I J M I
International Journal of Machine Intelligence
ISSN: 0975–2927 & E-ISSN: 0975–9166, Volume 3, Issue 4, 2011, pp-354-358
Available online at http://www.bioinfo.in/contents.php?id=31

COMPARATIVE STUDY ON BROWSING ON SMALL SCREEN DEVICES

KRISHNA MURTHY A.[1]*, SURESHA[2]
[1]Department of Studies in Computer Science, University of Mysore, Mysore, India
[2]Department of Studies in Computer Science, University of Mysore, Mysore, India
*Corresponding Author: Email- krishnarjun.research@gmail.com

Received: November 06, 2011; Accepted: December 09, 2011

Abstract- Delivering Web pages to Small Screen Devices such as Mobile Devices, Personal Digital Assistants (PDA) etc., has become possible with the latest wireless technology. However, these devices have very small screen sizes, memory capacities and low bandwidth. Today most of the Web pages are designed for Large Screen Devices, which makes browsing on Small Screen Devices extremely difficult. Therefore, a method to reconstruct Large Screen Devices optimized Web pages for Small Screen Devices is essential. Proposed methods which involves segment the Web page based on its structure, followed by noise removal based on block features and to utilize the hierarchy of the content element to regenerate a page suitable for Small Screen Devices. In this article we give a brief overview of existing approaches, their advantages and challenges. Finally we give an overview of comparison of results.
Key words – Mobile Browsing, Small Screen Browsing, Web page Segmentation, Noise Removal, Browsing on Wireless Devices.

INTRODUCTION

Web pages (contents) are currently designed for the Large Screen Devices (LSD) and rich memory resources. LSD users can use convenient input devices such as a mouse, keyboard to retrieve any Web page from any Website. Downloading time is rarely a problem as the Personal Computers (PC's) are usually connected to the internet through high capacity lines and the large screen allows much irrelevant (noise) information's such as advertisements to be placed on the screen without overly distracting the user. At present, experiencing the internet on Small Screen terminals such as Mobile, Personal Digital Assistants (PDA) etc., is becoming very popular. The current Web pages are intended for LSD's are not suitable for Small Screen Devices (SSD).

The straight forward solution for browsing on SSD is to re-design the Web pages using specific markup languages such as WML, XHTML. Some notable sites, which are already done this, include Yahoo, CNN, and Google among others. Nonetheless, the vast majority of sites on the Web do not have customized Web pages for SSD's because it's time consuming process and not economical [12]. Compared to LSD's SSD's are not ideal platforms for surfing the Web. Because in SSD's, wireless bandwidth is quite limited, it's very expensive and screen size varies for different devices such as mobiles, PDA's etc., and devices such as mobile phones have limited memory capabilities. Normally, the content of a single Web page will be larger than what a mobile phone can hold. Therefore, methods to reconstruct LSD's optimized Web pages for Small Screen terminals are essential.

Tasks Need to Do

We can't directly achieve the above addressed problem. We need to follow the following three methods:

a. First need to analyze the content structure of Web pages, which means need to segment the Web pages into piece of blocks by using semantic structure of web pages. It is also very useful for many Web applications such as information retrieval/extraction, data mining and automatic page adaptation etc., [1] [2] [3][22].
b. In second step, need to extract the features of each block and analyze it whether it has relevant information or not (this process is called as noise reduction/removal) [6][7][8][9][10][11][12].
c. In final step, retrieve the relevant information's after removing the noise and re-arrange the relevant information's to fit on SSD's[14][15][16][17][18].

Above mentioned methods reduces the bandwidth to get download, memory of the Web pages and size of the Web pages. Following figure (Fig.1) gives the clear picture of above mentioned tasks

