Web Content Adaptation for Internet-Enabled Mobile Handheld Devices

Wen-Chen Hu
Department of Computer Science
University of North Dakota
Grand Forks, ND 58202-9015
wenchen@cs.und.edu

Hung-Jen Yang
Dept. of Industrial Tech. Education
National Kaohsiung Normal University
Kaohsiung City 802, Taiwan
hjyang@nknucc.nknu.edu.tw

Lei Chen
Department of Computer Science
Sam Houston State University
Huntsville, TX 77341
LXC008@shsu.edu

Ruchitha Deshmukh
Department of Computer Science
University of North Dakota
Grand Forks, ND 58202-9015
ruchitha.deshmukh@und.nodak.edu

Abstract

The world has witnessed the blossom of mobile commerce in the past few years. Traditional Web pages are mainly designed for desktop or notebook computers. They usually do not suit the devices well because the pages, especially the large files, can not be properly, speedily displayed on the microbrowsers due to the limitations of mobile handheld devices: (i) small screen size, (ii) narrow network bandwidth, (iii) low memory capacity, and (iv) limited computing power and resources. Therefore, loading and visualizing large documents on handheld devices become an arduous task. Various methods are created for browsing the mobile Web efficiently and effectively. This paper investigates some of the methods:

- *Page segmentation*, which is used to segment Web pages.
- *Component ranking*, which is used to rank page components after segmentation.
- *Other ad hoc methods*, such as text summarization, transcoding, and Web usage mining.

Though each method employs a different strategy, their goals are the same: *conveying the meaning of Web pages by using minimum space.* The major problem of the current methods is that it is not easy to find the clear-cut components in a Web page. Other related issues such as mobile handheld devices and microbrowsers will also be discussed in this paper.

1 Introduction

Mobile commerce has drawn great attention these days and people start using mobile handheld devices such as smart cellular phones to perform all kinds of activities such as mobile Web browsing and instant messaging. According to Gartner, Inc., a market research company, the number of units of PCs, smartphones, and cellular phones shipped in 2008 are

- 302.2 million PCs including desk-based PCs, mobile PCs, and X86 servers (Gartner, Inc., 2009a),
- 139.3 million smartphones, which are mobile phones with advanced functions such as PC-like functions (Gartner, Inc., 2009b), and
- 1.22 billion mobile phones (Gartner, Inc., 2009c).

The number of smartphones shipped is increased fast in recent years and it is a little less than half of the number of PCs shipped. It is expected the number of smartphones shipped will surpass the number of PC shipped in the near future. When people started using handheld devices to browse the mobile Internet about ten years ago, Webmasters usually created two versions of their Web pages. One version using HTML is for desktop browsers and the other one using WML, cHTML, or other languages is for microbrowsers. However, this approach has been proved futile and time-consuming and most Web sites have only one version in HTML for both desktop browsers and microbrowsers today. Most Web pages are mainly designed for desktop or notebook computers. They usually do not suit the devices well because the pages, especially the large files, can not be properly, speedily displayed on the microbrowsers due to the limitations of mobile handheld devices: (i) small screen size, (ii) narrow network bandwidth, (iii) low memory capacity, and (iv) limited computing power and resources. Therefore, loading and visualizing large documents on handheld devices become an arduous task.

A wide variety of methods have been used for Web content adaptation for mobile handheld devices. This paper gives the challenges faced by these methods. It includes three themes:

- *Internet-enabled mobile handheld devices*: Mobile users browse the mobile Internet by using mobile handheld devices, which include six major components: (i) mobile operating systems, (ii) mobile central processing units, (iii) microbrowsers, (iv) input and output components and methods, (v) memory and storage, and (vi) batteries.
- *Microbrowsers*: Microbrowsers are a small version of desktop browsers such as Microsoft Internet Explorer and Firefox. They usually apply one of the four approaches to access the mobile Internet: (i) wireless language direct access, (ii) HTML direct access, (iii) HTML to wireless language conversion, and (iv) error.
- *Web content adaptation*: Various methods are used to browse the mobile Web and none of them is dominant. Most of them use the segmentation-and-ranking approach, that is, they display the page components in the order of their importance. This paper investigates some of the methods:
    - *Page segmentation*, which is used to segment Web pages.
    - *Component ranking*, which is used to rank page components after segmentation.
    - *Other ad hoc methods*, such as text summarization, transcoding, and Web usage mining.

  Though each method employs a different strategy, their goals are the same: *conveying the meaning of Web pages by using minimum space.* The major problem of the current methods is that it is not easy to find the clear-cut components in a Web page.

A related survey of Web content adaptation is also given by Alam & Rahman (2003).

2 Internet-Enabled Mobile Handheld Devices

Mobile users interact with mobile commerce applications by using small wireless Internet-enabled devices, which come with several aliases such as handhelds, palms, PDAs, pocket PCs, and smartphones. To avoid any ambiguity, a general term, mobile handheld devices, is used in this book. A mobile handheld device is small enough to be held in one hand and is a general-purpose, programmable, battery-powered computer, but it is different from a desktop PC or notebook due to the following three special features:

- *Limited network bandwidth*: This limitation prevents the display of most multimedia on a microbrowser. Though the Wi-Fi and 3G networks go some way toward addressing this problem, the wireless bandwidth is always far below the bandwidth of wired networks.
- *Small screen/body size*: This feature restricts most handheld devices to using a stylus for input.
- *Mobility*: The high mobility of handheld devices is an obvious feature that separates handheld devices from PCs. This feature also makes possible many new applications such as mobile recommendations that normally cannot be done by PCs.

