International Journal of Information & Network Security (IJINS)
Vol.3, No.3, June 2014, pp. 162~173
ISSN: 2089-3299

GreedyAdapter: Flexible Web Content Adaptation for Mobile Devices

Rajibul Anam*, Chin Kuan Ho*, Tek Yong Lim
* Faculty of Computing and Informatics, Multimedia University
Persiaran Multimedia, 63100 Cyberjaya, Selangor, Malaysia

Article Info

Article history:

Received Jun 12th, 2014
Revised Aug 20th, 2014
Accepted Aug 26th, 2014

Keyword:

Content Adaptation
Greedyadapter
Mobile Web Content
Smart Phone

ABSTRACT

Mobile web browsing usually becomes time-consuming moreover it requires horizontal and vertical scrolling. In addition to this, users interested in only a section of a web page are often burdened with cumbersome whole web contents that not only do not properly fit their mobile screens but also require a lot of delivery time. This problem can be addressed and resolved with the help of a mobile web content adaptation system. The system will enable users to access the target content faster on a mobile device. Existing web content adaptation systems focus on resizing contents to fit a mobile device and removing unnecessary contents from the adapted web page. To the best of our knowledge there is no web content adaptation system that rearranges the web contents to reduce the time taken for locating information on a web page. This research aims to address the gap by proposing a technique for web content adaptation. GreedyAdapter re-arranges the sequence of the blocks from a large web page while maintaining the relationship between the blocks with respect to content. It filters out unnecessary information by allowing users to see the most relevant blocks of a web page. In order to achieve this greedy algorithm is proposed. The proposed algorithm will first assign weights and profits to blocks of a web page. The algorithm will then select the best blocks based on a greedy heuristic and deliver them to handheld devices.

Corresponding Author:

First Author,
Faculty of Computing and Informatics, Multimedia University
Persiaran Multimedia, 63100 Cyberjaya, Selangor, Malaysia
Email: rajibul.anam08@mmu.edu.my

1. INTRODUCTION

The present era of 21st century has launched a new revolution of computing which is called mobile computing. Among all the mobile computing devices mobile phone is the most popular and mobile internet browsing is part of the ubiquitous computing. Over the last few years people have started using the network ready mobile devices like handheld computers, PDAs and smart phones to access the internet. Alexander Blekas [1] defines the term mobile device as a device specially designed for synchronous and asynchronous communication while the user is on the move [1]. The term “Mobile Web” was introduced when internet services for mobile devices started to use html based (W3C) standards for delivery [2]. However, mobile phones provide good mobility but very limited computational capabilities and screen size [3]. It is not pleasant for the mobile phone users to browse web pages via mobile devices because of low screen resolution, limited memory, processing speed and slow network [3], [4].

Since most of the existing web contents are originally designed to be displayed on the desktop computers, so the content delivery without layout adjustment and content adaptation makes the contents inapt for the mobile devices. Therefore, users need to scroll the screen horizontally and vertically to find the desired content, small screens provide less visible opportunities at any given time, requiring users to rely on their short term memory to build an understanding of web information [5]. Thus, each process of searching and browsing can be frustrating because most of the web sites are designed for the standard desktop display and thereby make most of these web contents unpleasant for mobile screen view [6].

The main purpose of this research is to create mechanism to adapt the web contents for smart phones. Many types of contents are used in the web to represent the information. So, this paper focuses on the techniques needed to adapt all kinds of text contents and also focuses on the techniques to change the html elements of the image, audio, flash and video contents in the web page so that the contents appear properly on the mobile screen.

In this paper, our proposed techniques which adapt web contents to improve mobile web browsing. The GreedyAdapter improves the web content accessibility by assigning profit to each object of a web page according to the user preferences. Then it assigns weigh to each object according to the properties of the object and changes the content resolution if required. It also uses the greedy algorithm to select the most profitable objects and generates subpages with the limit of the weights.

The rest of the paper is organized as follows. In section 2, provides detailed background of other existing deployed mobile content adaptation systems and discusses about the other adaptation frameworks. The proposed framework of GreedyAdapter, detailed comparisons with other existing systems are described in detail in section 3. In section 4, provides the analysis of the results of the experiments. Finally, presents the summary of the investigation and report in section 5.

2. REKATED WORKS

Many types of mobile web content adaptation systems have been deployed but only a few of them like Google Wireless Transcoder [7] and Skweezer [9] have been commercialized for public use and the techniques they use to deploy the systems are not open to public. On the other hand, in order to develop and improve the much needed mobile web content adaptation frameworks researchers over the years have applied different types of experimental methods of which some are automated and some semi-automated. The next section discusses both the deployed and experimental systems in detail.

2.1 Deployed Web Content Adaptation Systems

There are many deployed adaptation systems which not only contain most of the required adaptation features but also serve all kinds of web pages for mobile devices. The Google Wireless Transcoder (GWT) [7] is a mobile web content adaptation system useful in navigating to numerous web sites that are not optimized for mobile web browsing. The Google Wireless Transcoder is not only unable to deliver image and text on the same line but also doesn’t support any audio and video file. Moreover, it does not resize the images for adaptation [8]. Skweezer [9] is another type of mobile web content adaptation system. It adapts the web site which is designed for the desktop environment browsing and allows its users to access their favorite sites from mobile phones. Skweezer is unable to adapt image files like png, tif and tga, along with audio and video files. Therefore, these types of multimedia files don’t appear in the web pages after Skweezer adaptation. Thus, its main shortcoming is the failure to adapt some of the pages which contain these types of files [10].

