International Journal of Multimedia and Ubiquitous Engineering
Vol. 9, No. 12 (2014), pp. 405-420
http://dx.doi.org/10.14257/ijmue.2014.9.12.35

Application on Web Page Filtering Technology

Bo Shen*, Lei Li and Ning-wei Wang

Key Laboratory of Communication & Information Systems Beijing Municipal Commission of Education, Beijing Jiaotong University, Beijing100044, China
bshen@bjtu.edu.cn

Abstract

Web page filtering technology intends to filter out the large number of the repeated and theme-unrelated noise information and obtain useful information. Some web filtering methods cannot make full use of the layout and visual features. In view of the new mainstream "DIV+CSS" designing style of modern commercial web sites, this paper summarizes that elements laying in the same div blocks have common semantic features and proposed a DIV_FOREST model to represent the web pages. And in combination with the Vision-based Page Segmentation Algorithm, a DVPS Algorithm which considers both layout features and visual features was proposed to improve web page filtering efficiency.

Keywords: *Web Page Data Filtering, Web Page Segmentation, DIV_FOREST Model, DVPS Algorithm*

1. Introduction

As a pretreatment technology of Web information, page data filtering technology focuses on the purification of a large number of noise page information, including advertising, navigation, copyright information, *etc.*, [1] Although when browsing the web users make it easy to distinguish between them and the theme information, this noise information gives a great deal of interference to automatically information extraction. According to statistics, even the most efficient search engine --Google, the former, there are at least 28 records in the top 100 search pages which are nothing to do with the information user wants to obtain [2]. For search engines depending on web pages classification and indexing, page redundant information will seriously affect the accuracy of search results [3].

In recent years, because of the above reasons, the development of Internet technology makes network resources increase dramatically, and information extraction technology has begun to flourish. We can put the methods of data extraction from pages into two categories:

- By analyzing the set of the web from one or multiple sites with a common set of templates, extract the public part of each page, and remove them as noise.
- By using the DOM tree or other web page model, transform the page into easily logical splitting and mathematical processing model, and then by using the heuristic rules analytic web characteristics [6].

The following research is based on the former method. By sampling the pages of the site, a Style Tree can be built for the site, which is called the Site Style Tree (SST). Yi LAN, *et al.,* [4] introduced an information based measure to determine which parts of the SST represent noises and which parts represent the main contents of the site. Based on the policy of dynamic selection of threshold, Lin and Ho [5] determined the information blocks in web pages belonging to noise information blocks, or topic information blocks.

Another is based on the latter method. By setting the upper left corner of the screen to coordinate origin, Kovacevic [7] established a reference coordinate system, for positioning the relative positions of HTML objects in the screen.

In the actual page data extraction, the two methods have their own advantages and disadvantages because of different types of data sets. The first method is suitable for the case web with pages from one or a few websites. By using multiple pages the same site to extract their template, you can quickly distinguish topic information and noise information in all the pages of this website, high efficiency, but the applicability of the difference. The second method can make up the lack of in terms of flexibility brought by first method, this method can effectively deal with the situation that most web pages are not generated by the same template, but the diversity of the modern web, the complexity of the development has brought new challenges to the page data filtering scheme based on this idea.

Based on DIV tags dividing the content block of the page, this paper proposes a new data filtering scheme, DVPS algorithm. By determining if the block size factor Doc has reached the threshold, this algorithm decides how the page is divided into blocks, where each block is composed of several sub-tree of DIV, and corresponds the web visual block at the macro.

2. Analysis of Key Technologies

2.1. Web Information Extraction Based on HTML Structure Analysis

HTML is the foundation constituting the page. According to the positioning information of the Web page structure, this method parses the nested page structure into DOM tree. In this way, information extraction can be transformed into the operation on DOM tree. Based on this, the typical system includes RoadRunner [8], W4F [9], XWARP [10] and LIXTO [11], *etc.*,

XWARP can achieve human-computer interaction. The user can decide which area of the page can be as the starting position. The system is responsible for dividing the extraction area and extraction type. Based on the information provided by users, system treats the area divided by Table, list, *etc.*, as different semantic items, in order to generate different extraction rules, and then get the hierarchical relationship of the data, and finally XML file format return as the result. The deficiency of this system is that it is only applicable to the page with a clear area structure and is not smart enough.

W4F system includes an HTML parser for representing the page as a tree structure. Then by using a custom HEL language the system completes information extraction. Finally, the results will be saved in the custom data structure NSL.

In addition, as earlier automatic information extraction system, Implementation of the RoadRunner algorithm is getting a regular expression which can represent the general structure, by comparing the matching part of the homologous page DOM tree. It differs from the previous system, and does not require the user to perform tagging during the operation. Therefore, the system is also famous in the field of page information extraction.

Currently, HTML-based structural analysis approach is the most flexible page data filtering methods. It does not only fully consider the page structure and semantics characteristics, but also be able to design the different program to extract information according to need. So this method is a hot research field of information extraction.