Short battery life and limited memory, processing power, and functionality are additional features, but these problems are gradually being solved as the technologies improve and new methods are constantly being introduced. The limited network bandwidth prevents the display of most multimedia on a microbrowser. Though the Wi-Fi and 3G networks go some way toward addressing this problem, the wireless bandwidth is always far below the bandwidth of wired networks. The small screen/body size restricts most handheld devices to using a stylus for input. Figure 1 shows a typical system structure for handheld devices, which includes the following six major components: (i) mobile operating systems, (ii) mobile central processing units, (iii) microbrowsers, (iv) input and output components and methods, (v) memory and storage, and (vi) batteries, wich will be detailed next (Hu, et al, 2005). Synchronization connects handheld devices to desktop computers, notebooks, or peripherals to transfer or synchronize data. Without needing serial cables, many handheld devices now use either an infrared (IR) port or Bluetooth technology to send information to other devices.

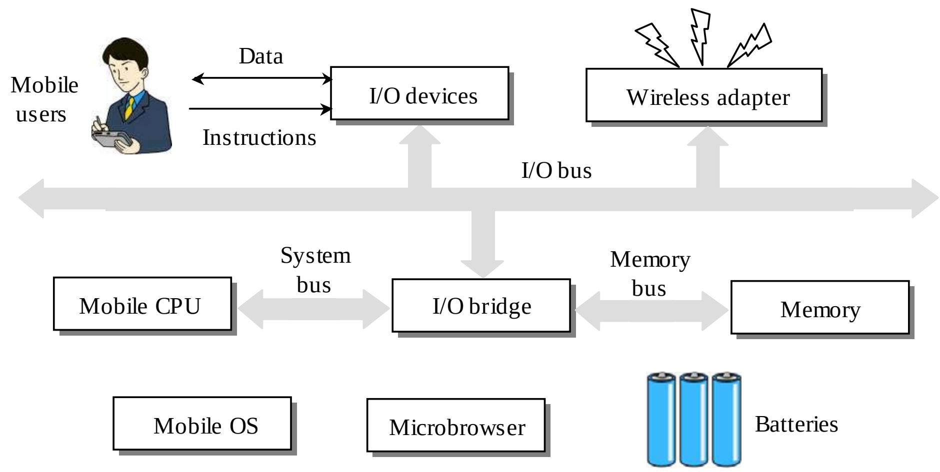


Figure 1: A System Structure of Mobile Handheld Devices.

2.1 Mobile Operating Systems

Simply adapting desktop operating systems for handheld devices has proved to be futile. A mobile operating system needs a completely new architecture and different features to provide adequate services for handheld devices. A generalized mobile operating system structure as shown in Figure 2 can be visualized as a six-layer stack: (i) applications, (ii) GUI, (iii) API framework, (iv) multimedia, communication infrastructure, and security, (v) computer kernel, power management, and real-time kernel, and (vi) hardware controller.

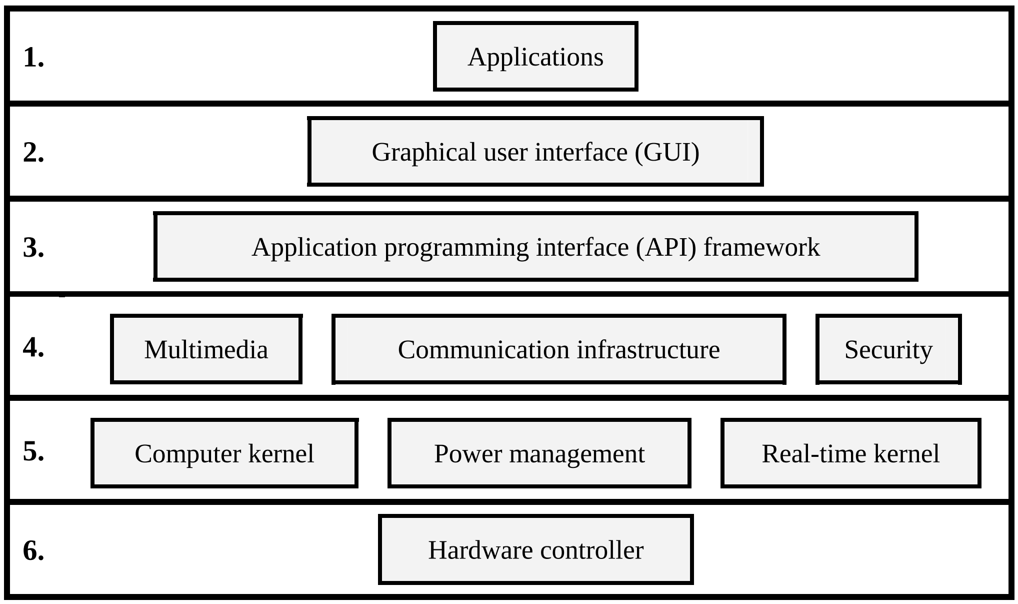


Figure 2: A Generalized Mobile Operating System Structure.

