International Journal of Grid and Distributed Computing
Vol. 9, No. 10 (2016), pp.311-320
http://dx.doi.org/10.14257/ijgdc.2016.9.10.28

An Approach of Information Extraction Based on Dom Tree and Weight Value

Haitao Wang[1] and Shufen Liu[2]

[1]School of Computer Science and Technology
Henan Polytechnic University
GaoXin District, JiaoZuo City, HeNan Province, China
[2]School of Computer Science and Technology
JiLin University
QianJin Street, ChangChun City, JinLin Province, China
[1]Jz_wht@126.com

Abstract

Eliminating noisy information and extracting information content from web pages are increasing to become an important research issue in information retrieval field. In this paper, we present an approach of information extraction based on Dom tree and weight value calculation, which contains the following steps, parse the web page to construct the Dom tree, extract the title and keywords, calculate the weight value and obtain the content. The experimental result shows that this method has the higher accuracy ratio by the various themes content extraction.

***Keywords**: Information extraction, Dom tree, Weight value, JSoup, Web pages*

1. Introduction

The Internet has become a major information source distributed over web pages. Conversely, web pages contain noisy content, including advertisement, banners, menus, and unnecessary links, which can adversely affect performance of text-based processing systems such as search engine, web summarization, question answering and text understanding. In this instance, informative context, i.e. text content, headline, date or author name, can be used to enhance the results of these techniques. However, how to extract body content is difficult, as a web page contains both noisy and informative content in a same file [1]. This file consists of Hypertext Markup Language (HTML) tags and context between these tags that allow us to display pages in web browsers.

Web is an invaluable source of data for studies, extracting correctly the web body information is the course which exploits and obtains the main content of web page [2], which is the foundation of information process, such as, information retrieval, data mining, machine translation and so on. So, how to extract the useful information from the web pages or massive data set correctly and efficiently is becoming a hot research issue in the field of information retrieval. As the motivation discussed above, this paper proposes an approach of extracting body information content from web page based on Dom tree.

The rest of this paper is organized as follows: Section 2 presents a brief description about relevant research work, Section 3 details the course of information extraction and the key steps, experimental results are presented in Section 4, a conclusion is given in Section 5.

2. Related Work

All printed material, including text, illustrations, and charts, must be kept within the parameters of the 8 15/16-inch (53.75 picas) column length and 5 15/16-inch (36 picas) column width. Please do not write or print outside of the column parameters. At present, there

ISSN: 2005-4262 IJGDC

exist some research results of web page body information extraction, which includes the following aspects in general, that is, the wrapper method, the feature method based on web page source code and web page framework method.

The former is the traditional informational extraction method and obtains the valuable information from the special web page information source according to a certain of information pattern. Although completing the information extraction of web page tree, it only caters to the web page from the same web portals and can't meet the task requirement of various web pages because of the un-normal and complex web page architecture, the middle one is to search the body domain of web page from the HTML source code, there are few research on this aspect, ref. [3] proposed a method of web page body information extraction based on classification algorithm, which obtains the body contents by means of constructing classifier and association rules, ref. [4] present a method of method based on uncertain data. The last one is get the information by parsing web page, analyzing the framework of web page. Ref. [5] applied the web structure and proposed an information extraction method.

Another some schemes are also proposed for information extraction, *e.g.,* HUA *etc.,* [6] presented a wrapper method using a classifier and a search technique to score subsets of features according to their predictive, which is why they tend to obtain better performances than filters, even at the expense of high computational cost [7-8]. To the issue of dealing with the large dataset and high dimensional data [9], filters are the most suitable alternative, feature selection methods are adequate for both classification regression tasks, although most research is done related to classification. In the feature selection field, it's difficult to find ideal methods that can deal directly with multiple class problems, as very little work has been done in this aspect. The main difficulties to be taken into account in multiple class algorithm are following: on one hand, the dataset presents one or several classes that contain a considerable high number of samples than the data of the other classed, the other hand, determining which features are appropriate for each class is complicated, because feature selection results in a set of attributes that could represent only the majority classes.