2.2 Experimental Adaptation Frameworks for Mobile Devices

There are experimental web content adaptation systems have been developed over the years like the CMo, Annotation-based Web document Framework, Xadaptor, Web Page Tailoring Framework and Context-Aware Content Filtering & Presentation for Pervasive & Mobile Information System.

The CMo [11] uses proxy-based architecture to adapt web contents for handheld devices. This system reduces information overload and thereby allows its users to see the most relevant fragments of the web pages and to navigate between fragments if necessary. It captures the context of the links and applies a simple topic boundary detection technique to identify the contents. Moreover, it uses the context to identify the relevant information on the next page with the help of Vector Machine.

A web clipping technique is used in the Annotation-based Web document Framework [12]. Web clipping is a technique where the system extracts and represents some parts of an HTML document for the mobile browser. An HTML annotation or parsers generally declare the properties that qualify for a particular portion of a target document. The system annotates some parts of the web page and provides the annotated contents to the content adaptation engine. The web clipping technique modifies the HTML structure, breaks the page into smaller parts and creates separate new pages with titles. Once in a while, it removes unnecessary objects from the smaller new pages. The working principle of this method is to read the web page, tag some parts of the page and regenerate the web page for mobile web browsing.

Xadaptor [13] uses rule-based and fuzzy logic approaches to adapt web pages. The rule-based approach assists extensible, systematic and adaptive content adaptation. Moreover, it integrates adaptation mechanisms for various content types to categorize them. Rules are applied based on user defined preferences. The HTML objects are transformed into content and pointer objects and the system uses the content parser to separate the objects from the HTML markup. It also reformats the standard tables from the HTML objects by using fuzzy-rules. Xadaptor is a semi-automated system and the users need to assign some parameters to adapt the contents.

The Web Page Tailoring System [14] facilitates personalization of user's mobile web browsing. By using this system, a mobile user can easily retrieve the important section of a web page that he would like to read every day. Moreover, the sequences of the blocks are rearranged by the support of a vision-based page segmentation algorithm. This system uses java script to trace the sections of the web page. First, the user needs to configure the preferences of a web page by using the desktop computer. Then the user needs to configure from the mobile browser in order to use the system. Most of the steps for these adaptation frameworks are same but the techniques are different from each other.

3. PROPOSED SOLUTION

The GreedyAdapter is a semi-automated mobile web content adaptation system. The GreedyAdapter runs on a proxy server and all the adaptation process runs on the proxy server. This framework is called GreedyAdapter because greedy heuristics are used to select and deliver the web contents. The greedy algorithm always makes a choice that is best suited to a current state based on a cost function. When the user requests for a web page (Figure 1), the GreedyAdapter adapts the content and delivers it on the mobile screen. Figure 2 illustrates the framework of the GreedyAdapter. Firstly, according to a user’s request for a web page the GreedyAdapter downloads the page. Secondly the GreedyAdapter constructs an HTML tree from the web page. Thirdly, the algorithm identifies important blocks from the HTML tree. The profit assignment algorithm in turn assigns a profit to each object of a block. The pre-processing algorithm modifies the inappropriate object elements. The weight assignment algorithm assigns weight to each object of the HTML tree. The greedy algorithm selects the best profitable blocks from the HTML tree. Fourthly if the user opens a navigation link, then the GreedyAdapter sends the HTML tree to content identification after navigating to another page. Finally the GreedyAdapter sends the subpages to the user’s mobile devices.

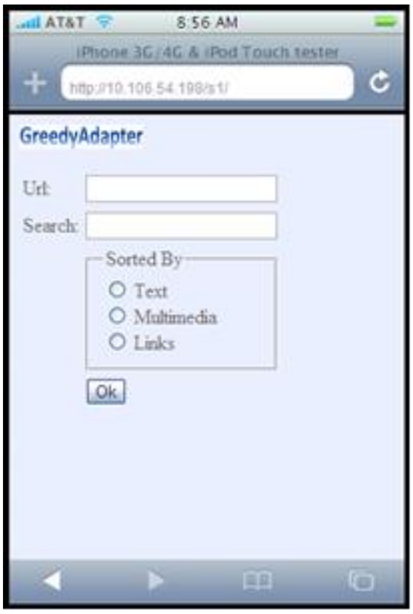


Figure 1. Screenshot of the GreedyAdapter page.