2.2. Text Pretreatment

2.2.1. Regular Expression: regular expressions [12] can provide a mechanism which can search the specific string from the character set. It is an expression consisting of uppercase and lowercase letters, numbers and metacharacters, which can match a class of string. Users can build a string matching pattern by expression and then build comparative relationship with data files, web pages and other established target objects. In the Java language, the string regular expression should first be compiled into an instance of Pattern class and then create the Matcher object based on the Pattern class. We can create a regular expression matching any string. States involved in the matching process are stored in matcher, so allowing multiple matchers are allowed to share the same pattern.

2.2.2. Chinese Word Segmentation: Chinese word segmentation is the process dividing the sequence of characters into separate words or characters. The most common methods are: methods based on statistics, dictionary and understanding:

In a text, the more times adjacent characters simultaneously appear, the more they are likely to constitute a word. Therefore, the frequencies statistics of the adjacent word co-occurrence can well reflect the likelihood that they can constitute a word. Chinese word segmentation based on statistics uses this idea, and do not rely on word dictionaries.

Dictionary-based approach is also called string matching method. This method makes Chinese Characters string match the dictionary studied by the machine. Matching principles include Maximum Matching, Minimum Matching and Best Matching

Methods based on understanding establish mechanism for computer simulating human understanding to identify the words. This method usually includes three parts, semantic system, segmentation system and the total controller. The method uses semantic and syntactic analysis to eliminate ambiguity, including artificial neural network word segmentation and expert system word segmentation.

3. Page Data Filtering Scheme

In data extraction for complex pages, both the two methods, based on the tag layout features [13, 14, 16] and visual characteristics [15], have advantages and disadvantages. The analysis based on the tag layout features has clear ideas and rules, but lack grasp of the macro. The other method based on visual features uses complex algorithms to mimic the procession that human eyes divide complex page semantic. This method can seize the key features of the page analysis but its knowledge representation is very vague. On the other hand, this method lacks micro control.

Therefore, this paper can find a pages data filtering analysis to combine the two methods, which can take advantage of both the layout law and the visual characteristics of web pages and find structural analysis and data filtering solutions for complex pages.

This chapter presents new rules for the HTML page source preprocessing, transforming it into a nested tree model each of whose leaf node is DIV subtree. This model named DIV_FOREST models. DIV_FOREST convert the relationship between DIV tag in the HTML document to the relationship between DIV tree and subtree.

3.1. DIV_FOREST Model Introduction

With the development and application of CSS technology, more and more large news websites and popular social media websites are using *DIV+CSS* layout. The advantage of this design approach is that: the designer can pay the related logical and semantic characteristics content into the same DIV block, in order to control the page style by using CSS.

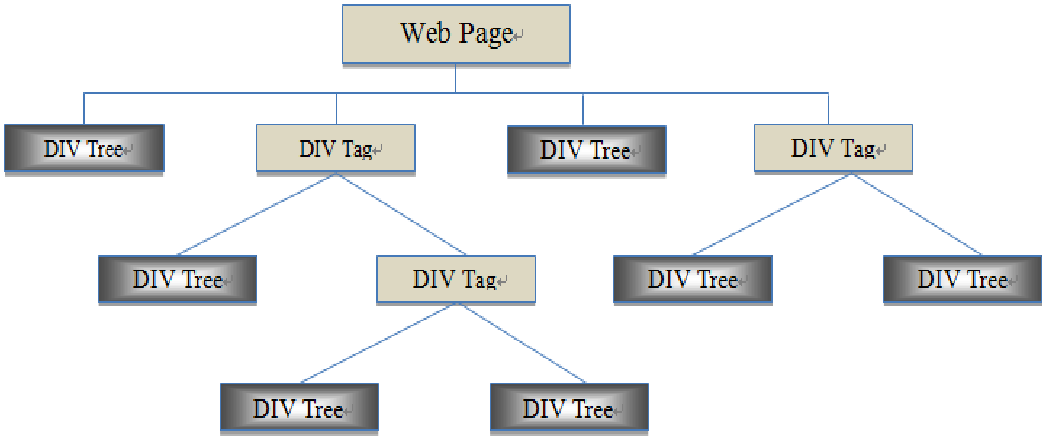


Figure 1. Structure of DIV_FOREST

Based on the analysis of Sina, Tencent, Netease and other major mainstream news site, we can see the page layout using *DIV+CSS* has replaced that using Table. So, the content extraction based on new layout page has become imperative for major professional search engine to update technology.

This paper proposes new rules for the HTML page source code pretreatment, transforming HTML page into a nested tree model each of whose leaf node is DIV subtree and named DIV_FOREST models.