Considering the above issues and preliminaries during research work, this paper presents an information extraction method, which contribution is mainly lies in following aspects: (1) presents the detailed approach and steps for obtaining the body information of web pages, (2) propose the concrete formula to calculating the weight value of sub tree and leaf node, (3) fully analyze the web tags feature and parse the web page frame in-depth, then construct the Dom tree of web page by means of JSoup tools. Experimental result and statistics data illustrates that this approach is feasible and own the relative high correct ratio.

3. Information Extraction Course

The approach developed in this study contains automatic rule creation instead of hand-craft rule insertion. These rules are used to infer informative content from simple HTML pages. Similar to other studies, our approach first extracts DOM-based features and utilizes these features to extract informative contents. A model is design for this task, which is based on two block tags: *DIV* and *TD,* selected as the most suitable markers for determining the boundaries of informative content. Because the system is constructed on *DIV* and *TD* tags, we can automatically determine the most comprehensive rule sets and maintain efficiency in the informative extraction. Figure 1 shows the workflow of this approach which included learning process, extraction process, rule selection and creation of a well-formed document based on the appropriateness criterion of the rule for the web pages. This workflow consists of two main steps.

1. Rule induction from a ML method

2. Efficient informative content extraction from rules

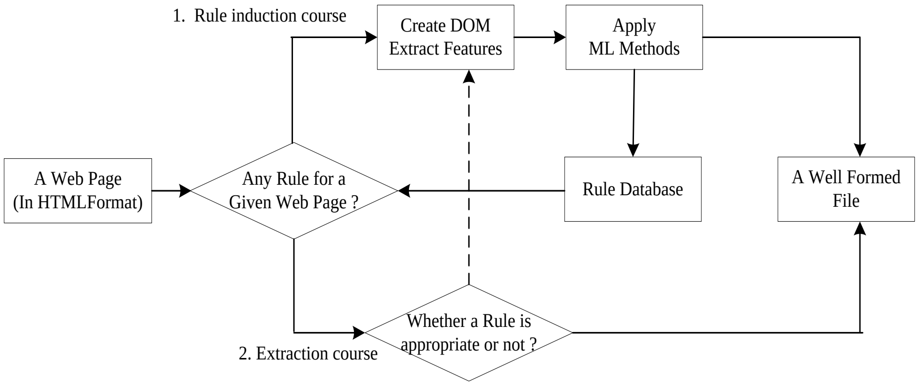

Figure 1. Workflow of Feature Extraction Method

In the first step, rule induction is performed by ML methods; in the second, the extract rules are used to determine informative content web pages. The rules inside a well-formed file that contain simple informative content are constructed. This procedure is as follows. The given web page is first tested against whether the rules are stored in the database. If the rules are in the database, then the test is performed whether there are appropriate rules for the web page or not. Otherwise, an ML method occurs to induct the rules and create the well-formed file. This decision is made when the rule produces a single result. In the rule induction phase, marked as step one, DOM is created and features are extracted from this DOM tree. An ML method then applied. For this step, we compare several different machine learning methods in the following sections and choose Decision Tree Learning with the sub tree Raising method as the most effective and accurate method for the dataset.

3.1. Rule Induction with ML

In the rule induction step, the learning stage produces rules for efficient web content extraction. Preparing the learning stage requires a data set and the appropriate features derived from this dataset. The dataset should be predefined, and it should not contain too much noise but should include a low number of samples. So, we utilized the HTML and DOM structure for feature selection. As we know, Html is a simple and effective markup language sued to develop web sites and contains several tag sets for visualizing content. A web browser simply interprets these tags and creates a web page that human can easily understand. Developers wanted to demonstrate their visual content with richer features, including JavaScript, used a hierarchy called DOM. An HTML tag is generally formed from the HEAD, which contains necessary information about the web site, and the BODY, which issued to visualize the content. DIV and TD tags, also called block tags, are used to separate the web site into several practical blocks, and they are generally referred to as block markers. An A HREF tag is used to give links to different web pages. H1 and P tags are used to format the text.

Html is simple and effective markup language used to develop web sites, and contains several tag sets for visualizing content. A web browser simply interprets these tags and creates a web page that a human can easily understand. According the function of tags in the file, Html tags consist of three kinds of classification.