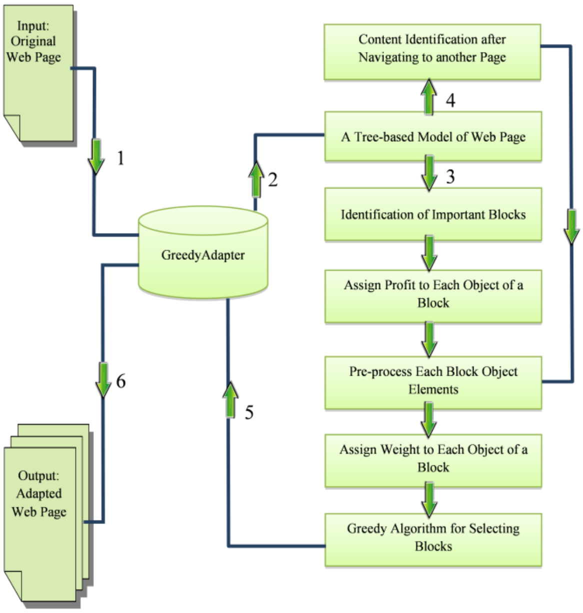

Input: Original Web Page
1
GreedyAdapter
2
Content Identification after Navigating to another Page
4
A Tree-based Model of Web Page
3
Identification of Important Blocks
Assign Profit to Each Object of a Block
Pre-process Each Block Object Elements
Assign Weight to Each Object of a Block
Greedy Algorithm for Selecting Blocks
5
6
Output: Adapted Web Page

Figure 2. The framework of the GreedyAdapter.

3.1 A Tree-Based Model of Web Pages

Hyper Text Markup Language (HTML) is the generic name for the group of languages that form the lingua franca of the World Wide Web [15]. All the web pages represent the contents in HTML format. The mobile web browsers also use the HTML objects to represent the contents. An HTML file can be easily converted into a tree structure and dynamic updates can be easily performed on the web page by modifying the tree. An HTML tree is defined as T = (V, E), where V is the set of vertex and E is the finite set of edges in the tree [16], [17]. Each node of the tree contains an HTML object.

3.2 Identification of Important Blocks

The HTML tree is then fed as input into an algorithm that identifies important blocks. This algorithm uses depth-first traversal method to traverse the tree. It identifies the important blocks (which are identified by using structure functionality [3] and only contain the important information [18]) by analysing their HTML elements and keeps the important blocks in the HTML tree [19]. If the algorithm identifies any object like Java Script and CSS (<script>, <style>) [20] then it prunes these objects from the html tree. These objects are not important for mobile viewing [10]. Figure 3 outlines the algorithm for identifying the important blocks. The input is an HTML tree *T*, the output is another tree *T* that contains only the important blocks, *v* is a vertex and *e* is an edge of the tree, *structure functionality* [3] is a set of HTML tags that helps to identify the important blocks. The algorithm traverses the branches of the html tree and searches for the structure functionality (refer to line 2). This algorithm filters the html tree and retains only the important block contents. Then it sends the html tree to the Assignment of Profit to Blocks.

```
Input:
            T is a tree;
Output:
            T contains only important blocks;
Visit(T,v)
1.    Perform visit the vertex v;
2.    If v match with structure functionality Then
3.      If v has neighbor Then
4.        For all neighbor e of v Do
5.          Visit(T,e);
6.          If e do not match with structure functionality Then
7.            Remove all the neighbor of e;
8.    Else
9.      Remove all the neighbor of v;
```

Figure 3. Algorithm for identification of important blocks

3.3 Assignment of Profit to Blocks

Figure 4 shows the selecting object algorithm and each object of the html tree contains a profit which is a positive numeric value. The profit is assigned according to the html element properties of the object. Figure 5 illustrates the profit algorithm where profit is assigned to the object if the object type matches with the user defined preferred objects like User Search Object, Text Object, Multimedia Object and Hyperlink Object [21].

```
Input:
            T is a tree;
Output:
            T with profit and weight;
Variables:
            v is a vertex;

Traverse(T,v)
Perform visit of the node v;
AssignmentOfProfit(v);
Preprocessing(v);
AssignmentOfWeight(v);
If v has neighbor Then
  For all neighbor e of v do
    Traverse(T,e);
```

Figure 4. Algorithm for selecting objects.

When the user uses the (User Search Object) search option (see Figure 1), the profit assignment algorithm assigns profit to the matched objects in the html tree. In Figure 5 (refer to lines 1-2) the algorithm shows that if the *search keyword matches* with the object then it multiplies *500* (this numeric value is used to assign high profit points because the user is searching for the target content) with the *number of matches* (the number of matched word in the object are counted. Sometimes one object contains more than one matched word) and then assigns profit points to v_p for the matched object.

A number of html elements are used to decorate and format [21], [26] the objects. All these types of html elements are used to make the contents more attractive to the readers as these types of contents contain foremost information. If the users select the text option (see Figure 1), then the algorithm assigns profit to the text object. Figure 5 (refer to lines 3-4) illustrates part of the profit assignment algorithm. If the algorithm finds any *decoration object* then it assigns *1* profit point to v_p for each matched html element because in most cases one object contains many *decoration* html elements.