To elaborate algorithm ideas more clearly, we define the concept of the algorithm as follows:

Definition1.DIV_FOREST model: DIV_FOREST model is a nested tree model used to represent page structure. Because the leaf node of the tree is also tree structure, we call this model DIV forest. The difference between DOM Tree model and DIV_FOREST is that: each node in the DOM tree corresponds to an HTML tag, but each non-leaf node in the DIV_FOREST corresponds to a DIV tag, and leaf node corresponds to DIV tag tree. Figure 1 is the structure of DIV_FOREST.

Definition2. Basic DIV block: Basic DIV block refers to the structure model consisted by each DIV subtree, shown in Figure 2. DIV subtree corresponds to the DIV_FOREST leaf node, which is the innermost DIV tags pair in all nested DIV tags of the HTML source code, whose root is a div tag. Other nodes are got by choosing non-DIV tags based on the model construction rules described in next section.

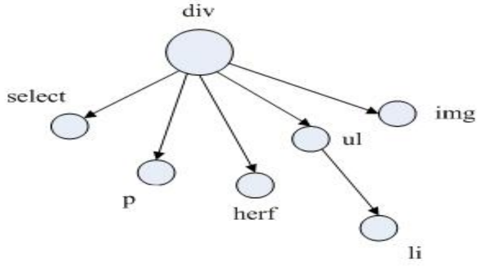


Figure 2. DIV Subtree

3.2. Model Construction Rules

Actually DIV_FOREST is an analytical model of the HTML source code. The difference with the DOM tree model is that DIV_FOREST is not a simple tree structure list of html tags, but different treatment for different kinds of tags in-depth analysis of the use of all kinds of tags. DIV_FOREST construction rules are as follows:

Depth-firstly traverse the DOM tree, mark and extract all div tags and retain a nested relationship of div tags. Finally, generate the DIV_FOREST tree structure.

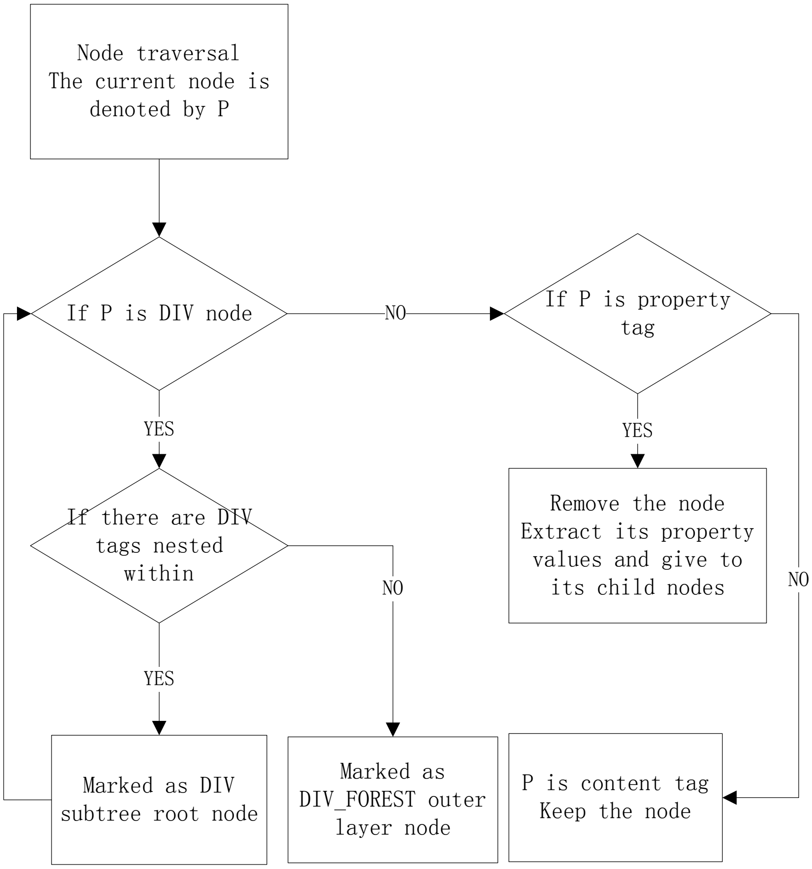

Node traversal
The current node is denoted by P
If P is DIV node
NO
If P is property tag
YES
YES
If there are DIV tags nested within
NO
Remove the node
Extract its property values and give to its child nodes
NO
YES
Marked as DIV subtree root node
Marked as DIV_FOREST outer layer node
P is content tag
Keep the node

Figure 3. The Core Algorithm Flowchart

There are many types of tags in the DIV subtree, which need determine if these tags should be deleted. First, classify non-div tags nesting within a div tag into property tags and content tags. These two types of tags relations in the DOM tree structure is: content tags are generally sub-node of property tags. When constructing DIV_FOREST model, only the content tags will retain as DIV tree leaf node, while the property tags will not exist as a node in DIV_FOREST model. This part is very crucial for the DIV similarity comparison and analysis on the importance of DIV blocks.

