A Novel Approach for Automatic Data Extraction from Heterogeneous Web Pages

Teena Merin Thomas
Department of Computer Science
Sri Venkateswara College
Chennai, India

V. Vidhya
Department of Computer Science
Sri Venkateswara College
Chennai, India

ABSTRACT

World Wide Web is a vast and rapidly growing source of information. Web Pages contain a combination of unique data and template material, which is present across multiple pages to achieve high productivity of publishing. The template detection becomes a more attractive technique in the web pages, since the unknown template degrade the performance of web applications due to the irrelevant terms in the templates. The web pages is clustered using Agglomerative Clustering Algorithm based on the similarity of templates in the web pages.The unknown number of web pages and the partitioning of web pages is dealt with the help of Rissanen's Minimum Description Length Principle. Wrappers are generated for clustered heterogeneous web pages and the data encoded in the web pages are automatically extracted. Hence, the proposed approach for automatic data extraction let the web page users to access the data in a quick and easiest manner with better effectiveness and scalability.

General Terms

Template Detection, Agglomerative Clustering, Minimum Description Length Principle, Wrapper Generation.

Keywords

Wrap_match, MDL Clustering, Essential Paths.

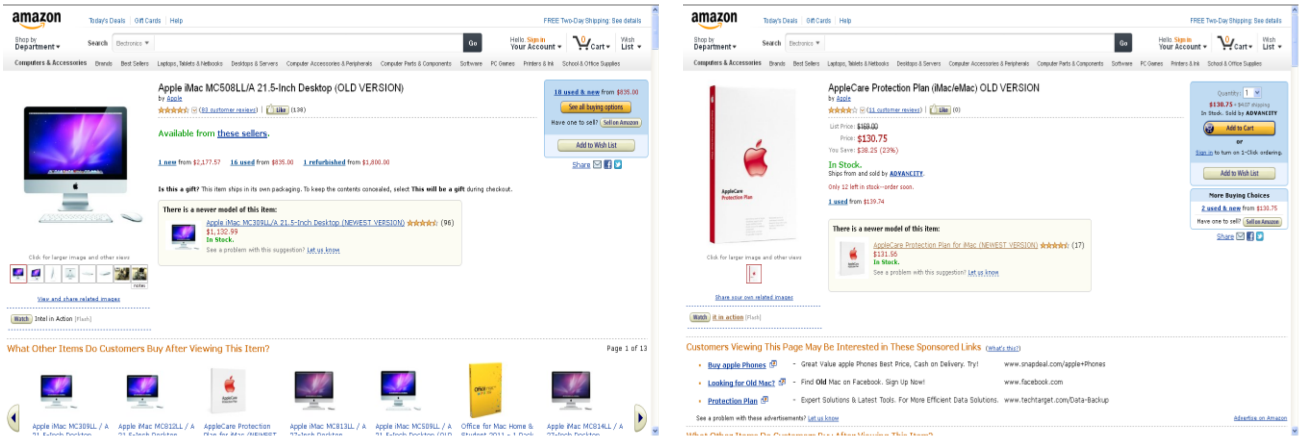
1. INTRODUCTION

World Wide Web is the richest and most dense source of information which is growing at a rapid rate, both in number of sites and in volume of relevant information. The websites contain a large number of pages which are generated from structured sources like databases. The data from the structured sources is encoded in semi structured HTML pages that employ templates for rendering. Thus web pages contain a combination of unique content and template material, which is present across multiple pages. The templates provide the users an easy way to access the contents guided by consistent structures. But the unknown templates contain a large number of irrelevant terms and thus template degrades the accuracy of information extraction from web pages. Thus template detection and extraction has become a more important technique to improve the performance of web applications like search engines, classification etc. For example, the value added services like comparison shopping, queries or manipulates the data from various Internet market places. But, there are mainly two challenges in automatically identifying the templates. First, the text which is a part of a template and text which is a part of data has to be correctly differentiated. Second, the schema of data in web pages is not a flat set of attributes, ie. the schema may contain non atomic attributes that are sets of values or optional attributes.