(1) *Layout tags* In arrangement, web pages contain several content-information field which called content block, while the content blocks are planned by the special tags which named the container tags with nesting. There are some tags which is usually used, such as <table>, <tr>, <td>, <p>, <form> and so on. According to the container tags, a tree architecture can indicates a web page and detail the layout framework of web content.

(2) *Decorated tags* A set of tags which called the important information tags is created by Html standard and used to decorate the display style, *e.g.,* bold, italic and so on. There are some important information tags which is usually utilized, such as <b>,<i>, <strong>, <h1>,<h2> *etc.*, The contents of these tags are very important and are used to attract the attention of user, so, they indicate certain of crucial content information which the designer wants to deliver.

(3) *Hyper link tags* A hyper link is an obvious feature of Html web page which differed from the traditional text and expressed the complicated relationship between web pages.

3.2. Determining Body Information Block

By utilizing DOM (abbreviated by Document Object Model) parsing and providing a kind architecture mark, parser makes each element, attribute of web document turn into the node of tree, such as the Figure 2, shown as follows.

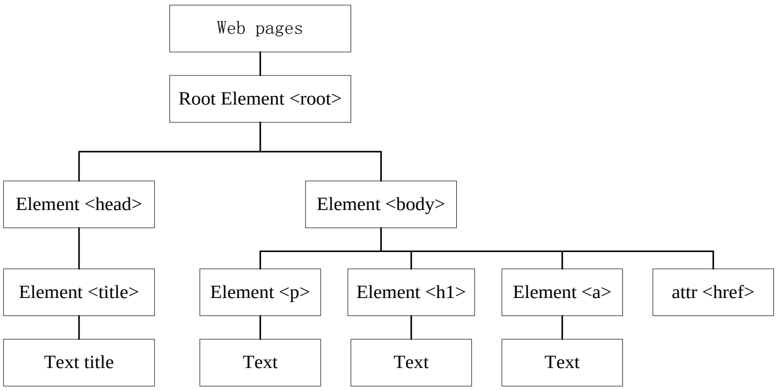

Figure 2. Dom Tree Architecture

From the Figure 2, we can see that among Dom tree only leaf nodes really include semantic information which is either link text or common text, the other nodes just play role of modifying the architecture layout, that is, the content of body text which will extract are all stored among the leaf node and the form information of body text are determined by the other none leaf node. To utilize the feature of architecture tree and body text information adequately, we set an information quantity for each node of structure tree, while the value of leaf node is determined by the feature of body text information contained and the value of other node by node type and information quantity of leaf node which parent node contained. Through analyzing a great number of theme web pages, it’s known that there is usually a certain trait for the distribution of body text information block in the theme web page [10-12]. By this trait, the location of body text information block is determined quickly in the architecture tree.

3.3. Information Extraction Steps

For a web page which will be processed, the body text extraction course based on weighting DOM tree includes the following steps:

(1) *Construct Dom tree for web page by parsing HTML source code of web page*. This step utilizes the Jsoup tool to parse the Html source code of web page, Jsoup is an Html parser of java source code, which can directly parse a URL or HTML text content, and present a convenient set of API to obtain or process the operation data.

(2) *Extract the title and key word of web page body* firstly, extract the content of title tag in web page, that is, the title information of body, then search all the meta-tag node, if a meta tag contains keyword item, then draw the content of it and construct the keyword sets of web page- $s_{meta\text{-}kw}$.

(3) *Preprocess the Dom tree* the visual info of web page all locates in the body tag, the body content is no exception. So, modify Dom tree by pruning and delete the sub tree of root node which Table 3.1, shown , *e.g.*, <script>, <form>, <object> tag node *etc.*

Table 1. Common HTML Tag Type

Tag type	Tag name
Block Tag	*body, div,table, p, blockquote, ul,ol*
Title Tag	*h1, h2,h3*
Row Tag	*span, img, strong, b, em, i*
Ignore Tag	*Script, style, form, noscript, object, embem*

Completing the preliminary pruning, combine the node of Dom tree according to table.1 shown and join the row tag node into the superior node. Besides there are usually multi-layer nesting situation for web pages, what happens is convenient to define the CSS (Cascading Style Sheets) and enables view layer to be friendly.

