A Webpage Segmentation Method Based on Node Information Entropy of DOM Tree

Shengnan Zhang¹, Jiawei Wu²* and Kun Yang²

1School of Software, Shenyang University of Technology, Shenyang, China
2School of Information Science and Engineering, Shenyang University of Technology, Shenyang, China

*Corresponding author email: 598583879@qq.com

Abstract. Aiming at the problems of DOM tree-based webpage segmentation methods in rationality of segmentation results, efficiency, and relying on auxiliary means, in this paper, we propose a new method based on node information entropy of the DOM tree. First, it classified webpage tags and only processed tags that affected the structure of pages. Second, comprehensively considering the content features and the structure features of DOM tree nodes, it calculated information entropy of nodes and maximum text density of subnodes, and based on them, determined whether a node was an independent page block. Finally, this method did node fusion to get the segmentation results. Segmentation tests of multiple web pages with different structures show that the proposed method can accurately and quickly segment the content of web pages.

Keywords: Webpage segmentation; DOM tree; Node information entropy.

1. Introduction

Today, Internet has become an important source for people to obtain information. However, when users use small-screen mobile terminals to browse web pages designed for PC, there will be some problems, such as unreasonable page layout, poor screen adaptability, and scrolling and zooming the screen multiple times to locate valid information. In order to improve the experience of users, it is necessary to consider the adaptation of web pages to mobile terminals. That is to extract the information that users are interested in from web pages and to transform it into a format suitable for small screen display on mobile terminals [1]. At present, an important method for solving the adaptation of web pages to mobile terminals is based on webpage segmentation, that is, page blocks with relatively independent themes in the webpage are divided and transformed accordingly. Among them, the page block refers to the semantic unit of the webpage, in which the information usually has a high correlation and occupies a continuous area visually. Information extraction based on blocks can not only improve the accuracy of extraction, but also facilitate the processing of webpage adaptation or search engine optimization [2].

The existing methods for segmenting web pages are mainly divided into several categories, which are based on visual feature, text information, tags and DOM trees, as well as hybrid strategies composed of various schemes. As a classical algorithm based on visual feature, VIPS has high accuracy, but due to the complexity of visual features, it needs to process the visual information of all nodes, which causes low performance[3]. Literature [4] presents a method based on text information, but it has a poor segmentation effect on web pages using JavaScript for layout. Literature [5], which is based on tags and visual feature, is only suitable for the webpage with < table > tags for layout. DOM tree-based method, as a widely concerned segmentation method is proposed based on characteristics that after DOM parsing, HTML document can form a tree structure that may accurately describe the hierarchical relationship between elements in the webpage and is convenient for computer processing. However, in addition to using the structure features of web pages, generally, DOM tree-based methods need other ways to assist segmentation and result in high calculation cost [6-7].

Through a large number of observations, we think that the webpage segmentation is not only affected by the structure features of web pages, but also closely related to the content features. In general, if a DOM tree
node contains both linked text and non-linked text, then the node is usually a collection of information blocks with different themes, which has large node information entropy. Based on this rule, this paper proposes a new DOM tree-based segmentation method of web pages. It comprehensively considers structure features and content features of web pages, use node information entropy to segment the nodes of the parsed DOM tree, and get the final segmentation result by node fusion.

2. Basic Idea of Webpage Segmentation Based on Node Information Entropy

In this paper, the algorithm is divided into three steps: webpage cleaning and parsing, tag classification, and webpage segmentation. Among them, webpage cleaning is mainly to remove script codes and comments in web pages that are irrelevant to the theme and not in the scope of algorithm calculation [8], as well as, to correct some syntax errors in the webpage codes. The purpose of tag classification is to filter out the tags that affect the structure of web pages, and improve the efficiency and accuracy of the algorithm by reducing the number of tags to be processed.

After a webpage is parsed into a DOM tree, the algorithm will segment the webpage mainly according to the content features and structure features of the DOM tree. In terms of content features, considering the low correlation between linked content blocks and non-linked content blocks in the webpage, if a node contains both linked and non-linked text, the node is considered uncertain. Therefore, the algorithm introduces node information entropy \( E \) to evaluate the uncertainty of the content of a node. In terms of structure features, because there are some special cases, such as a few links embedded in text, in order to avoid over segmentation caused by simply relying on node information entropy, the maximum text density \( M \) of subnodes is used to evaluate the degree of links embedded in text nodes.

The presented segmentation algorithm is a recursive algorithm. It first judges whether the current node in the DOM tree should be segmented according to the tag classification and processing rules. If the segmentation is no longer continued, the current node is treated as a separate page block and added to the page block set. Otherwise, the algorithm calculates the information entropy \( E \) of the current code, the maximum text density \( M \) of its subnodes, and get their average value \( T \). If \( T \) is less than or equal to the threshold value, it is also considered as a separate page block, Otherwise, it is further split. Finally, the segmentation algorithm outputs the set of the segmented page blocks.