Figure 3 is the algorithm flowchart for DIV_FOREST model building process:

3.3. DVPS Visual Block Algorithm

After DIV_FOREST tree structure generated, by using VIPS algorithm based on visual characteristics block division, we proposes DVPS algorithm based on visual characteristics. By using a top-down way, this algorithm divide DIV_FOREST model into a combination a combination of basic DIV data unit, corresponding to the visual block on the page. As shown in Figure 4.

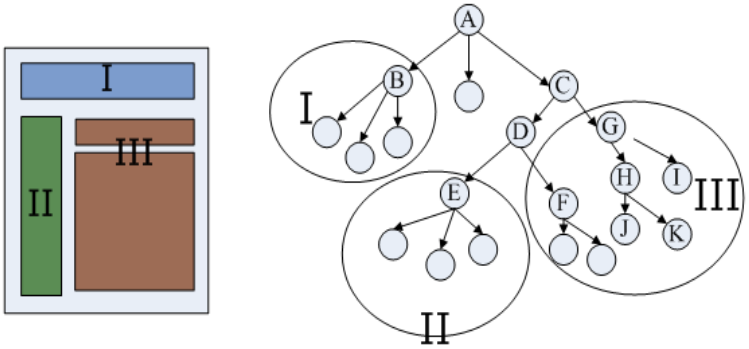


Figure 4. Web Page Segmentation and DIV Tree Model

A semantic block division process consists of three steps: page block extraction, separator bar detection and semantic block reconstruction.

Page block extraction: By using iterative loop, get the DOM tree structure and visual information. We start traversing from the root nodes and detect whether each of them can constitute a separate page blocks, giving a visual attributes according to their internal Doc (Degree of Coherence) value to the node which can, then save the page to the page block pool. Doc value was used to reflect the degree of close link inside the semantic block content, and satisfies the following two principles: the semantic block having closer link inside will get bigger Doc value; Doc value of sub-block is bigger than that of block.

Comparing Doc value and pre-set threshold PDoc (Permitted Degree of Coherence), we stop dividing when Doc reaches the threshold. In other words, the threshold value determines the degree of the partition.

After completing the first round of the page visual block detection, we have been able to extract visual subtree as independent DIV block. Then we use the method of use of separator bar detection to combine the DIV subtrees which not be used as an independent visual block on semantic level.

Separator bar detection: Separator bar is virtual boundaries on the page without crossing any block horizontally and vertically. We can know that the weight of separator bar is greater if the spacing between the two sides of separator bars is larger. We will increase the weight of separator bar if the property difference between two sides of separator bars is larger. Separator bar weights can be used as a measure of differences of the semantic blocks on two sides of the separator bar.

Semantic block reconstruction: After completing the separator bar detection and determine their weight, we start the semantic block reconstruction work. We combine semantic blocks on two sides of the separator bar which has the lowest weights into a new semantic block. Iterative loop runs until the divider has the largest weight. Then give new Doc value to the new semantic block. Determine whether the value of the new semantic block Doc meets the threshold criterion. If not, we start the next round of the semantic block reconstruction until all nodes meet the iteration stopping condition: Doc> PDoc. So far, we have completed the integration of visual division and semantic division.

Table 1. DIV Block Segmentation Rules

Rule 1	If DIV subtree contains other conventional visual block segmentation labels, such as <HR>,marked as D;
Rule 2	If all nodes in the current data block subtree are virtual text node (including text nodes). Marked as I, Doc value to 9;
Rule 3	If the background color of all DIV subtrees are the same in the current data block, marked as I, Doc value to 6-8 based on the number of sub-tree contains important Doc tags. Otherwise, marked as D;
Rule 4	The data block is marked as I if whose sub nodes' maximum size less than the threshold value, Doc value to 7;
Rule 5	If Data block contains a nested <table> tag or a <table> tag whose sub node' visual attributes such as font, background color and are obviously different, marked D;
Rule 6	If Chinese word count contained in the data block DIV below a preset threshold value T, labeled I, Doc value to 6.

We determine the principles Doc values here. DVPS algorithm is based DIV_FOREST tree structure, so we give the heuristic rule applied to DIV data block division according to the pages extraction of the visual and semantic characteristics. Because DIV subtree contains many different types of nodes, we need weight the data block according to the conclusions based on different heuristic rules and determine whether to continue dividing base on the weight. When a data block DIV met the rule judgment condition, it should be marked tags D (Dividable), otherwise marked I (In dividable), the rules listed in Table 1. We consider whether a node or a data block need to be further divided based on the Doc value.

4. DIV Data Block Feature Extraction

After the realization of the web data block division, and by extracting properties of each DIV subtree and analyzing CSS information, this section analyze the web features and quantify the importance of the web. Lastly, we give the evaluation index of the importance of the Web page data block.