2.2 Mobile Central Processing Units

The core hardware in mobile handheld devices is the mobile processor, and the performance and functionality of the devices are largely dependent on the capabilities of their processors. There used to be several brands available, but recently mobile processors designed by ARM Ltd. have begun to dominate the market. ARM is the industry's leading provider of 32-bit embedded RISC microprocessors, with almost 75% of the market. Handheld devices are becoming more sophisticated and efficient every day and mobile users are demanding more functionality from the devices. To achieve this advanced functionality, in addition to the obvious feature, low cost, today’s mobile processors must have the following features:

- *High performance*: The clock rate must be higher than the typical 30 MHz for Palm OS PDAs, 80 MHz for cellular phones, and 200 MHz for devices that run Microsoft’s Pocket PC.
- *Low power consumption*: This prolongs battery life and prevents heat buildup in handheld devices that lack the space for fans or other cooling mechanisms.

- *Multimedia capability*: Audio/image/video applications are recurring themes in mobile commerce.
- *Real-time capability*: This feature is particularly important for time-critical applications such as voice communication.

2.3 Microbrowsers

Microbrowsers are miniaturized versions of desktop browsers such as Netscape Navigator and Microsoft Internet Explorer. They provide graphical user interfaces that allow mobile users to interact with mobile commerce applications. Microbrowsers usually use one of the following four approaches to return results to the mobile user: (i) wireless language direct access, (ii) HTML direct access, (iii) HTML to wireless language conversion, and (iv) error. Details of microbrowsers will be given in the next section.

2.4 Input and Output Components and Methods

Because of their size, handheld devices necessarily use different input and output components, methods, and strategies from those used by PCs:

- *Input components and methods*: Entering data into handheld devices is never an easy task because the devices are so small. Various input methods for handheld devices have been developed, the most important of which are: (i) keyboards, (ii) navigator, (iii) touch screens, (iv) writing areas on screens, and (v) speech recognition. Another input option that is often used is to receive data and files directly from PCs.
- *Output components and methods*: Although several alternative input devices and methods are available for handheld devices, the options for output devices and methods are more limited, with the main output component for a handheld being its screen. Handheld devices normally use synchronization technology to print data and files via PCs; handheld printers are available, but they are not common.

2.5 Memory and Storage

Desktop PCs or notebooks usually have between a few hundred Mbytes and a few Gbytes of memory available for users, whereas handheld devices typically have only few tens or hundreds of Mbytes. PDAs normally have more storage space than smart cellular phones, with the former commonly having 64 Mbytes, and the latter a memory size that may be as low as a few Mbytes. Four types of storage are usually employed by handheld devices: (i) flash memory, (ii) hard disks, (iii) random access memory (RAM), and (iv) read-only memory (ROM). Hard disks, which provide much more storage capacity, are likely to be adopted by handheld devices in the near future. Table 1 compares these four types of storage; a comprehensive survey of storage options can be found in Scheible (2002). Today's wireless devices demand higher memory throughput for more advanced features, such as Internet browsing, e-mail, data streaming, and text messaging.

	Capacity	*Erasable*	*Price Per Unit*	*Speed*	*Volatile*	*Writable*
Flash Memory	~ 5 GB	Yes	3rd	3rd	No	Yes
Hard Disks	~ 100 GB	Yes	4th	4th	No	Yes
RAM	~ 2 GB	Yes	1st (highest)	2nd	Yes	Yes
ROM	~ 1 GB	No	2nd	1st (fastest)	No	No

Table 1: A Comparison of the Four Kinds of Storage Available for Handheld Devices.

2.6 Batteries

Replaceable, rechargeable lithium-ion batteries are most commonly used in handheld devices. In smartphones using this kind of battery, the talking time, standby time, and full recharging time currently take a couple of hours, a few hundred hours, and a couple of hours, respectively, and the browsing time will be slightly shorter than the talking time. In the future, it should be possible to use handheld devices without the need to recharge them frequently by replacing the lithium-ion batteries with fuel cells, which although they are not yet practicable are likely to represent the best choice in the long-term. Table 2 provides a comparison between lithium-ion batteries and fuel cells, and detailed descriptions are given below.

	Contents	*Output*	*Type & Method*
Lithium-Ion Battery	Lithium ions	Electricity	Rechargeable using a power outlet
Fuel Cell	Natural gas	Electricity and water	Refuelable using fuel such as natural gas

Table 2: A Comparison between Lithium-Ion Batteries and Fuel Cells.

3 Microbrowsers

Microbrowsers are miniaturized versions of desktop browsers such as Netscape Navigator and Microsoft Internet Explorer. They provide the graphical user interfaces that enable mobile users to interact with mobile commerce applications.

3.1 Features

Due to the limited resources of handheld devices, microbrowsers differ from traditional desktop browsers in the following ways:

- smaller windows,
- smaller footprints, and
- fewer functions and multimedia features.