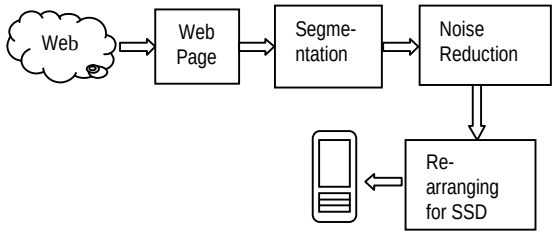

Fig.1- Block diagram of Browsing on SSD

LITERATURE REVIEW

In the recent years, browsing on SSD's have received substantial attention from both research communities and

market, but still remains very challenging on advanced technologies such as Flash, Silver light, XML and so on. From literature, it is clear that the existing systems are made based on HTML technology because of its more availability on Web. However, because of flexibility of HTML syntax, a lot of Web pages do not obey the W3C (World Wide Web Consortium) specifications, which might cause mistakes in Document Object Model (DOM) [4] tree structure. To provide better descriptions of the semantic structure of the Web page content, few new technologies are introduced. However as we can observe, still the majority of the Web pages on Web are written in HTML rather than the other technologies [1].

Web Page Segmentation

Today the Web has become the largest information source for people. Most information retrieval systems on the Web regard Web pages as the smallest and undividable units, but a Web page as a whole may not be appropriate to represent a single topic. A Web page usually contains various contents such as navigation, decoration, interaction, add(s) and contact information, which are not related to the topic of the Web page. Furthermore, a Web page often contains multiple topics that are not necessarily relevant to one another. Therefore, detecting the semantic content structure of a Web page could potentially improve the performance of Web information retrieval [1].

DOM tree construction for Web pages [4], tries to extract the structural information from HTML. However because of flexibility of HTML syntax, DOM might cause mistakes in tree structure. Moreover, DOM tree is initially introduced for presentation in the browser rather than description of the semantic structure of the Web page. For example, even though two nodes in the DOM tree have the same parent, it might not be the case that the two nodes are more semantically related to each other than to other nodes [1].

Vision Based Page Segmentation (VIPS) algorithm [1] is introduced to extract the semantic structure for a Web page. Such semantic structure is a hierarchical structure in which each node will correspond to a block. Each node will be assigned a value (Degree of Coherence) to indicate how coherent of the content in the block based on visual perception. The VIPS algorithm makes full use of page layout feature: it first extracts all the suitable blocks from the HTML DOM tree and then it tries to find the separators between these extracted blocks. Here separators denote the horizontal or vertical lines in a Web page that visually cross with no blocks. Finally, based on these separators the semantic structure for the Web page is constructed. VIPS algorithm employs a top-down approach which is very effective. VIPS works well even when the HTML structure is quite different from the actual layout structure. However, as it does not take into account the DOM tree information enough, if blocks are not visibly different, it may not work well and in many cases the weights of visual separators are inaccurately measured [2].

The DOM tree is a straight forward way to represent a Web page, but it's inconvenient for later processing: it does not describe layout information accurately and contains many useless nodes. Based on "Gestalt Theory" [2] a new method introduced to segment the Web pages. Gestalt Theory: A Psychological theory that can explain human's visual perceptive process. Four basic laws, Proximity, Similarity, Closure and Simplicity are drawn from Gestalt Theory and then implemented in a program to simulate how human understand the layout of Web pages.

First through pre-processing, a Web page is represented by a layout tree which concisely describes visual cues. Then, following the closure law, some commonly used design patterns are recovered. After that, the similarity and simplicity laws are recursively applied to the layout tree. At last the result is refined by the proximity law. The overall process is shown in Fig.2.

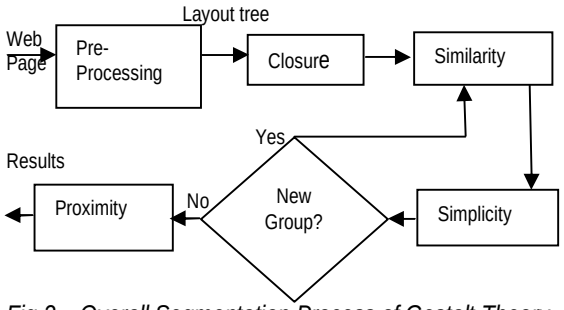

Fig.2 – Overall Segmentation Process of Gestalt Theory

Until this, the segmentation problem has mainly been addressed by analyzing the DOM structure of an HTML page, either by rendering and visual analysis or by interpreting or learning the meaning and importance of tag structures in some way, both using heuristic as well as formalized principled approaches.