4.1. Text Feature Extraction

After extracting the text data of DIV block by using regular expressions, we will have some operations such as Chinese word segmentation, passed part of speech filtering and stopped part of speech filtering, to achieve a text feature space dimension reduction, which will make the div block be expressed as n-dimensional feature vector : Seg(div)= (w_1,w_2,w_3.….w_n),where w_i represents the weight of t_i in the DIV data block, usually w_i is defined as frequency of t_i in a data block: tf_i(div), that is $w_i(d)=\varphi(tf_i(d))$.Generally we use a TF-IDF function to calculate the feature term weight:

$$\varphi = tf_i(d)\times\log\left(\frac{N}{n_i}\right) \tag{1}$$

Where n_i represents the number of the data block containing the characteristic feature words, and N represents the total number of data blocks.

After the quantization of data block text feature, we can calculate the similarity of the text blocks. Generally cosine similarity formula is used to evaluate vector similarity:

$$sim\left(D_i, D_j\right) = \frac{\sum_{k=1}^{n} d_{ki} d_{kj}}{\sqrt{\sum_{k=1}^{n} d_{ki}^2 \sum_{k=1}^{n} d_{kj}^2}} \qquad (2)$$

In addition, this section summarizes the high-frequency words that often appear in web function blocks, input feature terms into the database, and get the function block features thesaurus. Features thesaurus makes a collection of high-frequency such as *search*, *disclaimers*, *size* and *message*, *print*, *close*, *solemnly declare* and *copyright notice* and so on, which can used to determine whether DIV data block is the function block. If a DIV block contains a high-frequency Chinese text, and the number of high-frequency words in the features thesaurus reached the threshold, we can remove it as noisy block.

4.2. Spatial Feature Extraction

Common spatial feature of the page block is shown in Table 2. We use the relative length, the relative width, aspect ratio, and the relative distance from the center of the page, four variables, to quantify the spatial importance of the data blocks. Extraction process of spatial characteristics is as follows:

4.2.1. DIV Block Boundary Detection: DIV block boundary value is determined by the coordinates of the nodes it contains. Coordinate value of the node extracted from the CCS. The boundary values are determined by determining the boundary nodes of the DIV data block.

Table2 Space Feature Descriptions

Class	Feature	Characterization
Spatial Feature	BWIDTH	Data block width
	BHEIGHT	Data block length
	Center_Xc	X coordinate of center of data block
	Center_Yc	Y coordinate of center of data block

4.2.2. DIV Block Center Coordinate Calculation: Coordinate of center of data block, the data block length and width can effectively reflect the location and the percentage of the data blocks in the page. The center coordinates calculation formula as follows:

$$Center\_Xc = DIVLeft + BWidth/2 \tag{3}$$

$$Center\_Yc \ = \ DIVTop + BHeight/2 \tag{4}$$

To ensure that the location coordinates of the center can be objectively reflected, here the block center coordinates simplified to O_i = (X, Y) and normalized to （$Center\_Xc\ /PageWidth$， $Center\_Yc\ /PageHeight$）.

$DIVLeft$ is the left border of the DIV block, $BWidth$ is the width of the block, $DIVTop$ is the top border, $BHeight$ is height of the block, $PageHeight$ is height of the page， $PageWidth$ the width of page.

4.2.3. The Importance of the Evaluation Formula: The relative length and width relative are larger, the more likely it is the core of the page data block. The aspect ratio of the page is used to describe the shape of the DIV block. The aspect ratio parameter is used to evaluate the importance of the DIV data block based on the fact: most of the advertising page, copyright notice, and navigation pages are narrow DIV block and distributed around. Therefore, spatial importance formula can be obtained by Eq. (5):

$Imp_{(s)}(DIV_j)$ can be used as standard to measure importance of the spatial location of data block. Based on a large number of experimental data analysis, when w1 ~ w3 were taken to 0.5, 0.2, 0.3 we can get the most accurate results

$$Imp_{(s)}\left(DIV_j\right) = \frac{w_1\left(1-\frac{\sqrt{\left(X-0.5\right)^2+\left(Y-0.5\right)^2}}{\sqrt{2}/2}\right)+w_2\left(1-\frac{\min\left\{BWidth, BHeight\right\}}{\max\left\{BWidth, BHeight\right\}}\right)+w_3\left(\frac{BWidth}{PageWidth}+\frac{BHeight}{PageHeight}\right)}{w_1+w_2+w_3} \tag{5}$$

4.3. Semantic Feature Extraction

For different types of pages division based on the DIV block semantic properties analysis, semantic related attributes as shown in Table 3:

The relative amount of text data and the relative amount of body identification tags can be used as standard to determine theme block in a single topic page. The appearance of a combination of body identification tags indicate that the semantics block is more likely to body block. Such tags include <p>, <h1>-<h6>, <title>, <hr>, <p> so on.