Several microbrowsers, such as Microsoft Mobile Explorer and Wapaka Java Micro-Browser, are already available. America Online, Inc., the parent company of the Netscape Network, and Nokia are developing and marketing a Netscape-branded version of Nokia's WAP microbrowser with AOL enhanced features for use across a wide variety of mobile handheld devices. Figure 3 shows a microbrowser, NetFront Browser v3.5 from ACCESS, which supports Visual Bookmarks—a pan & zoom navigation tool for the desktop-like presentation of web pages on mobile devices with limited screen size (ACCESS Co., Ltd., 2006).

Figure 3: NetFront Browser v3.5 © 2008 ACCESS Co., Ltd.

3.2 Technologies

Several markup languages are used to present mobile content on microbrowsers. These may not be able to handle all the languages currently used, therefore some content will not be displayed by some microbrowsers. Microbrowsers usually take one of the following four approaches, as shown in Figure 4, to display mobile content (Lawton, 2001):

1. *Wireless language direct access*: Here, a microbrowser supports some wireless languages, such as WML, CHTML, and XML, and directly displays any content written in a wireless language supported by that microbrowser.
2. *HTML direct access*: This approach displays the HTML contents directly, with no intervention, but may distort the content. For example, large images cannot be displayed on the small screens of microbrowsers.
3. *HTML to wireless language conversion*: Some mobile middleware provides conversion software that converts an HTML script to the script of a wireless language supported by that microbrowser. For example, i-mode includes a Corporate Conversion Server that converts existing HTML files into i-mode-compatible HTML, the CHTML.
4. *Error*: If a microbrowser is not able to handle the content, it displays an error code such as "Invalid WML code."

Some microbrowsers, like most desktop browsers, can automatically send and receive information with the aid of a cache, which is known as Web caching (Davison, 2001). Web caching offers significant advantages, such as reduced bandwidth consumption, server load, and latency. Taken together, these advantages make accessing the Web less expensive and improve performance. These three components unique to mobile handheld devices, namely mobile OSs, mobile CPUs, and microbrowsers, result in a significant difference between the performance of handheld devices and desktop PCs; the remaining components do not play such a crucial role.

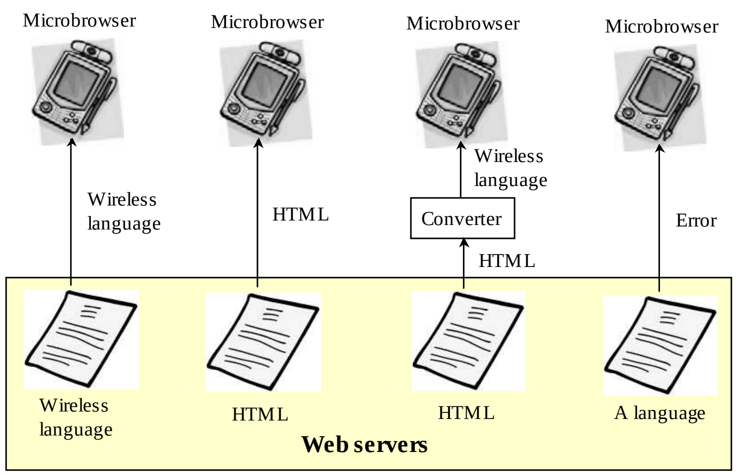


Figure 4: Four Approaches Used by Microbrowsers to Display Mobile Content.

3.3 Major Microbrowsers

A number of microbrowsers are currently available commercially. Four popular microbrowsers are: (i) Opera 8.65, (ii) Openwave Mobile Browser, Mercury Edition, (iii) Access NetFront Browser 3.5, and (iv) Microsoft Pocket Internet Explorer. Table 3 compares these four microbrowsers and detailed descriptions of the microbrowsers are given below. Some companies also provide microbrowser emulators/simulators such as Opera Mini Simulator that enable developers to test their products on desktop computers because small devices are not convenient for mobile application development.

	Mobile Browser 9.5	**Mobile Browser, Mercury Edition**	**NetFront Browser 3.5**	**Internet Explorer Mobile**
Vendor	Opera	Openwave	Access	Microsoft
Support HTML?	Yes	Yes	Yes	Yes
Support WML?		Yes	Yes	Yes if extra software installed
Major Technologies	Small-Screen Rendering	Progressive rendering of content	Smart-Fit Rendering™	Fit-to-Screen menu
Special Features	Flash	Ajax	Ajax	JScript

Table 3: A Comparison of the Four Leading Microbrowsers.

3.4 Two Examples of Built-in Web Content Adaptation

This sub-section gives two examples of how microbrowsers display Web content from the industry:

- *ACCESS*: ACCESS' NetFront Browser includes Smart-Fit Rendering technology (n.d.), which intelligently adapts standard web pages to fit the screen width of any mobile device enabling an intuitive and rapid vertical scrolling process, without degrading the quality or usability of the pages being browsed. Concretely, the following process is performed:
  - o Images larger than the screen width are scaled down to fit the screen width.
  - o Tables larger than the screen width are split and laid out vertically as shown in Figure 5.