(4) *Calculate the weight value of leaf node*, Such as the Figure 1 shown in Dom tree, the content of web body text usually locates in a leaf node or several leaf nodes. This paper sets a triple <$WordNum_i$, $LinkNum_i$, $TagNum_i$> which indicates the attribute of leaf node i, where the $\mathrm{WordNum}_i$ indicates the number of none link word that the node contains content, $\mathrm{LinkNum}_i$ indicates the link number the node containing content, TagNum_i indicates the number that the node contains the pairs of HTML tags.

To quantize the possibility that these nodes contents is the body of web pages, this paper adopts the SNR (Signal to Noise Ratio) concept which is used in digital signal process, calculates the SNR of leaf nodes and makes it as the weight of leaf node to decide the important degree that nodes contain contents. The Formula.1 which used to calculate the weight for a leaf node i is listed as follows.

$$NodeImp_i = \frac{WordNum_i}{LinkNum_i + TagNum_i} \qquad (1)$$

Form the formula.1 above, it' obvious fact that the more number of none link words and the less links and HTML tags the leaf node contains, the more weight the leaf node is, which indicate the more possibility that the contents a node contains is the body text of web pages.

(5) *Calculate the weight value of sub tree*, set a two-tuples, <$WidthAttr_j$, Ss_j>, for the attribute of sub tree, *j*, among Dom tree, where $WidthAttr_j$ indicates the percentage of page tag container width/the whole page width corresponding to a root node of sub tree j, which value is obtained by the node related attribute or style information, Ss_j indicates the top leaf node set of sub tree j.

According the feature that it's a common fact that the web page body part occupies the large ratio to the whole web page width, we set the formula used to calculate the weight value of sub tree as follows.

$$Weight_j = \mathrm{WidthAttr}_j \times \frac{1}{n} \sum_{i \in Ss_j} NodeImp_i \qquad (2)$$

Where n indicates the number of leaf nodes among *Ssj* set.

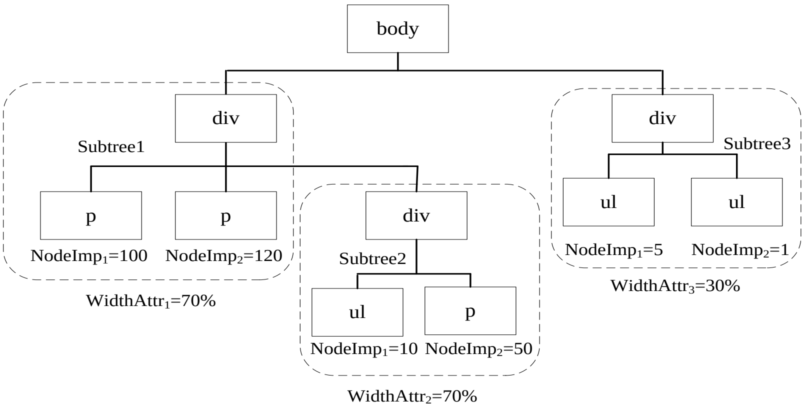

Figure 2. Dom Tree which Nodes Weighted

Assume a Dom tree, as the Figure 3 shown, which owns the NodeImp$_i$ value of each leaf node and $WidthAttr_j$ attributes of related sub tree, according the above formula, calculate the weights of sub tree1, sub tree2 and sub tree3 respectively, the value is 77 (which equals 70% × (100+120) / 2), 21 (which equals 70% × (10+50) / 2) and 31 (which equals 30% × (1+5) / 2) respectively.

(6) *Extract the body text of web page*, all weights of m sub trees among a Dom tree has calculated by above steps, among these weighted sub trees, if searching a sub tree which equals the Formula 3, it regards that the content which is contained in leaf node among Ss_j set is the body text of web page and the else context is the noise of web page.

$$Weight_j = \max(Weight_1, Weight_2, \cdots, Weight_m) \qquad (3)$$

As the Figure 3 shown, it's the biggest value of the sub tree 1 among all the weighted value of Dom tree, so we deem that the context which the leaf node in the Ss_1 set contained is the body text of web page.