Table 3. Semantic Feature Table

Property	Property Meaning
HtmlNumber	The number of HTML tags contained in a data block
TextQuantity	The number of text contained in a data block
ImgQuantity	The number of pictures contained in a data block
LinkQuantity	The number of links contained in a data block
LinkTextLength	The number of text containing links contained in data block

Using the method of regular expression matching to extract text and body identification tags in the DIV block, counts the number of tags, and calculate the proportion of body and key tags. The relative amount of text is calculated as follows:

$$RT = \frac{block\_textlen}{whole\_textlen} \tag{6}$$

Where $block\_textlen$ is the number of characters in the DIV block, $whole\_textlen$ is the number characters in the whole page. The relative amount of is calculated as follows:

$$RTopicTag = \frac{block\_TopicTags}{total\_TopicTags} \tag{7}$$

$block\_TopicTags$ is the number of body identification tags in the DIV data block, $total\_TopicTags$ is the number of body identification tags in the whole page.

Similarly, we can use the following formula determines the possibility that the DIV block is the function block:

$$RFunctionalTag = \frac{block\_FunctionalTags}{Total\_FunctionalTags} \tag{8}$$

Here is the formula to identify theme block in the single theme page:

$$Imp_{(y)}\left(DIV_j\right) = \frac{w_4 \frac{DIVTextLen}{PageTextLen} + w_5(1 - \frac{LinkTextLen}{DIVTextLen}) + w_6(1 - \frac{HtmlNumber}{TotalHtmlNumber})}{w_4 + w_5 + w_6} \tag{9}$$

5. Experimental Results and Analysis

5.1. Pages of Data Filtration Experiments

We use small-scale experiments to test DVPS algorithm. The purpose of the experiment is that: First, based on a single page information extraction, we prove the effects of DVPS algorithm. Second, we determine the threshold PDoc and the weights of w1 ~ w6 in evaluation formula. Web extraction experiments is under WindowsXP system environment, and we crawl pages of 157 sites including Sina, Sohu, Netease, Tencent, Dangdang, and Tianya to do small-scale testing

Table 4. Experimental Results of Identifying Web Topic Block

Internet site	Number of Pages	DVPS		VIPS	
		Precision Number	Precision Rate	Precision Number	Precision Rate
http://www.sina.com	20	18	0.9	18	0.9
http://www.sohu.com	22	21	0.95	20	0.91
http://www.163.com	19	18	0.95	17	0.89
http://www.qq.com	21	20	0.95	18	0.86

http://www.dangdang.com	23	22	0.96	21	0.91
http://bbs.tianya.cn	18	17	0.94	17	0.94
http://www.ifeng.com	17	15	0.88	16	0.82
http://www.china-pub.com	17	16	0.94	15	0.88

Information retrieval usually uses *Recall* and *Precision* as the evaluation criterion of information extraction efficiency. Recall refers to the ratio that the number of related documents retrieved in the number of that class of documents in the document library. Precision is also called retrieval accuracy which refers to the ratio of the number of related documents retrieved in the number of all documents retrieved.

This paper use VIPS algorithm and DVPS algorithm to extract the contents page of these sites. The precision rates of the two algorithms are as shown in Table 4.

By repeating test we can found that when threshold PDoc was set to 5.5, w1 ~ w6 were set 0.5, 0.2, 0.3, 0.7, 0.2, 0.1, we can get the most satisfying pages block and themes recognition effect.

From the experimental results we can draw a conclusion that the accuracy rate of DVPS algorithms is higher than that of VIPS algorithm. The algorithm combines the semantic description and visual information of block will bring great improvement on the page block partition and data filtering effect.

5.2. Application of Pages Data Filtering in Web Classification

5.2.1. Web Page Classification Algorithm: Web page classification is based on the automatic classification of text. By identifying and extracting thematic content of the page, we will be able to extract text content representing the page features. When we enter an unknown type of document into classifier, the document can be mapped to a given category of vector space field.

The construction method of Web page classifier including Artificial Neural Networks, Machine Learning and Web classification based on statistics model [17]. With its cross-cutting property of probability theory, approximation theory, statistics, and other areas, Machine Learning algorithm achieves many breakthroughs improving efficiency and reducing complexity, becoming the mainstream of the page text automatic classification methods. K-Nearest Neighbor (KNN) is one of the algorithms based on Machine Learning.

KNN algorithm is commonly used in web page classification algorithm. Its core idea is: if the most of the K most similar sample in the feature space of a sample are belong to a category or, we determine the sample can also fall into this category. The decision-making process can be quantified by the following formula:

$$y\left(x,c_i\right)=\sum_{d_i\in KNN} sim\left(x,d_i\right)y\left(d_i,c_j\right)-b_j \tag{10}$$

Where, x is the space vector waiting to be classified pages, d_i is the vector for the classification in training set, c_j refers to the web categories, $y\left(d_i,c_j\right)=\begin{cases}0 & d_i\in c_j\\ 1 & d_i\notin c_j\end{cases}$, sim (x, d_i) is the similarity of x and d_i.