Software modules called Wrappers are used to extract the structured information from semi structured template generated web pages. Thus, wrappers extract the data from a particular web information source and deliver the data of interest in a self describing format. The process of building wrappers around web sources offer two advantages. The ability to obtain relevant information from an individual source is enhanced and all the web pages for which the wrapper is build can be queried using a common language. The wrappers can be constructed manually or automatically. Manually writing wrappers is time consuming and error prone. Also, many collections of web pages contain optional attributes. The templates from which the web pages are generated change very frequently, thus it requires repeated human intervention in generating wrappers.

Despite all the challenges, information extraction has received a lot of attention recently. Arasu et al.[2], Crescenzi et al.[8], Reis et al.[5] and many other researchers contributed a lot to the field of web information extraction. The previous template detection techniques group training pages into several clusters or classes, based on cues like URLs. Hers, the assumption is that all the web pages in a single site are generated from single common template. But it is not possible to classify massively crawled web pages into homogeneous partitions based on URLs because the small changes in scripts or CGI parameters will result in significant difference. In figure 1, both the web pages share a common template. It is not possible to correctly group the web pages just by comparing their URLs. Thus, with the popularity of dynamic URLs, it is no longer effective to detect templates using URLs especially for large scale websites.

Any HTML document can be represented as a Document Object Model (DOM) tree and many similarity measures for trees have been evolved for clustering the web pages. But clustering is very expensive with tree related distance measures. Also, all the previous works needed a small size labeled training data as input. The templates should be differentiated from the data for efficient web information extraction. It is not possible to determine the templates just by the frequencies of words and it should be assumed that a word can be part of either template or data. Also, wrapper has to be generated for the clustered web pages automatically.


amazon
Today's Deals | Gift Cards | Help
FREE Two-Day Shipping: See details
Shop by Department
Search
Electronics
Go
Hello. Sign in Your Account
0 Cart
Wish List
Computers & Accessories
Brands
Best Sellers
Laptops, Tablets & Netbooks
Desktops & Servers
Computer Accessories & Peripherals
Computer Parts & Components
Software
PC Games
Printers & Ink
School & Office Supplies
Like
See all buying options
Sell on Amazon
Add to Wish List
Share
Click for larger image and other views
View and share related images
Watch
Add to Cart
Share your own related images
Advertise on Amazon

Apple iMac MC508LL/A 21.5-Inch Desktop (OLD VERSION)
by Apple
(83 customer reviews) | Like (138)

Available from these sellers.

1 new from $2,177.57 16 used from $835.00 1 refurbished from $1,800.00

Is this a gift? This item ships in its own packaging. To keep the contents concealed, select This will be a gift during checkout.

There is a newer model of this item:
Apple iMac MC309LL/A 21.5-Inch Desktop (NEWEST VERSION) (96)
$1,132.99
In Stock.
See a problem with this suggestion? Let us know

18 used & new from $835.00
Have one to sell?

Watch Intel in Action [Flash]

What Other Items Do Customers Buy After Viewing This Item?
Page 1 of 13

Apple iMac MC309LL / A 21.5-Inch Desktop | Apple iMac MC812LL / A 21.5-Inch Desktop | AppleCare Protection Plan for iMac (NEWEST | Apple iMac MC813LL / A 27-Inch Desktop | Apple iMac MC509LL / A 21.5-Inch Desktop (OLD | Office for Mac Home & Student 2011 - 1 Pack | Apple iMac MC814LL / A 27-Inch Desktop

AppleCare Protection Plan (iMac/eMac) OLD VERSION
by Apple
(11 customer reviews) | Like (0)

List Price: $169.00
Price: $130.75
You Save: $38.25 (23%)
In Stock.
Ships from and sold by ADVANCITY.
Only 12 left in stock--order soon.
1 used from $139.74

There is a newer model of this item:
AppleCare Protection Plan for iMac (NEWEST VERSION) (17)
$131.56
In Stock.
See a problem with this suggestion? Let us know

Quantity: 1
$130.75 + $4.07 shipping
In Stock. Sold by ADVANCITY
or
Sign in to turn on 1-Click ordering.

More Buying Choices
2 used & new from $130.75
Have one to sell?

Watch it in action [Flash]

Customers Viewing This Page May Be Interested in These Sponsored Links (What's this?)

- Buy apple Phones - Great Value apple Phones Best Price, Cash on Delivery. Try! www.snapdeal.com/apple+Phones
- Looking for Old Mac? - Find Old Mac on Facebook. Sign Up Now! www.facebook.com
- Protection Plan - Expert Solutions & Latest Tools. For More Efficient Data Solutions. www.techtarget.com/Data-Backup

See a problem with these advertisements? Let us know

What Other Items Do Customers Buy After Viewing This Item?