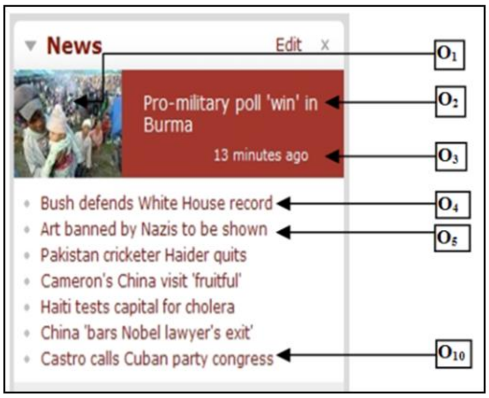
All the web pages use hyperlinks to enhance navigation to other sections of the web page or another web site. There are two kinds of hyperlinks available such as Independent and Dependent hyperlinks [21]. If the users select any of the link options (see Figure 1) the profit algorithm assigns profit according to the hyperlinks and *text length* of the object. In Figure 5 (refer to lines 6-7) the algorithm shows that it searches for the Dependent hyperlinks in the objects then it adds the text length of the hyperlink and *4* points to v_p as profit for the object. Then (refer to lines 8-9) the algorithm searches for the Independent hyperlinks in the objects and adds the text length of the hyperlink with *1* point (assigns low profit) to v_p as profit for the object.

If the user selects the multimedia option (see Figure 1) the profit algorithm assigns profit to the object according to the multimedia object's height and weight. Whenever the multimedia content pixels dimension is less, the profit is low but when this dimension is high the profit points are also high. Moreover, when the image width is less than 21 pixels this algorithm does not assign profit to the object because these images consider not. In Figure 5 (refer to line 11) the algorithm shows the object height (the height of the object) being multiplied with object width (the width of the object) and divided by 300 (variable 300 is the threshold to limit the multimedia profit points). Afterwards vp is assigned as profit for the object.

```
Input:
        v is a vertex of a tree;
Output:
        v with profit;
AssignmentOfProfit (v)
1.   If search keyword matches with v and preference enable Then
2.     Assign profit to $v_p = v_p + 500 \times number\ of\ match$;
3.   If decoration object matches with v and preference enable Then
4.     Assign profit to $v_p = v_p + 1$ ;
5.   If hyperlink object matches with v and preference enable Then
6.     If hyperlink is related to the other objects Then
7.       Assign profit to $v_p = v_p + 4 + text\ length$;
8.     If hyperlink is not related to the other object Then
9.       Assign profit to $v_p = v_p + 1 + text\ length$;
10.  If multimedia object matches with v and preference enable and multimedia object resolution > 20 Then
11.    Assign profit to $v_p = v_p + ( object\ height \times object\ width / 300)$ ;
```

Figure 5. Algorithm for assignment of profit to each object of a block.

Subsequently after the completion of all the profit assignment processes the system adds the total profits of all objects of a block to form a group profit and reassigns the profit of objects with this new value of group profit to each object in that block. So, in other words all the objects of the block carry the same profits.



News
Edit x
Pro-military poll 'win' in Burma
13 minutes ago
- Bush defends White House record
- Art banned by Nazis to be shown
- Pakistan cricketer Haider quits
- Cameron's China visit 'fruitful'
- Haiti tests capital for cholera
- China 'bars Nobel lawyer's exit'
- Castro calls Cuban party congress

Figure 6. Geometric order of the objects.

In the next step the system assigns profit according to the Geometric order of a web page which means the sequence of contents arrangement or display in a web page [11]. The Geometric order is very important to retain the original relationship among the objects in the block. Figure 6 illustrates the geometric order of the block objects of the web page where O_1 gets the highest profit because it appears at the top of the block. O_{10} gets the lowest profit because it appears at the bottom of the block. The system searches all the blocks and assigns profit to the objects by geometric order of a web page. After finishing the profit assignment process the html tree is sent to the Pre-Processing of Blocks.

3.4 Pre-Processing of Blocks

A number of HTML elements are used to format data and to highlight influential information for decoration purposes [3]. For mobile phone web browsing these kinds of supplementary formatted data presentation are not essential. Therefore, it is better to remove and edit these types of html elements. These kinds of objects need to be resized for better presentation and for effective web browsing on a mobile phone [22],[25]. The pre-process algorithm (Figure 7) first extracts all the html decoration elements and modifies the background colour of these elements from the html tree. Second, it edits the multimedia content dimension properties. If the multimedia content's height and width is more than mobile screen resolution then it reduces the multimedia content’s resolution and maintains the original aspect ratio of the multimedia content and reduces the resolution which is compatible to the mobile phone screen. After the completion of the pre-process assignment the html tree is sent to the Assignment of Weight to Blocks.

```
Input:
        $v$ is a vertex of the Tree;
Output:
        $v$ with new height and width;
Variables:
        height, width are the object height and weight;
        DeviceWidth, DeviceHeight are the mobile screen height and weight;
        200 is the threshold for an image;

Preprocessing($v$)
$Iscreen = (DeviceWidth / DeviceHeight) \times 200$;
If $v$ is decoration or highlight element Then
  Remove the element from $v$ and change the background color properties and reduce the font size;
If $v$ is multimedia element and multimedia height, width more than Iscreen Then
  $Ratio = height / width$;
  If $height > Iscreen$ Then
    $newheight = Iscreen$;
    $newwidth = height / Ratio$;
  Elseif $width > Iscreen$ Then
    $newwidth = Iscreen$;
    $newheight = width \times Ratio$;
update $v$ with the newwidth and newheight;
```

Figure 7. Algorithm for pre-processing the objects.