In segmentation, the threshold value is determined by experiments. In addition, in order to avoid excessive segmentation, the algorithm may use the structure features of DOM tree again to fuse the segmented results, and merge some nodes into a page block.

3. Webpage Preprocessing

3.1. Webpage Cleaning

Webpage cleaning includes tag clipping and correcting syntax errors in HTML codes, and its purpose is to improve the efficiency and accuracy of subsequent processing.

The specific clipping contents are as follows:
(1) \(<\text{style}>\) tag and its style sheet.
(2) \(<\text{script}>\) tag and its script code.
(3) Tag with display attribute value “hidden” and its content.
(4) Tag with empty content.
(5) \(<\text{head}>\) tag and its contents.
(6) \(<!- -- >\) tag and its contents.

After clipping, web pages need to be parsed into DOM trees. In this paper, we use jsoup, which is an open source tool based on Java. In addition to parsing, jsoup can also correct the syntax errors in the page source code and standardize the tags that do not follow the W3C specification.

3.2. Rules for Classifying and Processing of Webpage Tags

Different tags in DOM tree have different effects on the structure and layout of web pages. For example, \(<\text{p}>\), \(<\text{em}>\), \(<\text{a}>\) and other tags only decorate text content, while other tags, such as \(<\text{div}>\), \(<\text{table}>\), have a decisive impact on page layout. In this paper, the former is called text tag, the latter is called block tag. We stipulate that a tag that does not belong to the text tag is a block tag, the algorithm only processes block tags. Text tags and some block tags are listed in Table 1.
Table 1. Text tag and some block tag.

| Text tag                  | Block tag                  |
|---------------------------|----------------------------|
| A, abbr, acronym, address, area, aside, b, base, basefont, bdi, bdo, big, br, caption, cite, col, colgroup, dd, del, dfn, dt, dir, em, embed, figure, font, h1-h6, head, hr, i, img, ins, kbd, keygen, legend, li, link, mark, menuitem, meta, meter, noframes, noscript, optgroup, option, output, p, param, pre, q, rp, rt, s, samp, script, small, source, span, strike, strong, style, sub, summary, sup, tfoot, th, thead, time, title, track, tt, u, var, video, wbr | applet, article, audio, body, button, canvas, center, code, datalist, div, dl, fieldset, figure, footer, form, frame, frameset, header, html, iframe, section, select, table, tbody, textarea, ul, etc |

In segmentation, different types of tags are distinguished and processed accordingly. Processing rules are defined as follows:
Rule 1: If the tag types of all subnodes under the current node are both text tags, it is no longer split.
Rule 2: If the current tag type is block tag and its subnodes also contain block tag type, the current node is split according to the following segmentation algorithm.

4. Webpage Segmentation

4.1. Feature Definition

Content features: In terms of content, a webpage is basically composed of linked text and non-linked text, therefore, in this paper, linked text density \( \rho_1 \), non-link text density \( \rho_{nl} \) within a node, and the corresponding node information entropy \( E \) are regarded as content features. \( \rho_1 \) and \( \rho_{nl} \) are defined as follows:

\[
\rho_1 = \frac{ltn}{tn}, \quad \rho_{nl} = \frac{ntn}{tn}.
\]  

Where, \( ltn \), \( ntn \), and \( tn \) respectively represent the number of linked text, non-linked text and all text under the current node.

Node information entropy \( E \) is defined as follows:

\[
E = -\rho_1 \log_2 \rho_1 - \rho_{nl} \log_2 \rho_{nl}.
\]  

Where, smaller \( E \) is, the less uncertainty this node has, the more likely it is to represent an independent page block. On the contrary, this node may contain both linked content blocks and non-linked content blocks.

Structure features: In terms of structure, although linked content blocks and non-linked content blocks are generally separated, there is also a special case, that is, the page content is mostly composed of non-linked text, but a small amount of linked text is embedded in it. Its general structure is shown in Figure 1.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{structure.png}
\caption{Structure of a few links embedded in non-linked text}
\end{figure}

In this case, the non-linked content blocks are scattered under each subnode with only a small number of text. If it is split, it will cause excessive segmentation of the original related content blocks. Therefore, in this paper the maximum text density of subnodes is regarded as structure feature, and it is defined as follows:

\[
M = \max(N_{c1}, N_{c2}, \ldots, N_{cn}) / N.
\]  

Where, \( \max(N_{c1}, N_{c2}, \ldots, N_{cn}) \) represents the maximum number of non-linked text in all subnodes of the current node, \( N \) is the number of non-linked text under the current node.

Based on formula (2) and formula (3), the average value \( T \) of \( E \) and \( M \) is calculated, and we use it to measure whether a node should be split.