Fig 1: http://www.amzon.com/dp/B002QQ8IO6 and http://www.amazon.com/gp/product/B00007J8SC

In this paper, in order to overcome the limitations in extracting the data from the web pages, a novel algorithm for extracting the data by identifying the templates is proposed. The web pages and the templates are represented as a set of paths in its DOM Tree structure. The overhead in measuring the similarity among the web pages is reduced by considering only the paths. The Rissanen's Minimum Description Length (MDL) Principle is used in clustering to deal with the unknown number of web pages. After, clustering the web pages using the MDL Principle, the templates are the clustering model itself and it can be clearly differentiated from the data. Wrappers are generated for the clustered web pages using a matching algorithm. The wrappers are then used to extract the data encoded in the web pages. The proposed algorithm is fully automated and robust.

2. RELATED WORK

Several approaches have been proposed for the accurate extraction of data from web pages. An HTML web page can be represented as Document Object Model tree. Many similarity based measures are there for clustering of web pages based on similarity between trees. But tree related distance measures are very expensive for clustering.

Arasu et al.[2] proposed extraction of structured data from web pages which deals with extracting structured data from large collections of web pages with a common template. The work considered any word as part of template. But the elements of a template are detected only by the frequencies of words. Moreover, all web pages are assumed to be of same template.

Reis et al[5] developed a system which automatically extract web news and relevant text passages from the web pages of a given web site using Tree Edit Distance. But it is very expensive to apply operations on trees to a large number of web pages. Also, it is not easy to select proper training data of small size.

Crescenzi et al.[8] proposed a method for clustering the web pages based on their structure. Web page clustering without data extraction is focused in this work. Also, web pages are compared only by their link information. Zhao et al.[11] proposed a fully automatic wrapper generation for search engine results. The work concentrated only on the problem of extracting result records from search engines.

Vieira et al.[3] proposed a fast and robust method for web page template detection and removal. Since clustering is based on tree related distance measures, it is very expensive. Zheng et al.[10] proposed a joint optimization technique which combines wrapper generation and template detection. It needs labeled training data for clustering. The algorithm used only leverages the HTML Tag Tree structure and does not involve any content. Also, tree related distance measures used for clustering is very expensive. Nazli et al.[1] proposed automatic data extraction from template generated web pages. The technique does not take into account dynamic features like Java Script.

3. PROBLEM FORMULATION

As the templates degrade the accuracy of information extraction, the proposed work focuses on the identification of template and extraction of data to increase the performance of search engines. The proposed work extracts the effective data from web pages in an easiest manner. The scalability is also ensured by employing the Minimum Description Length Principle.

4. SYSTEM MODEL

The proposed model involves four phases. The primary phase of the proposed model is HTML parsing which is explained in section 4.1. In HTML parsing the web pages are parsed to generate the DOM tree. The path of each node is identified from the DOM tree. The second phase deals with the identification of essential paths from the DOM trees of web pages and is explained in section 4.2. After identifying the essential paths, the essential path matrix is calculated for the web pages. The third phase deals with the clustering based on MDL cost which is explained in section 4.3. In MDL cost based clustering, the Minimum Description Length Principle is employed to cluster the web pages with similar template structures. The fourth phase of the model is wrapper generation which is explained in section 4.4. The wrapper generation deals with generating a wrapper for the clustered web pages to extract the data encoded in the web pages. The wrapper is generated by inferring a grammar for the HTML code and it helps in extracting the data encoded in the web pages.