To a special URL or HTML file, using the JSoup traverse the DOM node of web page and discard the tags which can be ignored. Conduct to parse the DOM tree from the bottom to top, firstly, find all the bottom layer tag nodes which are the leaf nodes constructing the extended DOM tree, calculate their weight value by the Formula.1, then, classify these leaf node which own the same parent node into a set, compose the set and the corresponding parent node and turn into a child tree, analyze the width ratio of tag correspond to the parent node in the web page, calculate the weight of child tree by the Formula.2, finally, find the child tree which has the maximum weigh value among all the child trees by the Formula.3 and extract their contents. So, the contents of tag which located in every leaf node of child tree correspond to a paragraph.

4. Experiment and Analysis

Author names and affiliations are to be centered beneath the title and printed in Times New Roman 12-point, non-boldface type. Multiple authors may be shown in a two or three-column format, with their affiliations below their respective names. The experimental hardware/software circumstance is: Inter(R) Core(TM)i5-2430M 2.4GHz, 2.0G RAM, OS Win7, developmental circumstance : Jkd1.7, MyEclipse. The experiment tools utilizes JSoup to parse web page, the reason is that JSoup is an opening source code java class library which is used to operate HTML document, traverse the node of DOM, search and extract the data of CSS and deal with the attribute and content which the HTML element contains.

To quantitative analyze the experimental result, define the formula to calculate the accuracy ratio which extracts from the body text as following, p = m/n, where m represents the number of web page which obtained correctly, n is on behalf of the number of all the web page which conducted to extract. Firstly, extract the news web page in the manner of batch, select 100 news web pages randomly from Sina, 163, QQ, Souhu web portal, which include three kinds of themes covering the society, business and technology. The related experimental result is illustrated in Table 1. Moreover, randomly select 200 blog web pages from Sina and 163 and show the effect of body extracting, the related experimental result is illustrated in Table 2.

Table 2. News Web Page Body Extraction Result

Origin	Number	Number of Correct extraction	Deviation number	Without content number	Accuracy ratio
Sina news	*100*	*97*	*2*	*1*	*97%*
Sina business	*100*	*98*	*2*	*0*	*98%*
Sina technology	*100*	*97*	*2*	*1*	*97%*
163 news	*100*	*98*	*1*	*1*	*98%*
163 technology	*100*	*96*	*3*	*1*	*96%*
QQ news	*100*	*99*	*1*	*0*	*99%*
QQ business	*100*	*98*	*2*	*0*	*98%*
QQ technology	*100*	*97*	*2*	*1*	*97%*
Souhu news	*100*	*96*	*3*	*1*	*96%*
Souhu business	*100*	*97*	*1*	*2*	*97%*

Table 3. Blogs Web Page Body Extraction Result

Origin	Number	Correct extraction number	Deviation number	Without content number	Accuracy ratio
Sina blogs	*200*	*196*	*1*	*3*	*98%*
163 blogs	*200*	*194*	*2*	*3*	*97%*

The average accuracy ratio of Table 2, is 97% and the one of Table 3, is 97.5%. The reason of deviation and error is that the corresponding web pages contain a part of image or video, which result in the wrong extraction that non-body contents takes for the body content, or directly deem the web page as non-content type web page. Because the information extraction author studied mainly aim at the text content type of web page, rather than the multimedia contents type of one which beyond this article research range. Therefore, this part of deviation which the extraction obtained will not influence the detecting result of similar web page, what's more, it is obvious fact this extraction course obtains the relatively higher accuracy ratio from the experimental data.

In Table 2, and Table 3, accuracy ratio is related with three factors, that is main, headline and article informative content. Main and headline content extraction achieve 83.7% and 75.5% of the whole accuracy respectively. This rate decreases in article information extraction because the extraction course doesn't care about tag naming in this class. It can be inferred that operators mostly consider this section unimportant, and it contains most operator errors. The accuracy of the main content extraction reaches 93.7% proportion of overall accuracy if the tag contains class and id attributes. This means that the method we presented is careful to prepare the main tag naming.

Investigating the Table 2, in depth, we can see that there is a slight drop of accuracy ratio about 163 web pages, this case caused by the fact that there are many ad banners, flash vide, and so on. This situation directly affects the extraction course and weight value of Dom tree. Moreover, to extract article information, including the date and author name, the method of several keyword-based also can be adopted though this approve is not error-tolerant for different languages, and preparing selective keywords is difficult.