4.2. Node Fusion

After segmenting DOM tree roughly, some segmentation results need to be fused. The reason is that in some web pages, the subject content and noise may be located under the same parent node, as shown in Figure 2.
When the content in the `<div>` tag is noise, the paragraph composed of three `<p>` tags, which belongs to the same page block, may be divided into separate page blocks. So, we define the fusion conditions as follows:
(1) Text tag node.
(2) Nodes with a common parent node.
(3) Nodes are adjacent.
When two or more nodes satisfy the above three conditions at the same time, the nodes can be fused. Figure 3 is a schematic diagram of the fusion process of Figure 2. After node fusion, the over segmented nodes can be repositioned under the same parent node, that is, they can be fused into a page block.

4.3. Threshold Setting
In segmentation, a threshold should be set as the basis to determine whether the node continues to be segmented. Because different threshold has a great influence on segmentation results, we obtain the best threshold through experiments. In the experiments, 20 kinds of web pages with different features were selected as the test pages. The range of threshold was [0.1, 0.9], and the threshold interval was 0.1. The accuracy of segmentation results was evaluated by manual scoring, the scoring range was [0,1]. The higher the score was, the more reasonable the segmentation was.
Experimental results show that the segmentation accuracy increased with the increase of threshold at first, reached the maximum when the threshold was 0.5, and then decreased with the increase of threshold. According to the experimental results, 0.5 was selected as the segmentation threshold of this algorithm.

5. Experiment and Analysis
In order to verify the effects of the above method, the algorithm was implemented by Java and compared with VIPS algorithm.
There were 100 test pages in the experiments, which were respectively from Baidu Encyclopedia, Sina Blog, Tencent News, Blog Park and other websites. These pages had obvious differences in content and structure, and they can well illustrate the applicability of the algorithm. The segmentation results are shown in Figure 4 and Figure 5.

![Figure 2. DOM tree before fusion](image1)

![Figure 3. DOM tree after fusion](image2)

![Figure 4. Segmentation results of our algorithm](image3)
Figure 5. Segmentation results of VIPS

It can be seen from Figure 4 that two algorithms can segment the blog homepage better. Figure 5 shows that the segmentation effect of our algorithm for the news page was in line with human visual perception, but VIPS algorithm existed over segmentation. The reason was that when there was a large interval between elements under a certain node in DOM or there was an explicit separation line, the content that should be under the same node would be over segmented.

In addition, we tested the efficiency of two algorithms, and the comparison result is shown in Figure 6. We can find that our algorithm was superior to VIPS in efficiency. The reason was that VIPS used a lot of time to deal with the visual information of all nodes. Although the process of HTML parsing into DOM tree also took a certain time, the efficiency was still better than that of VIPS.

Figure 6. Comparison of the efficiency between two algorithms

6. Conclusion
Considering that most of the current web pages use DIV + CSS layout, and a small number of web pages use <Table> tags for layout, the proposed algorithm first classifies the tags, only deals with the tags that affect the page structure, reduce the amount of calculation and ensure the integrity of text content. Then, according to the information entropy of nodes, it judges whether a node is segmented, and introduces the maximum text density of subnodes and node fusion to avoid over segmentation.

Although this method has some advantages, we also find that the segmentation effect of the page with less text is poor. Therefore, in the next study, we will optimize it to make it more general.

Acknowledgments
This research was financially supported by general project of scientific research in colleges and universities of Liaoning Province (LFGD2017014).

References
[1] Jangam A, Zhang D, Zhou L, User-centered, device-aware multimedia content adaptation for Mobile web. IEEE/ACIS International Conference on Computer & Information Science. IEEE, 2014.
[2] Xuebo Sun, Dawei Zhang, A separator bar-based webpage segmentation algorithm, J. Computer Applications and Software. 31(2014) 286-289 (in Chinese).
[3] Qin Wu, Lijuan Hu, Jiuzhen Liang, Web information extraction based on block importance model and 2D conditional random fields, J. Journal of Nanjing University (Natural Science). 50(2014)79-85 (in Chinese).
[4] Jinseng Li, Huixiao Le, Mingwen Tong, Novel method of web page segmentation based on title machine learning. J. Computer Science. 45(2018)583-587 (in Chinese).
[5] Xianfa Wang, Yan Guo, Yue Liu, et al, Research on web page information extraction based on visual feature, J. Chinese Information Processing. 33 (2019) 103-112 (in Chinese).

[6] Kezhen Jia, Web page segmentation method based on DOM and image processing. J. Modern Computer (Professional Edition). 08 (2018)52-54 (in Chinese).

[7] Jun Li, Jun Chen, Lingfang Wang, et al, Approach to webpage segmentation and information extraction for vertical Websites. J. Application Research of Computers. 30(2013)844-847+852 (in Chinese).

[8] Kun Yang, Research on the migration technology of web information system to wechat platform. Shenyang: Shenyang University of Technology, 2017.