5.2.2. Evaluation Criteria: In web classification process, while Precision increases, Recall tends to decline. Therefore, it is not intuitive to evaluate the algorithm by respectively using Recall and Precision. When evaluating the effect of pages extraction, we want to make a composite measure of this two performance indicators. F-Measure can be as a comprehensive standard. Definition of F-Measure is shown as follows:

$$F = \frac{\left(\beta^2+1\right)PR}{\beta^2 P + R} \tag{11}$$

Generally, we take β = 1. When analyzing extraction effect of experiments with a large number of data, the macro value F1can be used as the criterion, the value of Macro-F1 can be obtained by the equation:

$$Macro - F_1 = \frac{2\times\sum_{i=1}^{m} P_i \times \sum_{i=1}^{m} R_i}{\left(\sum_{i=1}^{m} P_i + \sum_{i=1}^{m} R_i\right)\times m} \tag{12}$$

P_i is the Precision of the ith document category; R_i is the Recall of the ith document category. We can see from the formula that only when both the value of p and r are large, the macro value F1will be relatively large.

5.2.3. Experimental Data Sets: By using web crawlers, we extract eight categories news web pages on *Sina* Website, including military, social, arts, health, finance, technology, travel, sports. 3280 articles are extracted, 2320 as the training set, 960 as the test set.

After data acquisition, we input the pages untreated and purified by VIPS and DVPS algorithm into KNN classifier. We have to determine what value of K can be to achieve the best classification results based on the experimental data. The variation of K values as shown Figure 5:

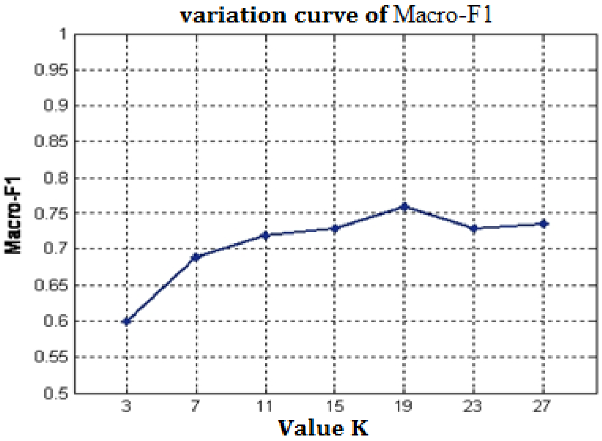


Figure 5. Extraction Result

Based on the above results, in subsequent experiments, we take K = 19.

The number of training set, test set and the Precision, Recall of the Web page classification experiment are listed in Table 5.

Table 5. Experimental Results of Identifying Web Topic Block

Category	Training Set	Test Set	Correct Classification	Classified to This Category	Recall	Precision
Military	310	120	105	120	0.875	0.875
Social	290	120	114	123	0.95	0.927
Arts	323	120	111	117	0.925	0.949
Health,	240	120	117	153	0.975	0.965
Finance	260	120	108	108	0.9	1
Technology	320	120	102	108	0.85	0.944
Travel	287	120	117	136	0.975	0.929
Sports	290	120	96	114	0.8	0.842

5.2.4. Classification Result Analysis: The effect of the web classification using plain text classification techniques is not ideal. Efficiency and accuracy of Web page classification is dependent on the representation model and classification algorithms, which can reflect the pages structural features completely as far as possible. On the basis of DVPS algorithm, we divide the web page into blocks and extract their feature. Finally, we achieve pages classification by using the extraction of theme block text feature. Comparative experimental results shown in Figure 6:

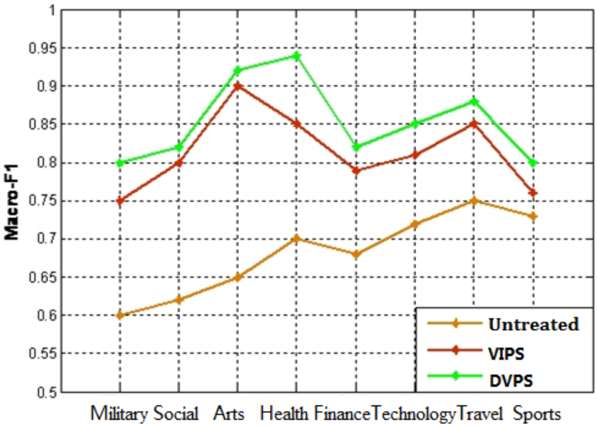


Figure 6. Extraction Result

6. Conclusions

Based on the analysis of the most popular DIV + CSS page layout, we proposed a DIV_FOREST web representation model for this new popular web design standards. By learning the thought of DOM tree page representation. The page is represented as a tree structure with DIV tags as the partition boundaries. Compared to the DOM tree method, DIV_FOREST model retains the structure and semantic correlation of the internal nodes in the div tag. Based on the division of the page, DVPS algorithms respectively extracted and quantified block DIV semantic feature, spatial characteristics, and visual features, and treated them as the standards determining the data block type and the degree of importance of the data block. DVPS algorithm combines the semantic description and visual information of block will bring great improvement on the page block partition and data filtering effect.