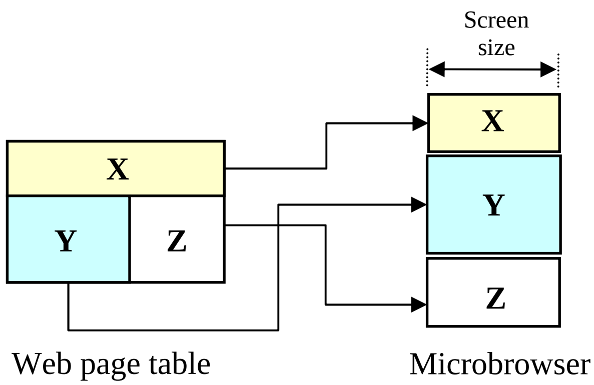


Figure 5: A Web Page Table Split by ACCESS' NetFront Browser.

- *Opera*: Opera's Small-Screen Rendering technology (n.d.) reformats Web page to fit it inside the screen width and eliminate the need for horizontal scrolling. All the content and functionality is still available, it is only the layout of the page that is changed. Figure 6 shows an example of the Opera's method.

(a) (b)

Figure 6: Screen Shots of Opera' Small-Screen Rendering™: (a) before Rendering and (b) after Rendering.

4 Web Content Adaptation

Most Web pages are designed for the use of desktop or notebook browsers like Microsoft Internet Explorer and Firefox in mind. When the pages are accessed from microbrowsers, they are distorted or not functioning fully or properly because many of their features such as images and Ajax are removed or disabled. Various methods are created to try to solve or relieve the problems. This section divides the methods into three categories: (i) page segmentation, (ii) component ranking, and (iii) other ad hoc methods, and each category will be detailed next. Though each method employs a different strategy, their goals are the same: *conveying the meaning of Web pages by using minimum space.*

4.1 Page Segmentation

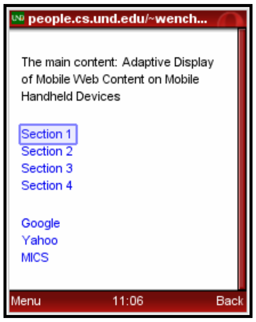
Mobile users usually are not interested in every detail of a Web page. For example, a typical commercial Web page usually includes three columns: the left navigation, the main content, and the right navigation/advertisements. Most mobile users would like the main content being displayed first if they have choices. The main idea behind the method of page segmentation is to display parts of Web pages instead of the whole pages when using microbrowsers. In order to realize this idea, Web pages need to be segmented. Several methods are designed to be used to segment Web pages (Gupta, Kumar, Mayank, Tripathi, & Tapaswi, 2007; Hua, Xie, Liu, Lu, & Ma, 2006); Xie, Miao, Song, We, & Ma, 2005; Chen, Ma, & Zhang, 2003). The most popular methods are to analyze the HTML source code and segment the pages according to the HTML tags. For example, Figure 7 shows a typical Web page and its corresponding HTML source code. The method studies the HTML code and re-organizes the columns, which use the HTML tags `<table>`, `<tr>`, `<td>`, and `<th>`. The central column is usually displayed first. Figure 8 shows the sample page after being re-organized and its corresponding HTML source code.

(a)

```
<html><body>
 <table width="100%" valign="middle"><tr>
  <td width="15%">
   <a href="Sec_1/">Section 1</a><br />
   <a href="Sec_2/">Section 2</a><br />
   <a href="Sec_3/">Section 3</a><br />
   <a href="Sec_4/">Section 4</a><br />
  </td>
  <td width="30%">
   The main content: Adaptive Display of Mobile Web Content
      on Mobile Handheld Devices
  </td>
  <td width="15%" align="right">
   <a href="http://www.google.com/">Google</a><br />
   <a href="http://www.yahoo.com/">Yahoo</a><br />
   <a href="http://www.micsymposium.org/">MICS</a><br />
  </td>
 </tr></table>
</body></html>
```

(b)

Figure 7: (a) A Sample Web Page and (b) the Corresponding HTML Source Code.




```
<html><body>
 <table width="100%" valign="middle">
  <tr><td>
    The main content: Adaptive Display of Mobile Web Content
       on Mobile Handheld Devices
  </td></tr>
  <tr><td>
   <a href="Sec_1/">Section 1</a><br />
   <a href="Sec_2/">Section 2</a><br />
   <a href="Sec_3/">Section 3</a><br />
   <a href="Sec_4/">Section 4</a><br />
  </td></tr>
  <tr><td>
   <a href="http://www.google.com/">Google</a><br />
   <a href="http://www.yahoo.com/">Yahoo</a><br />
   <a href="http://www.micsymposium.org/">MICS</a><br />
  </td></tr>
 </table>
</body></html>
```

(a) (b)

Figure 8: (a) The Sample Page of Figure 7.a after Being Re-Organized and (b) the Corresponding HTML Source Code.

However, this method suffers two major disadvantages:

- *The segmentation inaccuracy*: Page segmentation is to find Web page components. The problem is most page components are not clean-cut. Therefore, the segmented components may not be ideal.
- *The ranking problem*: This method tries to display the important parts of a Web page first. The question is how to define the importance of a Web component, which is ambiguous. Therefore, page segmentation is usually followed by component ranking, which will be explained next.