3.5 Assignment of Weight to Blocks

This algorithm assigns weight to each object according to text length and multimedia object’s resolution. If the algorithm encounters any text content in the object then it counts the text length, transforms its length into pixels and assigns weight to the object. Figure 8 (refer to lines 1-2) shows how the algorithm transforms the text length into pixels. Here v_w is the weight of an object, *10* is the pixel size of a single character, and *30* (pixels) is the threshold for each single line, *number of text character* is the number of characters in the object [13]. Next if it finds any multimedia object in the html tree it checks the object’s properties. If the multimedia object’s resolution is specified in the html element then it takes the resolution for the weight; in other respects the algorithm checks the multimedia content and gets the resolution by itself. Figure 8 (refer to lines 3-4) shows the algorithm where the *height of the image* is the height of the multimedia content and the *width of the image* is the width of the multimedia content. The v_w is the weight of the object which contains the area of the multimedia object. After the completion of this process the html tree is sent to the greedy algorithm.

```
Input:
        $v$ is a vertex of a tree;
Output:
        $v$ with weight;
AssingWeight($v$)
1.  If $v$ is text object Then
2.    Assign weight to $v_w = v_w + (30 + 10 \times number\ of\ text\ character)$;
3.  If $v$ is image object and image height or width pixels > 20 Then
4.    Assign weight to $v_w = v_w + (height\ of\ the\ image \times width\ of\ the\ image)$;
```

Figure 8. Algorithm for assignment of weight to each object of a block.

3.6 Greedy Algorithm for Selection of Blocks

The Greedy Algorithm picks the most profitable objects from the HTML tree and limits the objects' quantities in every subpage by the weight of the objects. The total weight of the objects can't exceed the mobile screen resolution (*height* × *width*) for each subpage. Thus, the greedy algorithm selects the most profitable objects according to the user defined preferences and the *weight* must be less than or equal to the mobile screen resolution.

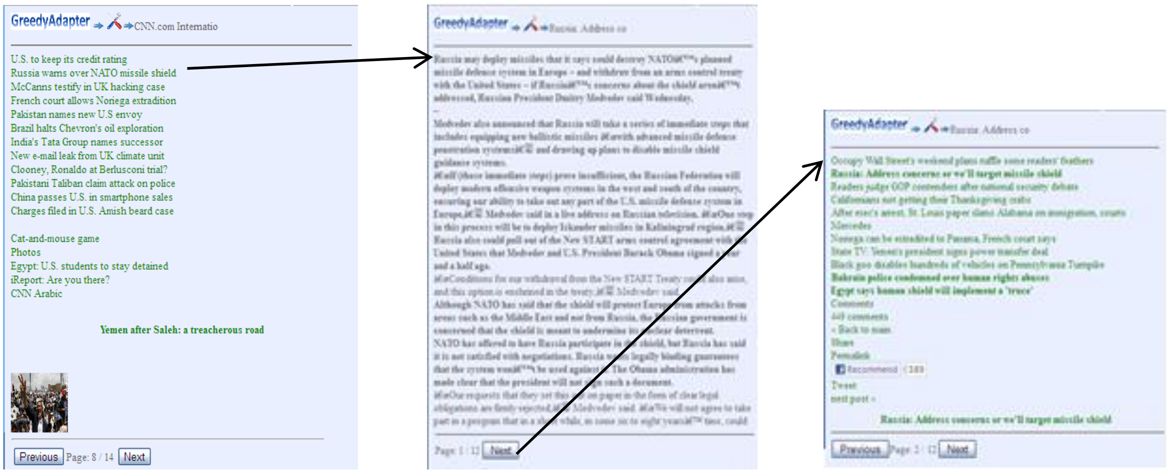
Figure 9 illustrates the greedy algorithm process where contentsi is the HTML tree object, wi is the weight and pi is the profit allocated by the ith item and A is the area based on the mobile screen resolution (height × width). The greedy algorithm selects the highest profitable objects with object weight lower or equal to A; the process stops when weight exceeds the area A (refer to line 1). The algorithm then delivers the selected contents in a subpage. The next and previous buttons are used to navigate the next and previous subpages. If the user selects the next page, then p selects the maximum profitable objects (refer to lines 2-3) and if the user selects the previous page, then p selects the minimum profitable objects (refer to lines 4-5). Thus, in this process the greedy algorithm creates subpages from the html tree. The first subpage contains the most important information for the users. And compared to this first subpage the second subpage contains less important information.

```
Input:
        contents are tree objects;
Output:
        subpage contain objects;
Variables:
        i is the number of offset;
        A is the resolution of the mobile screen;
        w is the weight;
        next is the next button;
        previous is the previous button;
        p is the profit;
Greedy( )
1.   while weight < A Do
2.     If next = true Then
3.       Do i = best new selected contents according by $p_i$ maximum value;
4.     Elseif previous = true Then
5.       Do i = best old selected contents according by $p_i$ minimum value;
6.     If weight + $w_i \leq A$ Then
7.       subpage = subpage append $contents_i$
8.       weight = weight + $w_i$
9.   i go to next offset until n;
```

Figure 9. Greedy algorithm for selecting blocks to display on mobile device.

3.7 Content Identification after Navigating to another Page

Some web pages only display the condensed information (as link) on the home page. In such cases users need to navigate to another page to read the full contents of the condensed information (link) [11]. The Greedy algorithm, which identifies the important contents from the newly opened page according to the navigation content of the first page, provides only the related contents to the user. This in turn helps users to access the target contents faster.