Finally, the web de-noising process is used as the improvements in Chinese Web Page Classifier. Then we input the pages untreated and treated by DVPS algorithm into Chinese Web Page Classifier to test. The test results prove that DVPS algorithm can improve the Chinese web classification accuracy.

Acknowledgements

This work was partially supported by the National Natural Science Foundation of China under Grant 61172072, 61271308, the Beijing Natural Science Foundation under Grant 4112045, the Beijing Science and Technology Program under Grant Z121100000312024, the Academic Discipline and Postgraduate Education Project of Beijing Municipal Commission of Education.

References

[1]. S. Xiaohui, L. Jian and W. Jinlin, CSS Based Segmentation of Web Pages, vol. 9, no. 29, **(2009).**

[2]. F. Tao, “The method of noise elimination in web information based on DOM tree and display attribute”, JOURNAL OF SHANGQIU TEACHERS COLLEGE, vol. 9, no. 26, **(2010).**

[3]. G. Yan, G. Shiwen and T. Liqiu, JOURNAL OF SOUTHWEST JIAOTONG UNIVERSITY, **(2007).**

[4]. L. Yi, B. Liu and X. Li, “Eliminating noisy information in Web pages for data mining”, KDD, **(2003).**

[5]. L. Shian-hua and H. Jan-ming, “Proceeding of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining”, ACM, **(2002).**

[6]. C. Deng, Y. Shipeng, W. Ji-Rong and M. Wei-Ying, VIPS:a Vision-based Page Segmentation Algorithm, vol. 79, **(2003).**

[7]. M. Kovacevic, M. Diligenti, M. Gori and V. Milutinovic, “Recognition of Common Areas in a Web Page Using Visual Information: a possible application in a page classification”, the proceedings of 2002 IEEE International Conference on Data Mining (ICDM’02), **(2002)** December 61-62, Maebashi City, Japan.

[8]. V. Crescenzi and G. Mecca, Road Runner: Towards Automatic Data Extraction from Large Web Sites”, In Proceedings of the 27th International Conference on Very Large Database, **(2001),** pp. 318-326, Roma, Italy.

[9]. A. Sahugue and F. Azavan, “Building Intelligent Web Applications Using Lightweight Wrappers”, Data Knowledge Engineering, vol. 3, no. 36, **(2001).**

[10]. L. Baoli, C. Yuzhong and Y. Shiwen, “Research on Information Extraction: A Survey”, Computer Engineering and Applications, vol. 10, **(2003).**

[11]. R. B. Gartmer, S. Flesca and G. Gottlobb, “Visual Web Information Extraction with Lixto”, Proceedings of 27th International Conference on Very Large Database, **(2001),** pp. 119-126, Roma, Italy.

[12]. M. Kovacevic, M. Diligenti, M. Gori and V. Milutinovic, “Recognition of Common Areas in a Web Page Using Visual Information: a possible application in a page classification”, In: the proceedings of 2002 IEEE International Conference on Data Mining (ICDM’02), **(2002)** December 61-62, Maebashi City, Japan.

[13]. Y. Man-quan, C. Tie-rui and X. Hong-bo, “Research and design of HTML parser based on page segmentation”, Computer applications, vol. 4, no. 25, **(2005).**

[14]. L. Cun-he and X. Chao, “DSS_DOM: A New Page Segmentation Model”, Journal of Information & Computational Science, **(2009).**

[15]. Y. Yang and X. Liu, “A Reexamination of Text Categorization Methods”, Proceedings of ACM SIGIR Conference on Research and Development in Information Retrival, **(1995).**

[16]. C. Hong-yao, Z. Zheng-yu, C. Ye, Z. Peng and Z. Li-fang, “Content extraction technique for web pages based on HTML-tags”, Computer Engineering and Design, vol. 31, no. 24, **(2010).**

[17]. Z. Lina, “Research of Feature Selection of Chinese Web Page Categorization”, China University of Petroleum, **(2009).**

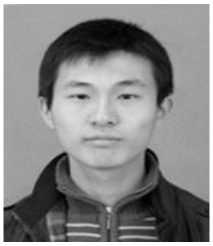
Authors

Bo Shen

He is an Associate Professor in School of Electronic and Information Engineering in Beijing Jiaotong University. He received his Ph.D. degree in The Communication and Information System, and his research interests include the Recommendation System and the Computer Communication.

Lei Li

She received B.E. degree of Communication Engineering in Beijing Jiaotong University, 2011. Now she is a graduate student in department of Electronic and Information Engineering. Her major is Communication and Information systems.



Ning-wei Wang

He is studying in Beijing Jiaotong University. Now he is in grade one. He is studying in department of Electronic and Information Engineering. His major is Communication and Information systems.