4.2 Component Ranking

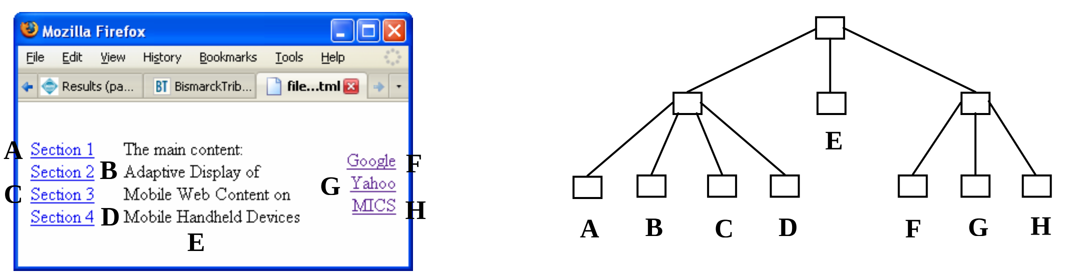
Page segmentation is usually followed by component ranking, which is used to rank the page components. So they can be displayed in the order of their importance. The following Web page features can be used to rank components:

- *Audio/figure/flash/table/video caption*: A caption is usually a description of the subject.
- *Content*: Web page content provides the most accurate and full-text information. However, it is also the least-used information for a search engine since content extraction is still far less practical.
- *Description*: Web page descriptions can either be constructed from the meta tags or submitted by webmasters or reviewers. A metatag is an HTML tag that provides information such as author, expiration date, a list of keywords, about a web page.
- *Distance*: Components closer to the central point of a page are usually more important than components far away from the central point.
- *Hyperlink text*: Hyperlink text is normally a title or brief summary of the target page.
- *Hyperlink*: Hyperlinks contain high-quality semantic clues to a page's topic. A hyperlink to a web page represents an implicit endorsement of the page being pointed to.

- *Keyword*: Keywords can be extracted from full-text documents or metatags. Filtering operations are applied to a document before retrieving keywords from the full-text document. Typical operations include the removal of common words using a list of stopwords, the transformation of upper-case letters to lower-case letters, etc.
- *Page structure*: HTML source code has a tree structure. Important information may be revealed from the structure. For example, the central column of a three-column table usually contains more important information than other two columns do.
- *Page titles*: The title tag defines the title of an HTML document.
- *Size*: Large-size components are usually more important than small-size ones.
- *Text with a different font, style, color, or size*: Emphasized text is usually given a different font to highlight its importance.
- *The first sentence*: The first sentence of a Web page is usually an introduction or an abstract.

Many methods are created to rank Web page components and each method is quite different from the others (Borodin, Mahmud, & Ramakrishnan, 2007; Hattori, Hoashi, Matsumoto, & Sugaya, 2007). The following example shows one of the methods and readers can check the references to find other methods. The example uses the PageRank algorithm, which is used by Google search, to rank page components (Yin & Lee, 2004). It performs the following tasks in sequence:

1. Segment a Web page and collect the page components.
2. Convert the Web page into a graph, whose nodes are page components and edges are relationships among components. Each edge is associated with a weight. For example, each paragraph could be a component and two consecutive paragraphs have an edge between them. Figure 9 shows a segmented page and its corresponding graph/tree. The root of this tree is the `<html>` element, which has three children: the left column, the central column, and the right column.



(a) (b)

Figure 9: (a) A Sample Page with Eight Components Labeled with Letters from A to H and (b) the Graph Corresponding to the Page.

3. The algorithm "PageRank" is then applied to the graph to find the ranks of page components or graph nodes. It analyzes the edges to uncover two types of pages:
   - *authorities*, which provide the best source of information on a given topic and
   - *hubs*, which provide collections of links to authorities.

   Two major steps are used to find the authorities and hubs and their weights:

a. *a sampling component*, which constructs a focused collection of several components likely to be rich in relevant authorities; and
b. *a weight-propagation component*, which determines numerical estimates of hub and authority weights by an iterative procedure:
    - *Authority weight update:* If a component is pointed to by many good hubs, we would like to increase its authority weight x_p, for a component p, by the sum of y_q over all components q that link to p.
    - *Hub weight update:* In a strictly dual fashion, if a component points to many good authorities, we increase its hub weight.

The authority and hub weights are then used to decide the component ranks.

4.3 Other Ad Hoc Methods

A wide variety of methods are created for Web content adaptation for microbrowsers. This paper is not possible to cover all of the methods. Other than the above two methods: page segmentation and component ranking, this sub-section lists some of the methods: (i) page summarization, (ii) transcoding, and (iii) Web usage mining. Some other methods can be found in the related articles, for example, multimedia adaptation (Maekawa, Hara, & Nishio, 2006; Laakko & Hiltunen, 2005), context-aware adaptation (Pashtan, Kollipara, & Pearce, 2003), RSS feeds method (Blekas, Garofalakis, & Stefanis, 2006), and grammar induction method (Kong, Ates, Zhang, & Gu, 2008).