GreedyAdapter → → CNN.com Internatio

U.S. to keep its credit rating
Russia warns over NATO missile shield
McCanns testify in UK hacking case
French court allows Noriega extradition
Pakistan names new U.S envoy
Brazil halts Chevron's oil exploration
India's Tata Group names successor
New e-mail leak from UK climate unit
Clooney, Ronaldo at Berlusconi trial?
Pakistani Taliban claim attack on police
China passes U.S. in smartphone sales
Charges filed in U.S. Amish beard case

Cat-and-mouse game
Photos
Egypt: U.S. students to stay detained
iReport: Are you there?
CNN Arabic

Yemen after Saleh: a treacherous road

Previous Page: 8 / 14 Next

GreedyAdapter → → Russia: Address co

Russia may deploy missiles that it says could destroy NATOâ€™s planned missile defence system in Europe – and withdraw from an arms control treaty with the United States – if Russiaâ€™s concerns about the shield arenâ€™t addressed, Russian President Dmitry Medvedev said Wednesday.

Medvedev also announced that Russia will take a series of immediate steps that includes equipping new ballistic missiles â€œwith advanced missile defence penetration systemsâ€ and drawing up plans to disable missile shield guidance systems.
â€œIf (these immediate steps) prove insufficient, the Russian Federation will deploy modern offensive weapon systems in the west and south of the country, ensuring our ability to take out any part of the U.S. missile defense system in Europe,â€ Medvedev said in a live address on Russian television. â€œOne step in this process will be to deploy Iskander missiles in Kaliningrad region.â€
Russia also could pull out of the New START arms control agreement with the United States that Medvedev and U.S. President Barack Obama signed a year and a half ago.
â€œConditions for our withdrawal from the New START Treaty could also arise, and this option is enshrined in the treaty,â€ Medvedev said.
Although NATO has said that the shield will protect Europe from attacks from areas such as the Middle East and not from Russia, the Russian government is concerned that the shield is meant to undermine its nuclear deterrent.
NATO has offered to have Russia participate in the shield, but Russia has said it is not satisfied with negotiations. Russia wants legally binding guarantees that the system wonâ€™t be used against it. The Obama administration has made clear that the president will not sign such a document.
â€œOur requests that they set this out on paper in the form of clear legal obligations are firmly rejected,â€ Medvedev said. â€œWe will not agree to take part in a program that in a short while, in some six to eight yearsâ€™ time, could

Page 1 / 12 Next

GreedyAdapter → → Russia: Address co

Occupy Wall Street's weekend plans ruffle some readers' feathers
Russia: Address concerns or we'll target missile shield
Readers judge GOP contenders after national security debate
Californians not getting their Thanksgiving crabs
After exec's arrest, St. Louis paper slams Alabama on immigration, courts Mercedes
Noriega can be extradited to Panama, French court says
State TV: Yemen's president signs power transfer deal
Black goo disables hundreds of vehicles on Pennsylvania Turnpike
Bahrain police condemned over human rights abuses
Egypt says human shield will implement a 'truce'
Comments
449 comments
« Back to main
Share
Permalink
Recommend 183
Tweet
next post »

Russia: Address concerns or we'll target missile shield

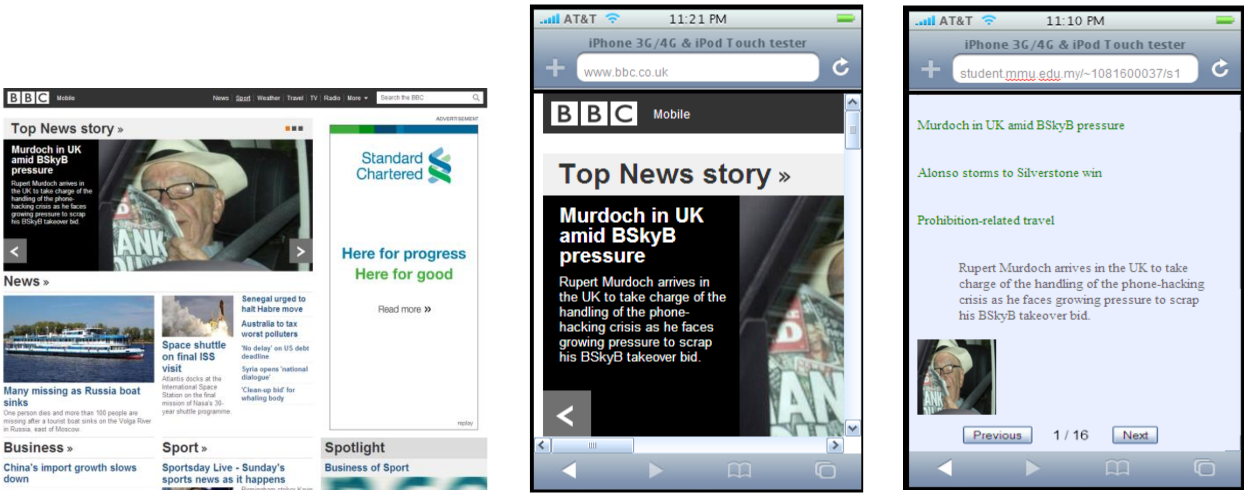
Previous Page 2 / 12 Next

Figure 10. (a) The adapted CNN page by GreedyAdapter on handheld before navigating "Russia warns over NATO missile shield", (b) the first subpage of adapted version of the navigated page, (c) the second subpage of the adapted page.