However, the number of possible DOM layout patterns is virtually infinite, which inescapably leads to errors when moving from training data to Web scale. The system [3] with abstract block-level page segmentation model focuses on the low-level properties of text instead of DOM-structural information. Here the key observation is that the number of tokens in a text fragment (or more precisely, its token density) is a valuable feature for segmentation decisions. This reduces the page segmentation problem into 1-D partitioning problem and presents the Block Fusion algorithm for identifying segments using the text density metric. This approach is orthogonal to existing work and considers new and complementary aspects to solve the segmentation task.

Noise Removal

Unlike conventional data or text, Web pages typically contain a large amount of information that is not part of the main contents of the pages, e.g., banner, ads, navigation bars, copyright notices and so on. Such irrelevant information's (Web page noise) in Web pages can seriously harm Web Mining tasks (e.g. Web page clustering, Web page classification), search results as

well search speed, reduces the page citation, affects the Small Screen Browsing and so on.

By considering the above issues, new system [9] proposed to formulate the block importance estimation as a learning problem. Here VIPS [1] is applied for segmentation and Spatial features, content features of each blocks are extracted to construct a feature vector for the each block and then learning algorithms such as SVM and Neural Network methods are used to train a model to assign importance to each block. Following by this [6] system called Webpage Cleaner for eliminating noise blocks from Web pages is introduced, it first extract Web blocks using VIPS [1] then relevant Web page blocks are identified as those with high importance level by analyzing such physical features of the blocks as the block location, percentage of Web links on the block and level of similarity of block contents to other blocks.

The effective approach for boilerplate (noise) detection using shallow text features is proposed [8] (average word length, average sentence length, absolute number of words and link density etc.,) for classifying the individual text elements in a Web page and then compared the approach to complex, state-of-the-art techniques and shown that competitive accuracy achieved, at almost no cost. Recently [12] introduced the system to detect multiple noise patterns from Web pages. The method is based on the basic idea of Case Based Reasoning (CBR) to find noise pattern in current Web page by matching similar noise pattern kept in Case-Based. And applied back propagation Neural Network algorithm to classify the stored various noise patterns by matching similar noise data.

Browsing on SSD

Most of the Web pages in existence today are designed for desktop PC's, which makes viewing them on SSD's extremely difficult due to limited bandwidth, small screen & limited memory. Very first system [16] proposed by using a ranking algorithm similar to Google Page Ranking algorithm to rank the content objects within a Web page. This allows the extraction of only important parts of Web pages for delivery to mobile devices.

Followed by this system, [15] introduced with new page-adaptation technique which analyzes Web page structure and splits it into smaller, logically related units that can fit onto a mobile device screen. Here author first analyzed the HTML DOM tree and detected the high-level content blocks and then analyzed the content inside each high-level content block to identify explicit separators to determine where to split the blocks. Finally detect implicit separators to help split the blocks further. The overall analysis is to split Web pages into appropriate blocks so that users can browse page blocks on SSD's. [17] Introduced the method to reconstruct the PC's optimized Web pages for mobile browsing, here the approach is to segment the Web pages based on its content distance and utilize the hierarchy of the content element to regenerate a page suitable for mobile phone browsing.

In 2009, [18] proposed the novel approach to segment Web pages into mobile-fitted blocks guided by four general laws in E-Gestalt Theory. This method first group's visually and semantically coherent content into hierarchical parts according to similarity, closure and simplicity laws in E-Gestalt theory and then divides them into mobile fitted blocks using proximity law. Finally through proxy automatically re-author HTML documents into mobile-intended structures using segmentation results.

RESULTS AND DISCUSSION

In this section, the experimental results of Web page Segmentation, Noise Removal and re-arranging retrieval information's to fit on SSD's are discussed in detail. Table-1 [1] gives the performance comparison of query expansion using different page segmentation methods. The average retrieval precision can be improved after partitioning pages into blocks, no matter which segmentation algorithm is used. In the case of FULLDOC, the maximal average precision is 19.10% when the top 10 documents are used to expand the query. DOMPS obtains 19.67% when the top 50 blocks are used, a little more than FULLDOC. VIPS gets the best result 20.98% when the top 20 blocks are used and achieves 26.77% improvement [1]. It achieves best precision when no. of segments less than 30 Fig. (3). Fig. 4(a) shows the processing time of Gestalt Theory method [2]. About 87% of pages are processed in less than 2 seconds. The time is almost in proportion to the size of the layout tree, which is about one third of the size of the DOM tree. Here authors used recall to evaluate the performance of Gestalt method with VIPS [1] and PAV [5]. Here recall is the fraction of correctly recognized blocks over the standard blocks marked manually. N is the number of result blocks. As N grows, a page is broken into more and more smaller blocks. It is observed that Gestalt method always outperforms than PAV. But VIPS achieves the best results Fig. 4(b) when N is small because proximity plays the key role at that time [2].