4.3.1 Page Summarization

This method tries to display a summary of a large document on a microbrowser. Text summarization gives a short version of a document without losing its meaning. It has been a research topic for a long time and no major breakthrough was made in many years because of its high difficulty. Yang and Wang (2004) propose a fractal summarization for large documents. It is based on the theory of fractal, which is a geometric shape that is repeated itself under several levels of magnification. It generates a brief skeleton of summary at the first stage, and the details of the summary on different levels of the document are generated on demands of users. Otterbacher, Radev, and Kareem (2006) use the method of hierarchical summarization, which displays the most important sentences in an article first. If the reader finds the initial summary interesting or relevant, he/she may “drill down” the details of the story by expanding the message. The hierarchical summarization includes two stages. First, it identifies the salience of each sentence in a document and ranks the sentences accordingly. Second, it builds a tree of all sentences such that its root is the sentence with the highest salience.

4.3.2 Transcoding

Transcoding is to convert one document to another. For mobile Web browsing, transcoding tries to translate a Web document to another and expects the latter document will be better displayed on handheld devices compared to the former document. Hwang, Kim, and Seo (2003) develop a syntax-based Web transcoding system that allows universal access to Web pages without manual reauthoring. It is based on *structure-aware* transcoding heuristics, which preserve the original Web page's underlying layout as much as possible. The proposed heuristics extract the relative importance of Web components from an intelligent syntactic analysis and display them in the order of their importance. Hsiao, Hung, and Chen (2008) propose an architecture of versatile transcoding proxy (denoted by VTP) for Web content adaptation. In their framework, the proxy can accept and execute the *transcoding preference script* provided by the client or the server to transform the corresponding data or protocol according to the user's specification.

4.3.3 Web Usage Mining

World Wide Web Data Mining includes content mining, hyperlink structure mining, and usage mining. All three approaches attempt to extract knowledge from the Web, produce some useful results from the knowledge extracted, and apply the results to certain real-world problems. The first two apply the data mining techniques to Web page contents and hyperlink structures, respectively. The third approach, Web usage mining, is the application of data mining techniques to the usage logs of large Web data repositories in order to produce results that can be applied to many practical subjects, such as improving Web sites/pages, making additional topic or product recommendations, user/customer behavior studies, etc. Zhou, Hui, and Chang (2006) try to enhance mobile-browsing experience by using Web recommendations. Each user is observed as a unit of unknown identity, although some properties may be accessible from demographic data. A runtime component dynamically inserts recommended or related links into the top of each requested page. Therefore, their system can generate recommendations even for a new mobile user with no historical access records. A related research can be found in the article from Hu, et al. (2008).

5 Summary

Mobile commerce is a promising trend of commerce and mobile handheld devices are the mandatory tools for performing mobile commerce transactions. It uses microbrowsers to access the mobile content. However, many problems are associated with mobile Web browsing. This paper discusses various issues related to mobile Web browsing. The first issue is the study of mobile handheld devices, which include six components: (i) mobile operating systems, (ii) mobile central processing units, (iii) microbrowsers, (iv) input and output components and methods, (v) memory and storage, (vi) batteries. The component closely tied to mobile Web browsing is microbrowsers, which usually apply one of the four approaches to access the mobile Internet: (i) wireless language direct access, (ii) HTML direct access, (iii) HTML to wireless language conversion, and (iv) error. The last issue is the difficulty of mobile Web browsing. Various methods are created for browsing the mobile Web efficiently and effectively. Each method has its own advantages and disadvantages and none of them is dominant. Though each method employs a different strategy, their goals are the same: *conveying the meaning of Web pages by using minimum space.* This paper investigates some of the methods:

- *Page segmentation*, which is used to segment Web pages.
- *Component ranking*, which is used to rank page components after segmentation.

- *Other ad hoc methods*, such as text summarization, transcoding, and Web usage mining.

Most methods segment the Web page first and then display the components in the order of their ranks. The major problem of the current methods is that it is not easy to find the clear-cut components in a Web page.

References

ACCESS. (n.d.). *Small-Fit Rendering*. Retrieved June 14, 2008, from http://www.access-company.com/products/netfrontmobile/contentviewer/mcv_tips.html#Anchor-Smar-45765

Adipat, B. & Zhang, D. (2005). Adaptive and personalized interfaces for mobile Web. *Proceedings of the 15th Annual Workshop on Information Technolgies & Systems (WITS)*, Las Vegas, Nevada.

Alam, H. & Rahman, F. (2003). Web document manipulation for small screen devices: a review. *Proceedings of the 2nd International Workshop on Web Document Analysis (WDA2003)*, Edinburgh, UK.

Arase, Y., Maekawa, T., Hara, T., Uemukai, T., & Nishio, S. (2006). A Web browsing system based on adaptive presentation of Web contents for cellular phones. *Proceedings of the 2006 International Cross-Disciplinary Workshop on Web Accessibility (W4A)*, Edinburgh, U.K., pages 86-89.

Blekas, A., Garofalakis, J., & Stefanis, V. (2006). Use of RSS feeds for content adaptation in mobile Web browsing. *Proceedings of the 2006 International Cross-Disciplinary Workshop on Web Accessibility (W4A)*, Edinburgh, U.K., pages 79-85.

Borodin, Y., Mahmud, J., & Ramakrishnan, I.V. (2007). Context browsing with mobiles—when less is more. *Proceedings of the 5th International Conference on Mobile Systems, Applications, and Services (MobiSys'07)*, San Juan, Puerto Rico, pages 3-15.