4.1 HTML Parsing

4.1.1 DOM Tree Generation

The Document Object Model (DOM) tree is the representation of the HTML Page. Each DOM node of a DOM Tree represents an HTML tag pair. Thus DOM represents an HTML web page as a tree structure. The entire web page is represented as a document node, every HTML element as a tag node and the texts as text node. Any node can be considered as a part of the template. For example, consider a simple HTML web page 'd' in figure 2. The DOM Tree for the web page 'd' is given in figure 3.

```
<html>
<body>
<h1>hello</h1>
<br>List
</body>
</html>
```

Fig 2: Simple Web Page 'd'

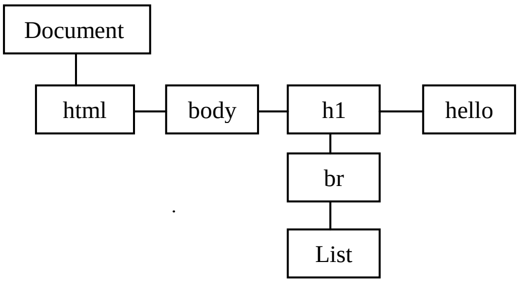


Fig 3: DOM Tree for the web page 'd'

The Path of a node in DOM tree is denoted by listing the nodes from the root to the particular node. The delimiter "|" is used as the delimiter between the nodes. For example, in the web page d, path of the node "hello" is "Document|html|body|h1|hello".

4.2 Essential Path Identification

For a given web page set, D={d_1,d_2…. d_n}, a path set P_d is defined as the set of all paths in D and is denoted by {p_1,p_2…. p_n}. The support of a path is defined as the number of web pages in D which contain the path and is denoted by s_i. A minimum support threshold is defined for each web page d_i The mode of the support values of all paths in each web page is defined as the minimum support threshold for each web page. If there are several modes, minimum mode is taken as the minimum support threshold. The Essential Path for a web page d_i is defined as the path which is contained in web page d_i and support of the path is at least the given minimum support threshold. The set of all essential paths in d_i is denoted by EP(d_i). Given web page set, D and path set P_d , an essential path matrix M_E is defined with 0/1 values to represent the web page with their essential paths. If path p_i is present in web page d_j, then the value of the cell (i,j) in matrix, M_E is 1, otherwise 0. The matrix M_E is calculated using the procedure given in figure 4.

4.3 MDL Cost based Clustering

The large number of web pages are clustered based on the similarity of templates used in the web pages. Thus the wrapper for each cluster is generated simultaneously.

4.3.1 Matrix Representation

For a web page set D, n clusters are defined such as C={c_1,c_2…. c_n}. A cluster c_i is denoted by a pair (T_i, D_i) where T_i is a set of paths representing the template of c_i and D_i is the set of web pages belonging to c_i. A web page is allowed to be in single cluster only. To represent clustering model C={c_1,c_2…. c_n} for the web page set, D, three matrices are defined. The matrix M_T represents each cluster with its template paths, M_D represents each cluster with its member web pages and M_Δ is a difference matrix with 0/1/-1 values so that the value of M_E is defined as follows

$$M_E = M_T . M_D + M_\Delta \ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(1)$$

```
Procedure essentialPath(D/* Document set*/)
begin
for each document $d_j$ in D do{
for each path $p_i$ in $d_j$ do{
if $s_i \geq$ minsupport threshold$_i$
        Mark $p_i$ as Essential Path of $d_j$
}
}
return Essential Path
end

Procedure essentialPathMatrix(D, $P_D$)
begin
Declare matrix $M_E$ of order $|P_D|$ x $|D|$
for each document $d_j$ of D do{
for each path $p_i$ in $P_D$ do{
if path ε essential path of $d_j$
      set $M_E$ [i][j]=1
}
}
return $M_E$
end
```

Fig 4: Essential Path and Essential Path Matrix Calculation

4.3.2 Minimum Description Length Principle

The unknown number of clusters is dealt by Rissanen's Minimum Description Length Principle. The Minimum Description Length Principle states that the best model for a given set of data is the one which minimizes the sum of the length of the model and the length of encoding of the data, when described with the help of the model. The above sum is defined as the MDL cost of the clustering model, C.

A clustering model, C is described by partitions of web pages with their similar template paths. The encoding of the data is given by the matrix, M_Δ. The MDL cost of the clustering model C is L(C) and the matrix is L(M). Let the probabilities of 1s and -1s in the matrix be Pr(1) and Pr(-1) respectively and Pr(0) be that of zeros. The entropy H(X) of a random variable x is given by equation 2. The variable x can take the values -1, 0 and 1.

$$H(X) = \sum -Pr(x) \log_2 Pr(x) \text{ ---------------------------(2)}$$

The MDL cost of L(M_T) and L(M_Δ) is calculated by the equation 3. The L(M_D) is given by |D|. $\log_2$|D|.