Three learning methods are used [9] to learn the models such as SVM, non-linear SVM with RBF kernel and a RBF network. The best performances obtained by these methods are reported in Table-2. SVM with RBF kernel achieved the best performance with Micro-F1 80.2% and Micro-Acc 86.8%. The linear SVM performed worse than both SVM with RBF kernel and RBF network. The results indicate that a nonlinear combination of the features is better than a linear combination. In WPC [6] experiment, Naïve Baye's text classification is applied on three different data sets, cleaned using three approaches of Not Cleaned (NC), Template (TPL) [23] and Web Page Cleaner (WPC) to check the performance. Table-3 shows average classification (Naïve Baye's text classification) accuracy and standard error on four-fold cross validation are for method NC (79.68, 4.07), for method TPL [23] (96.23, 1.18) and for method WPC (98.12, 0.29).

The result [17] in Fig. 5(a) indicates that, for "Bottom" target content elements, the system is four times more efficient than the Google Wireless Trans-coder, and twice as efficient for "Middle" elements. These results prove that the proposed method significantly improves the usability of Web browsing on the mobile phone. Segmentation results of E-Gestalt are compared [18] with VIPS [1] and

Gestalt [2]. As illustrated in Fig. 5(b), E-Gestalt produces the most PERFECT blocks and the least ERROR blocks. The count of NOT-BAD blocks is much higher in other two methods, mainly because large blocks tend to be split into sub-blocks much smaller than screen size by VIPS and Gestalt. This reversely demonstrates the effectiveness of the fine-tuned dividing process for generating mobile-fitted blocks.

CONCLUSION

In this paper, we have presented a brief overview of challenges involved in designing a system to browse Web pages on SSD's. In addition to this, we have discussed different kind of existing techniques on Web page segmentation which is helpful to Web adaptation, information retrieval, information extraction and so on. Forward by this we have discussed some existing techniques of Noise Removal on Web pages which are helpful to improve the mining results (using Data Mining Techniques such as clustering & classification), Web page adaptation on SSD's, it also helps to improve search speed as well as search result and so on. We can conclude that existing Noise Reduction methods are feasible to clean noise data from any kind of HTML Web pages. Moreover we have discussed existing techniques to adapt Web pages for SSD's by using Web page segmentation and noise removal techniques as well as we have discussed some experimental results and comparative studies of existing systems.

Acknowledgments

Our thanks to the experts who have contributed towards development of the template

.

References

[1] Cai D., Yu S., Wen J. R., Ma W.Y. (2003) *MSR-TR*-2003-79.

[2] Xiang P.F., Yang X. and Shi Y.C. (2007) *Conference on Multimedia and Expo*, pp. 2253-2256 *IEEE*.

[3] Kohlsch utter C. and Nejdl W. (2008) *October 26-30, 2008 Napa Valley, California, USA.*

[4] *http://www.w3c.org/DOM/*

[5] Xiang P.F. (2006) *Web Intelligence*, 831 - 840.

[6] Jing Li and Ezeife C.I. (2006) *Springer*-Verlag Berlin Heidelberg, *LNCS 4080*, pp, 560-571.

[7] Haitao YAO, Zhiyi YINI (2009) *IEEE computer society* 1 - 5.

[8] Christian Kohlschutter, Peter Fankhauser, Wolfgang Nejdl (2010) *ACM WSDM* 441-450.

[9] Ruihua Song, Haifeng Liu, Ji-Rong Wen, Wei-Ying Ma, (2004) *ACM SIGKDD Explorations*, 6.

[10] Lan Yi, Bing Liu, and Xiaoli Li. (2003) *ACM SIGKDD 03*, 296-305.