Figure 10.a shows the adapted home page of the CNN web site. Suppose a user selects the "Russia warns over NATO missile shield" link to read the main content this algorithm adapts the newly opened page, identifies the important blocks from the html tree and searches for the related contents "Russia warns over NATO missile shield" in the html tree. Then it selects those objects which match with the related content. Figure 10.b depicts the first subpage after the adaptation of the navigated page and Figure 10.c shows another subpage containing the contents related to "Russia warns over NATO missile shield". This algorithm first searches for all the text objects then assigns profit points if the objects' text matches with the navigation text. Secondly, the algorithm searches for the multimedia objects and assigns profit points if the objects' metadata (multimedia content) matches with the navigation text. After the completion of this process the html tree is sent to the Pre-Processing of Blocks.

3.8 The Implementation and Applications of GreedyAdapter

Figure 11.a displays the BBC home page as a standard desktop computer view and Figure 11.b shows the same web page in the iPod where a user needs to scroll in both directions to reach the target contents. Figure 11.c shows the BBC home page in the iPod after being adapted by the GreedyAdapter.


BBC Mobile News | Sport | Weather | Travel | TV | Radio | More ▾ Search the BBC

Top News story »

Murdoch in UK amid BSkyB pressure

Rupert Murdoch arrives in the UK to take charge of the handling of the phone-hacking crisis as he faces growing pressure to scrap his BSkyB takeover bid.

News »

Many missing as Russia boat sinks

One person dies and more than 100 people are missing after a tourist boat sinks on the Volga River in Russia, east of Moscow

Space shuttle on final ISS visit

Atlantis docks at the International Space Station on the final mission of Nasa's 30-year shuttle programme.

Senegal urged to halt Habre move
Australia to tax worst polluters
'No delay' on US debt deadline
Syria opens 'national dialogue'
'Clean-up bid' for whaling body

ADVERTISEMENT
Standard Chartered
Here for progress
Here for good
Read more »

Business »
China's import growth slows down

Sport »
Sportsday Live - Sunday's sports news as it happens

Spotlight
Business of Sport

AT&T 11:21 PM
iPhone 3G/4G & iPod Touch tester
www.bbc.co.uk

BBC Mobile

Top News story »

Murdoch in UK amid BSkyB pressure

Rupert Murdoch arrives in the UK to take charge of the handling of the phone-hacking crisis as he faces growing pressure to scrap his BSkyB takeover bid.

AT&T 11:10 PM
iPhone 3G/4G & iPod Touch tester
student.mmu.edu.my/~1081600037/s1

Murdoch in UK amid BSkyB pressure

Alonso storms to Silverstone win

Prohibition-related travel

Rupert Murdoch arrives in the UK to take charge of the handling of the phone-hacking crisis as he faces growing pressure to scrap his BSkyB takeover bid.

Previous 1 / 16 Next

Figure 11. (a) desktop version BBC homepage, (b) desktop version BBC homepage in mobile, (c) desktop version BBC homepage adapted by GreedyAdapter.

3.9 Comparison of GreedyAdapter with Experimental Frameworks

There are numerous significant differences between the proposed content adaptation techniques namely the GreedyAdapter and the other existing experimental content adaptation techniques. As a consequence the Xadaptor [13], WebPage Tailoring System [14] and CMo [11] have been selected for comparing the components. The proposed GreedyAdapter framework consists of five components. Table 1 depicts the comparison between the experimental framework components.

Table 1. Comparison of framework components

Components Used	(1) Xadaptor	(2) CMo System	(3) Web Page Tailoring System	(4) GreedyAdapter	Comments
User preferences	Yes	Yes	Yes	Yes	System 1 and 3 uses predefined user preferences from database but system 2 and 4 uses dynamic preferences from the users.
Transfer web page to Tree	Yes	Yes	Yes	Yes	System 1 use structure tree, system 2 use frame tree, system 3 use DOM tree and system 4 use simple html tree.
Block identification	Yes	Yes	Yes	Yes	System 1- 4 identify the blocks by using html tags.
Object identification	Yes	No	No	Yes	System 1and 4 identify the objects by using html tags.
Html elements modification	Yes	No	No	Yes	System 1 modifies the html elements according to the user database but system 4 modifies the html elements according to the mobile screen size (auto detected).
Block sequence re-arrange for display	No	No	No	Yes	System 4 re-arranges the blocks by the user preferences.
Mechanism to select the target contents for display	No	Yes	Yes	Yes	System 2 delivers the related information, System 3 delivers information according to the tag pattern matching and system 4 delivers exact information.

According to Table 1, unlike other frameworks that do not re-arrange the sequence of the blocks, the GreedyAdapter retains the original relationship among the objects, rearranges the blocks and then delivers them to the user's mobile phone. This feature, as stated earlier, helps the users to reach the target contents faster and thereby makes the GreedyAdapter more efficient and effective.