$$L(M)= |M|. H(X) \text{ ------------------------- (3)}$$

The MDL cost of the clustering model C is given by equation 4.

$$L(C)= L(M_T) + L(M_D) +L(M_\Delta)\text{------------------------(4)}$$

4.3.3 Agglomerative hierarchical clustering

The clustering algorithm used is given in the figure 5. The input parameter is D, a set of web pages. The output is a set of clusters, C={c_1,c_2…. c_n}, in which c_i is represented by the template paths T_i and the member web pages D_i. Initially each web page is considered as an individual cluster. Whenever a

```
Input: Document Set D
Output: Clustering of documents in D with similar template
structures
algorithm MDL Clustering(Document Set D)
Clustering Model C=(c1,c2,...cm) ci=(di,ep(di))
ci, cj- best pair to be merged and ck -merged pair
(ci, cj ck)= FindBestPair(C)
C'=C –{ci, cj} U (ck)
while C' ≠ C
(ci, cj ck)= FindBestPair(C')
end

procedure FindBestPair(C)
minMDLCost=1
for each pair (ci, cj) in C
begin
tmpMDL=calculateMDLCost(ci, cj)
if (tmpMDL<minMDLCost)
minMDLCost=tmpMDL
ck=(ci, cj)
end
return (ci, cj , ck)

procedure CalculateMDLCost(ci, cj)
Dk=Di U Dj
for each px in Ek
begin
if (sup(px , Dk) ≥ (j Dj+1)/2
add Px to EPk
Ck=(Dk, EP(dk)
C'=C-(ci, cj) U (ck)
MDL= $(\beta/\alpha)$( No: of 1s in $M_T$+ No: of 1s in $M_\Delta$ + No: of
-1s in $M_\Delta$)+L($M_D$)
end
return MDL
```

Fig 5:MDL Clustering Algorithm

pair of clusters is merged, the template matrix and document matrix for the new cluster is identified. ie. the clustering model C is denoted by matrices, M_T and M_D . Let α be the Pr(1) and β be the H(X) of matrix M_E. Then the MDL cost for the clustering model, C ie. L(C) is given by the equation 5.

$L(C)=\beta/\alpha($ # of 1s in M_T + # of 1s and -1s M_Δ)+ $L(M_D)$ --(5)

The pair with the maximum reduction in MDL cost is considered as the best pair to be merged. Suppose support(P_x,D_k) be the number of web pages in a cluster c_k with p_x as an essential path. Then the subset of EP_k which has all the paths whose support is greater than or equal to ($|D_k|$+1)/2 is considered as an optimal template path for the cluster. The algorithm for clustering the web pages is specified in [4]. Let the size of the essential path is 1. Then the complexity of the algorithm is $O(n^2l)$.

4.4 Wrapper Generation

The optimal paths in each cluster is the templates in the web pages and it will be identified. Thus the clustering is followed by a wrapper generation to achieve accurate extraction of data. The wrappers for each cluster are generated simultaneously. The wrapper is generated for a set of web pages in the cluster by inferring a grammar for the HTML code. An algorithm called wrap_match is used to extract the data from the clustered web pages. The wrap_match algorithm deals with two web pages at a time. A wrapper and a sample.

```
Procedure wrap_match(di, dj)
begin
for each $d_i$ and $d_j$є D $_i$ of cluster c $_i$
set $d_i$ as wrapper and $d_j$ as sample
parse each line of di with each line in dj
if there is Text mismatch
replace string in wrapper with #PCDATA
else if there is Tag mismatch
      {
if tag is optional {
        Identify optional is in sample or  wrapper
        Generalize wrapper with optional  using regular
          grammar
        }
else if tag is iterator{
        Identify whether iterator is in sample or wrapper
        Generalize wrapper with iterator using regular
         grammar
        }
      }
end
```

Fig 6: wrap_match Algorithm

Both the sample and the wrapper is converted into a list of tokens in which each token is either an HTML tag or a string value. The algorithm is explained in figure 6. The sample is compared with wrapper in the algorithm.

First, assume that the input web pages complies to the XHTML specification. Initially, one input web page is considered as the wrapper and is represented in the form of a regular expression. The sample web page is scanned using the wrapper. The grammar representing the wrapper is refined to find a common regular expression for the two web pages. This is done by identifying the dissimilarities between the wrapper and the sample. Two types of dissimilarities can occur while comparing. A Tag dissimilarity and Text dissimilarity. Whenever a dissimilarity occurs, solve the dissimilarity by generalizing the grammar which represents the wrapper. Once the scanning is completed, the wrapper generated is the common regular expression for the two pages.