[11] Lan Yi, Bing Liu, Lan Yi, Bing Liu (2003) *IJCAI-03*, Aug 9-15.

[12] Thanda Htwe, Khin Haymar Saw Hla (2010) *IEEE* , 281 - 285.

[13] Xin Yang, Yuanchun Shi (2008), *IEEE*.

[14] Shumeet Baluja (2006), *Proceedings of the 15th international conference on WWW*, *ACM*.

[15] Chen Y., Xie X., Ma W. Y. and Zhang H. J. (2005) *IEEE*, 9(1):50–56.

[16] Yin X. and Lee W. S. (2004) *In 13th WWW*, pages 338–344.

[17] Hattori G., Hoashi K., Matsumoto K. and Sugaya F. (2007), *In WWW*, 361-370.

[18] Xin Yang, Yuanchun Shi (2009) *IEEE/WIC/ACM*, 3, 46 - 49 .

[19] Dou Shen A., Qiang Yang A., Zheng Chen (2007), Published by *Elsevier* Ltd,43(6).

[20] Michal Marek, Pavel Pecina, Miroslav Spousta (2007) *Cahiers du Cental*, 5, 1-8.

[21] Bar-Yossef, Z. and Rajagopalan S. (2002) *In Proceedings of the 11th International World Wide Web Conference (WWW2002)*, 580-591.

[22] Xinyue Liul, Xianchao Zhan, Ye Tian and Hongfei Linl (2010) *NSFC*.

Table-1. Performance comparison using different page segmentation methods [1]

Number of Segments	Base line (%)	FULL DOC (%)	DOMPS (%)	VIPS (%)
3	16.55	17.56 (+6.10)	17.94 (+8.40)	18.01 (+8.82)
5		17.46 (+5.50)	18.15 (+9.67)	19.39 (+17.16)
10		**19.10 (+15.41)**	18.05 (+9.06)	19.92 (+20.36)
20		17.89 (+8.10)	19.24 (+16.25)	**20.98 (+26.77)**
30		17.40 (+5.14)	19.32 (+16.74)	19.68 (+18.91)
40		15.50 6.34)	19.57 (+18.25)	17.24 (+4.17)
50		13.82 (-16.50)	**19.67 (+18.85)**	16.63 (+0.48)
60		14.40 (-12.99)	18.58 (+12.27)	16.37 (-1.09)

Table-2. Comparison of learning methods [9]

Meth-ods	Level 1	Level 2	Level 3	Mi cro-F1	Mi cro-Acc
SVM (RBF)	0.787 (P) 0.813 (R)	0.807 (P) 0.808 R)	0.837 (P) 0.754 (R)	0.802	0.868
SVM linear	0.691 (P) 0.740 (R)	0.745 (P) 0.737 (R)	0.823 (P) 0.675 (R)	0.731	0.821
RBF net-work	0.727 (P) 0.717 (R)	0.752 (P) 0.755 (R)	0.777 (P) 0.799 (R)	0.746	0.830

Table-3. WPC average accuracy and standard error [5]

Case	Train	Test	Methods	Avg Accuracy (%)	Standard Error
1-1	25 (5 per class)	2475	NC	79.41	2013
			TPL	88.63	1.41
			WPC	**91.10**	0.69
1-2	50 (10 per class)	2450	NC	90.52	1.01
			TPL	92.42	0.96
			WPC	**95.44**	0.40
1-3	75 (15 per class)	2425	NC	95.44	0.42
			TPL	94.19	0.70
			WPC	**97.05**	0.20
1-4	100 (20 per class)	2400	NC	94.89	0.43
			TPL	94.45	0.33
			WPC	**97.11**	0.20
1-5	250 (50 per class)	2250	NC	97.40	0.37
			TPL	97.33	0.21
			WPC	**98.64**	0.12
1-6	500 (100 per class)	2000	NC	97.97	0.27
			TPL	98.09	0.09
			WPC	**99.00**	0.06