5. Conclusions

To the issue of web page information extraction, this paper proposes an method based on Dom tree which owned the weight value, that is, determine the information block according element tags of web page, construct the Dom tree by the information block, then calculate the weight of leaf node and sub-tree, finally, extract the body content of web page by the means of the maximum weight value of sub-tree. This method not only improves the efficiency of web page processing, but also owns the universality and feasibility well, at the same, builds the foundation for text mining based on web, information retrieval, *etc.* So, this approach can apply on the following aspects, *e.g.,* data mining, social networks, gene expression and combinatorial chemistry and so on. Moreover, it brings the obvious advantages for data de-duplication, such as reducing the measurement cost and storage requirements, improving the classification performance. Finally, the experimental results demonstrate that our method achieves promising improvement on feature selection and classification accuracy.

Of course, there exist some aspects needed to improve, for example, identifying the information blocks accurately, getting rid of the un-useful advertisement information, especially the web page having the vague body information, *etc.*, how to settle these issues efficiently is the research direction for the authors in the future.

Acknowledgement

The authors are grate to the editor and anonymous reviewers for their valuable comments on this paper, and the work of this paper is supported by the National Nature Science Foundation of China (No.61300216), the National Innovative Approach Special Project (No.2012IM010200), the Science and Technology Research Projects of HeNan Province (No. 132102210123), the Open Laboratory Project of Mine Information Key discipline of Henan University.

References

[1] M. Baroni, F. Chantree, A. Kilgarri and S. Sharo, "Cleaneval: A competition for cleaning web pages", In Proceedings of the sixth international, language resources and evaluation (LREC'08), **(2008)**.

[2] D. Chakrabarti, R. Kumar and K. Punera, "Page-level template detection via isotonic smoothing", In WWW'07: Proceedings of the16th international conference on World Wide Web NewYork, NY, USA:ACM, **(2007)**, pp. 61–70.

[3] W. Jianwei and D. Yang, "A approach of web page information extraction based on classification algorithm", Computer Science, vol. 35, no. 3, **(2008)**, pp. 90-93.

[4] J. REN, D. L. SUN and X. CHEN, "Naïve bayes classification of uncertain data", IEEE the Ninth International Conference on Data Mining.

[5] K. Hofmann and W. Weerkamp, "Web corpus cleaning using content and structure", In Building and exploring web corpora UCL Presses Universities de Louvain, **(2007)**, pp. 145–154.

[6] C. Kohlschutter, P. Fankhauser and W. Nejdl, "Boilerplate detection using shallow text features", In Proceedings of the third ACM international conference on Web search and data mining (WSDM'10) New York, NY, USA:ACM, **(2010)**, pp. 441–450.

[7] B. Chiblovskii and L. Lecerf, "Scalable feature selection for multiclass problems", In Proceedings of the 2008 european conference on machine learning and knowledge discovery in databases (ECML PKDD'08), **(2008)**, pp. 227-240.

[8] I. Guyon, S. Gunn, M. Nikravesh and L. Zadeh, "Feature extraction, foundations and applications", Springer, **(2006)**.

[9] J. HUA, D. T. Waibav, "Performance of feature selection method in the classification of high-dimension data", Pattern Recognition, vol. 42, no. 3, **(2009)**, pp. 409-424.

[10] X. FAN, "Research and implement of feature selection in text categorization", Xi'an Northwest University, **(2011)**.

[11] M. Song and R. Zhang, "Approach of web page body information extraction", Journal of Dalian University of Technology, vol. 49, no. 4, **(2009)**, pp. 595-598.

[12] B. Xue and C. Fu, "A New Clustering Model Based on Word2vec Mining on Sina Weibo Users' Tags", International Journal of Grid Distribution Computing, vol. 7, no. 3, **(2014)**, pp. 213-221.

Authors

Haitao Wang, born in 1974.10, Ph.D. vice professor, major in computer system architecture, his research interests includes cloud computing, parallel computing, data mining, high performance computing.

Shfeng Liu, born in 1950, professor, Ph.D. supervisor, research on computer cooperative work, clouding computing, simulation modeling, preside over national 863 key project, supportive project, major special projects and so on.