4.5 Content Retrieval

The wrappers generated from each cluster is used to extract the data present in the web pages. The data extracted from the web pages are displayed in a structured format. Since the template detection and wrapper generation is done in an efficient way, the data extracted from the generated wrappers is accurate. Along with the data, the probability of using each template is estimated.

5. IMPLEMENTATION RESULTS

All the experiments mentioned in this paper is performed on an Intel Core2 duo 1.67GHz machine with 2GB RAM, running Windows Operating System. All the algorithms were implemented in JAVA with JRE version 1.6.0. The input HTML files are parsed using HTMLParser version 1.6 (http://htmlparser.sourceforge.net).

5.1 Real life Dataset

The dataset is the one which is used in EXALG[2]. The web pages which are from nine templates and the number of web pages from each template is from 10 to 50. The total number of web pages is 240. The web pages with similar template structures are correctly identified.

5.2 Performance

The web pages generated from similar templates are clustered and the data encoded in those web pages are extracted accurately. No web pages from different templates were clustered in same cluster. The wrapper generated, which is in the form of a grammar, accurately extracts the data from the web pages. The proposed model shows better accurate extraction of data over results of RTDM[5] and TEXT-MDL[6].

6. CONCLUSION

Many websites contain large set of pages which are generated using common templates. The widespread use of templates on the web, harmfully affects relevance judgment in many web IR and web mining methods such as classification and clustering. Templates negatively impact the performance and leads to wastage of resources. Hence to avoid these problems, a novel approach for identifying the template and extracting the data from the heterogeneous web pages have been proposed in this work.

7. ACKNOWLEDGMENTS

Our thanks to the staffs of Sri Venkateswara College, Chennai, India who have supported us towards development of the work.

8. REFERENCES

[1] Abdur Chowdury, Ling Ma and Nazli Goharian, 2008 Automatic Data Extraction from Template Generated Web Pages, Journal of Software, vol.19, pp.209-223.

[2] Arasu A. and Gracia Molina H. 2003, Extracting Structured Data from Web Pages, in Proceedings of the ACM SIGMOD International Conference on Management of Data, San Diego, USA, pp. 337-348.

[3] Cavalcanti J., da Silva A., de Moura E., Freire J., Pinto N. and Vieira K. 2006, A Fast and Robust Method for Web Page Template Detection and Removal, in Proceedings of the 15th ACM International Conference on Information and Knowledge Management, Virginia, USA, pp. 258-267.

[4] Gibson D., Punera K., amd Tomkins A. 2005, The Volume and Evolution of Web Page Templates, in Proceedings of the 14th International Conference on World Wide Web, Chiba, Japan, pp.830-838.

[5] Golgher P., Laender A., Reis D. and Silva A. 2004, Automatic Web News Extraction from Tree Edit Distance, in Proceedings of the 13th International Conference on World Wide Web, New York, USA, pp.502-511.

[6] Kim C. and Shim K. 2010, TEXT: Automaticc Template Extraction from Heterogeneous Web Pages, IEEE Transactions on Knowledge and Data Engineering, vol.23, no.4, pp 612-626.

[7] Liu B., Grossman R. and Zhai Y. 2003, Mining Data Records in Web Pages, in Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, USA, pp. 601-606.

[8] Merialdo P., Missier P. and Crescenzi V. 2005, Clustering Web Pages based on their structure, IEEE Transactions on Data and Knowledge Engineering, vol.54, no.3, pp. 279-299.

[9] J. Rissanen 1978, Modeling by Shortest Data Description, Automatica vol. 14, pp 465-471.

[10] Song R., Wen J., Wu D. and Zheng S. 2007, Joint Optimization of Wrapper Generation and Template Detection, in Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, California, USA, pp.894-902.

[11] Hongkun Zhao and Weiyi Meng 2005, Fully Automatic Wrapper Generation for Search Engines, in Proceedings of the 14th International Conference on World Wide Web, New York, USA, pp. 66-75.

[12] HTMLParser: http://htmlparser.sourceforge.net.