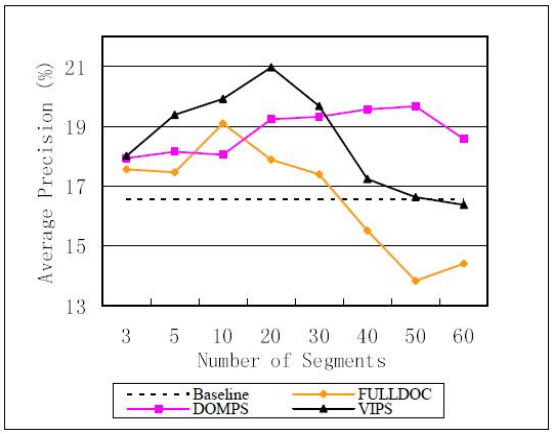

Fig 3. Average precision vs. Number of segments [1]

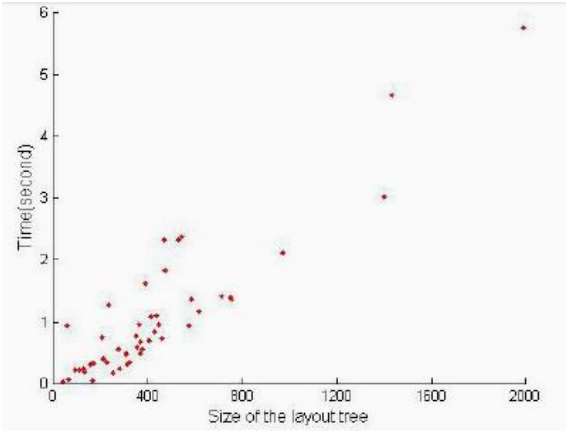

Fig 4(a). Time of page segmentation [2]

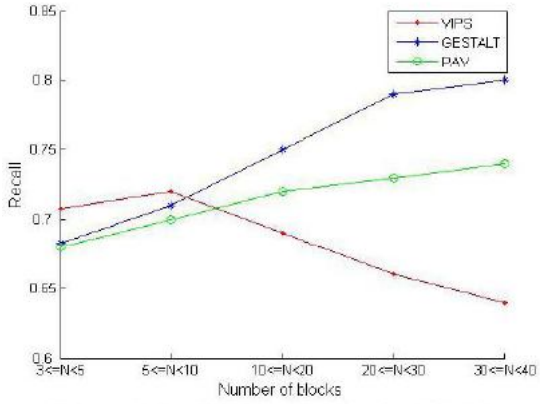

Fig 4(b). Recall of Gestalt, PAV and VIPS [2]

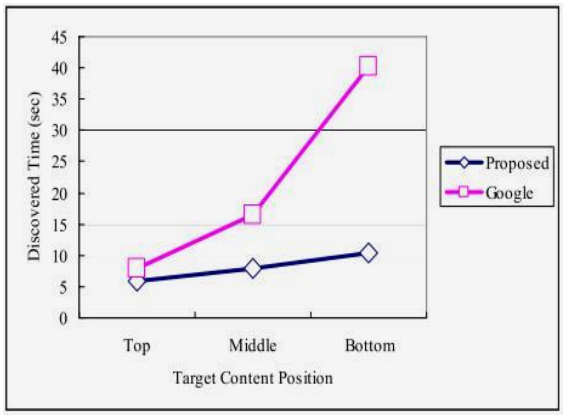

Fig 5(a). Usability Evaluation Results [17]

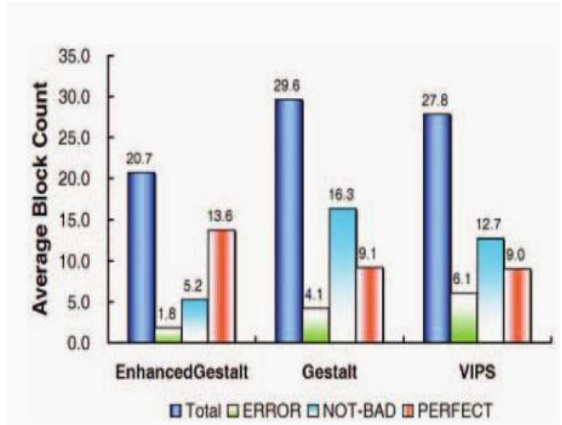

Fig 5(b). Block content of E-Gestalt, Gestalt and VIPS [18]