3.10 Time Complexity Comparisons of GreedyAdapter with Experimental Frameworks

This section discusses the time complexity of some algorithms involved in the GreedyAdapter and other frameworks. Table 2 illustrates the comparison of the algorithms where the first one is the blocks identification algorithm. In the GreedyAdapter blocks identification algorithm as seen in Figure 3 the complexity is $O(n)$ because the algorithm uses an iterative loop and recursive function where n is the number of vertex. The FindBlocks [11] algorithm uses iterative loops. So the complexity of the algorithm is $O(n)$, where n is the number of nodes in frame tree. Secondly, in the proposed objects identification algorithm the complexity is $O(n+e)$ because Figure 4, uses an iterative and recursive function, where n is the vertex and e is the number of edges. However, the FindContext [11] algorithm uses nested iterative loop. So the complexity is $O(n^2)$, where n is the number of nodes from leaf frame tree. Thirdly, according to Figure 6, in the object elements modification algorithm, where n is the number of vertex, the complexity is $O(n)$ and in the subpage navigation algorithm [24], where n is the number of nodes of the DOM tree, the complexity is $O(n)$. Fourthly, in Figure 7, the geometric height and width calculation algorithm complexity is $O(n)$ where n is the number of vertex and in the subpage navigation algorithm [24], where n is the number of nodes in DOM tree, the complexity is $O(n)$. When compared with other algorithms the over-all complexities of GreedyAdapter algorithms are almost the same. However, the object identification algorithm's complexity is less than the FindContext [11] algorithm.

Table 2. The time complexity comparison of the algorithms.

Algorithm	Time Complexity of Greedy-adaptor	Time Complexity of Other Systems
Blocks Identification Algorithm	$O(n)$	$O(n)$
Objects Identification Algorithm	$O(n+e)$	$O(n^2)$
Object Elements Modification Algorithm	$O(n)$	$O(n)$
Contents Geometric Height and Width Calculation Algorithm	$O(n)$	$O(n)$

4. USABILITY STUDY

This section describes the background of the usability study methods and the experiments. The study has been designed to investigate efficiency, effectiveness and users satisfaction (in some tasks situation) of both the GreedyAdapter and the Google Wireless Transcoder systems. The primary goal of this evaluation is to test whether the hypothesis can be supported by the GreedyAdapter or not. In the following sections the GreedyAdapter is referred to as GA and the Google Wireless Transcoder is referred to as GWT. The reason for selecting GWT is that it has user preference option. Moreover, it not only transforms all the web contents to fit the mobile phone screen but also converts long web pages into subpages. McGrath's [23] practical framework follows a simple structured approach to gather information and evidence. Moreover to conduct the test use the laboratory.

ACKNOWLEDGEMENTS

I am grateful to Mr. Zhang Li, the director of information system risk evaluation department of China Information Technology Security Evaluation Center. His knowledge and support has led to innumerable improvements of this paper. Thanks are also due to a number of my colleagues who generously read the paper entirety and to give me the benefit of their tremendous suggestions. Each of them deserves credit for the quality and style of this paper.

REFERENCES

[1] J GUAN, *et al*. "Knowledge-based information security risk evaluation method", Journal of China Universities of Posts and Telecommunications, v 20, pp. 60-63, December 2013

[2] ZHANG PEI, , *et al*. "Moving toward probabilistic reliability evaluation methods". International Conference on Probabilistic Methods Applied to Power Systems, AMES, USA, 2004, pp. 906 – 913.

[3] Wang yi, *et al*, "On the FAHP Method in Information Security Risk Evaluation", Computer Engineering & Science, Vol 28, No. 9, pp.2-4, 2006.

[4] Hongbo Xu and Xiaoguo Jiang, "Research on Business Types Recognition Based on the Method of AHP-ELECT RE", Computational Risk Management, Part 3, pp.275-283, 2011.

[5] KAABACAK B, *et al*, "Information security risk analysis method". C'omputers & Security, 2001. 24(2), pp147-159.

[6] Aagedal J, *et al*. "Model-based Risk Assessment to Improve Enterprise Security", 5th international EDOC conference, pp 51–62.

[7] NGAI T and WAT F. "Fuzzy decision support Systems for risk analysis in e-commerce development".Decision support system, 2005, 40(2), pp 235-255.

[8] WANG Lingyu, *et al*. "Toward measuring network security using attack graphs". Workshop on Quality of Protection. New York: ACM. 2007,pp.49-54.

[9] K. Ingols, *et al*, "Practical attack graph generation for network defense", Proc. of the 22nd Annual Conf. on Computer Security Applications, pp. 121-130, Dec. 2006.

[10] ISO/IEC 27005: 2008 "Information Technology – Security techniques - Information security risk management", Berlin: German Institute for Standardization, 2008.

[11] BOMIL S and INGOO H. "The IS risk analysis based on a business model". Information & Management 2003 (41), pp. 149–158.

BIBLIOGRAPHY OF AUTHORS

Tong Xin received the Ph.D. degree in information security from Beijing University of Posts and Telecommunications University in 2008. She is associate researcher of system evaluation division of China Information Technology Security Evaluation Center. Her interests are information system risk evaluation and communication security.

Ban Xiaofang is currently the vice director of system evaluation division of China Information Technology Security Evaluation Center. She received her BS degree in Computer Engineering from Chongqing University in 2005. She has very rich practical experience on information system security risk evaluation. Her scientific research direction: network security and information system risk evaluation.