Chen, Y., Ma, W.-Y., & Zhang, H.-J. (2003). Detecting Web page structure for adaptive viewing on small form factor devices. *Proceedings of the 12th International Conference on World Wide Web*, Budapest, Hungary, pages 225-233.

Davison, B. D. (2001). A Web caching primer. *IEEE Internet Computing*, 5(4):38-45.

Gartner, Inc. (2009a). *Gartner Says in the Fourth Quarter of 2008 the PC Industry Suffered Its Worst Shipment Growth Rate Since 2002*. Retrieved March 15, 2009, from http://www.gartner.com/it/page.jsp?id=856712

Gartner, Inc. (2009b). *Gartner Says Worldwide Smartphone Sales Reached Its Lowest Growth Rate With 3.7 Per Cent Increase in Fourth Quarter of 2008*. Retrieved March 18, 2009, from http://www.gartner.com/it/page.jsp?id=910112

Gartner, Inc. (2009c). *Gartner Says Worldwide Mobile Phone Sales Grew 6 Per Cent in 2008, But Sales Declined 5 Per Cent in the Fourth Quarter*. Retrieved March 19, 2009, from http://www.gartner.com/it/page.jsp?id=904729

Gupta, A., Kumar, A., Mayank, Tripathi, V. N., & Tapaswi, S. (2007). Mobile Web: Web manipulation for small displays using multi-level hierarchy page segmentation. *Proceedings of the 4th International Conference on Mobile Technology, Applications, and Systems*, Singapore, pages 599-606.

Hattori, G., Hoashi, K., Matsumoto, K., & Sugaya, F. (2007). Robust Web page segmentation for mobile terminal using content-distances and page layout information. *Proceedings of the 16th International Conference on World Wide Web*, Banff, Alberta, Canada, pages 361-370.

Hsiao, J.-L., Hung, H.-P., & Chen, M.-S. (2008). Versatile transcoding proxy for Internet content adaptation. *IEEE Transactions on Multimedia*, 10(4):646-658.

Hu, W.-C., Yeh, J.-h., Chu, H.-J., & Lee, C.-w. (2005). Internet-enabled mobile handheld devices for mobile commerce. *Contemporary Management Research*, 1(1):13-34.

Hu, W.-C., Zuo, Y., Chen, L., & Yang, C.-H. (2008). Adaptive mobile Web browsing using Web mining technologies. Memmola M. & Al-Hakim, L., editors, *Business Web Strategy: Aligning the Internet with Corporate Design*, Information Science Reference.

Hua, Z., Xie, X., Liu, H., Lu, H., & Ma, W.-Y. (2006). Design and performance studies of an adaptive scheme for serving dynamic Web content in a mobile computing environment. *IEEE Transactions on Mobile Computing*, 5(12):1650-1662.

Hwang, Y., Kim, J., & Seo, E. (2003). Structure-aware Web transcoding for mobile devices. *IEEE Internet Computing*, 7(5):14-21.

Jindal, A., Crutchfield, C., Goel, S., Kolluri, R., & Jain, R. (2008). The mobile Web is structurally different. *Proceedings of the 27th Conference on Computer Communications*, Phoenix, Arizona, pages 1-6.

Kong, J., Ates, K. L., Zhang, K., & Gu, Y. (2008). Adaptive mobile interfaces through grammar induction. *Proceedings of the 20th IEEE International Conference on Tools with Artificial Intelligence*, Dayton, OH, pages 133-140.

Lawton, G. (2001). Browsing the mobile Internet. *IEEE Computer*, 35(12):18-21.

Maekawa, T., Hara, T., & Nishio, S. (2006). Image classification for mobile Web browsing. *Proceedings of the 15th International Conference on World Wide Web*, Edinburgh, Scotland, pages 43-52.

Opera Software ASA. (n.d.). *Opera's Small-Screen Rendering*. Retrieved June 23, 2008, from http://www.opera.com/products/mobile/smallscreen/

Otterbacher, J., Radev, D., & Kareem, O. (2006). News to go: hierarchical text summarization for mobile devices, *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Seattle, Washington, pages 589-596.

Pashtan, A., Kollipara, S., & Pearce, M. (2003). Adapting content for wireless Web services. *IEEE Internet Computing*, 7(5):79-85.

Xie, X., Miao, G., Song, R., Wen, J.-R., & Ma, W.-Y. (2005). Efficient browsing of Web search results on mobile devices based on block importance model. *Proceedings of the 3rd IEEE International Conference on Pervasive Computing and Communications (PerCom 2005)*, Kauai, Hawaii, pages 17-26.

Yang, C. C. & Wang, F. L. (2003). Fractal summarization for mobile devices to access large documents on the Web. *Proceedings of the 12th International Conference on World Wide Web*, Budapest, Hungary, pages 215-224.

Yin, X. & Lee, W. S. (2004). Using link analysis to improve layout on mobile devices. *Proceedings of the 13th International Conference on World Wide Web*, New York, NY, pages 338-344.

Zhang, D. (2007). Web content adaptation for mobile handheld devices. *Communications of the ACM*, 50(2):75-79.