People's Post and Telecommunications, 2007.

[3] Zhang Weiming. The integration technology of information system
[M]. Beijing: The publishing company of electronic industry, 2002,
132-163.

[4] VTD-XML:The Future of XML Processing, http://vtdxml.
sourceforge.net

[5] A Quick Overview on Virtual Token Descriptor, http://vtdxml.
sf.net/VTD.html

[6] J. W. W., Wan, G., Dobbie. Mining Association Rules from XML
Data using XQuery. Proceedings of the second workshop on
Australasian information security, Data Mining and web Intelligence,
and Software Internationalisation, 32:169–174, 2004.

[7] Li, Lan; Rong, Qiao-mei Research of Web Mining Technology Based
on XMLNetworks Security, Wireless Communications and Trusted

Computing, 2009. NSWCTC '09. International Conference.