Clustering Web Pages to Facilitate Revisitation on Mobile Devices

Jie Liu Chun Yu Wenchang Xu Yuanchun Shi
Department of Computer Science and Technology, Tsinghua University,
Beijing, P.R. China
{liujiejesse, yc2pcg, wencxu}@gmail.com; shiyc@tsinghua.edu.cn

ABSTRACT
Due to small screens, inaccuracy of input and other limitations of mobile devices, revisitation of Web pages in mobile browsers takes more time than that in desktop browsers. In this paper, we propose a novel approach to facilitate revisitation. We designed AutoWeb, a system that clusters opened Web pages into different topics based on their contents. Users can quickly find a desired opened Web page by narrowing down the searching scope to a group of Web pages that share the same topic. Clustering accuracy is evaluated to be 92.4% and computing resource consumption was proved to be acceptable. A user study was conducted to explore user experience and how much AutoWeb facilitates revisitation. Results showed that AutoWeb could save up a significant time for revisitation and participants rated the system highly.

Author Keywords
Mobile Web, Clustering, Revisitation.

ACM Classification Keywords
H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

General Terms
Human Factors, Algorithms

INTRODUCTION
People spend more time surfing the Internet on mobile devices nowadays. However, user experience on mobile Web browsing is still affected by many factors, such as small screens and low CPU performance. Small screens make it difficult to get a normal view of what is displayed and also affect the accuracy of input via pen taps or direct finger touches. When users want to revisit an opened Web page among a large number of previous pages, they have to scroll the long history list vertically or horizontally to find the desired one, which makes the revisitation frustrating.

IUI'12, February 14–17, 2012, Lisbon, Portugal.

To facilitate revisitation, searching scope should be reduced. We can put opened Web pages into groups with different topics according to their contents. When a user wants to revisit a previous Web page, s/he can first choose the group with the related topic and then in this group find the target. The scope of searching is narrowed down greatly. In that case, revisitation becomes topic-specific that needs less scrolling and navigation changes from one layer to two layers.

If we simply apply classification to mobile browsers in the same way as desktop browsers [8, 12], users have to classify opened Web pages manually, which in turn increases the input complexity. Instead, classification can be performed by browsers themselves, which is clustering. As a result, we design AutoWeb (Figure 1) to cluster opened Web pages into different groups on their contents to facilitate the revisitation. In AutoWeb, at revisitation, users first step into Group Level to choose a group, and then in In-Group Level, find the desired Web page.

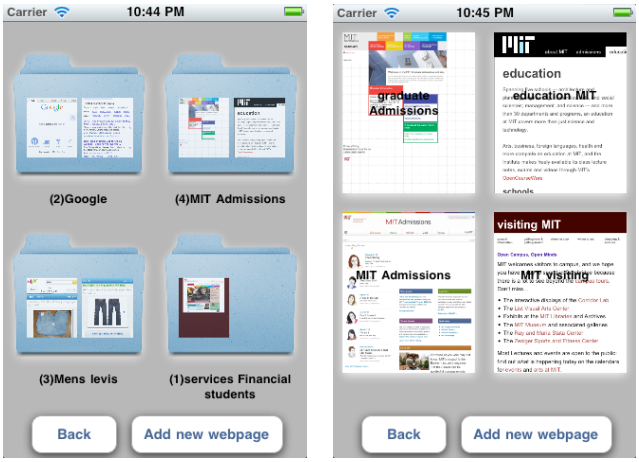

Figure 1. Revisitation in AutoWeb
(Left: Group Level, Web pages are clustered into groups. Right: In-Group Level, similar Web pages are in one group)

RELATED WORK
Various studies have been conducted to investigate mobile Web browsing and how to facilitate revisitation. We discuss them in both existing revisitation methods and clustering algorithms for Web pages.

Revisitation Methods

Usually, users resort to history lists, back buttons and bookmarks for revisitation. However, these tools can only help to revisit Web pages in a specific context [3, 7]. Many novel tools try to facilitate revisitation. Session Highlights [6] arranged Web page thumbnails in chronological order besides browser. Visual Snippets [11] presented a compact representation of Web pages to re-find previous pages. MacKay [8] and Wang [12] designed browser plugins to support multi-session tasks. Users manually managed tasks and Web pages.

Clustering Algorithms for Web Pages

Many algorithms have been proposed for clustering Web pages, such as link-based analysis algorithms, content-based analysis algorithms and usage-based algorithms. Link-based analysis algorithms use hyperlink information inside Web page source. If one page has a link to another page, these algorithms infer that these two Web pages are relevant to each other. But linkage information cannot show the sematic meaning of Web pages [5]. Content-based analysis algorithms use information retrieval methods to extract keywords from a set of Web pages and figure out the similarity among them. It is more accurate than the link-based analysis algorithms [9]. We use content-based analysis algorithm for clustering in AutoWeb.

AUTOWEB

We design AutoWeb to achieve the goal of facilitating revisitation for mobile browsing. In AutoWeb, we use two levels to represent opened Web pages. Group Level holds all groups of topics, and In-Group Level contains opened Web pages in each group sharing the same topic.

Considerations in Design

We choose iPhone as the platform. A recent report in May, 2011 reveals that iPhone tops first in mobile phones' network traffic in 10 out of 13 main countries [4]. Besides, it is easy for development using Xcode and distribution via App Store.

To help users identify Web pages quickly, an appropriate Web page representation is necessary. We use a hybrid way by combining thumbnails and text. Thumbnails give users visual cues and provide large touch areas for selection. Words used for clustering are also displayed on top of thumbnails to help users find the target. In Group Level, we use a folder as metaphor for a group of clustered Web pages. Smaller thumbnails also appear in the folder to indicate the content of this group.

As with the order of each opened Web page, we adopt the recency-based approach to sort groups and opened Web pages in each group. Recency is a strong reuse pattern when browsing the Web. It needs less cognitive effort when users search for newly visited sites on top of the list. When a Web page is loaded, a pop-up window will show the topic of this Web page for one second to give users first impression. It guide users match their mental models with clustering results during revisitation later on.

Clustering Algorithm in AutoWeb

Clustering is a computation intensive task. In consideration of limited computing resources on mobile phones, an efficient clustering algorithm is necessary.

We use vector-space model (VSM) to represent each Web page source as vectors in a multidimensional Euclidean space. The frequency of each word is counted. AutoWeb uses TF-IDF algorithm [1]. Term frequency $TF(d,t)$ is the number of times term t occurs in document d, which is $n(d,t)$. So, $TF(d,t) = \frac{n(d,t)}{\sum_{\tau} n(d,\tau)}$.

For Inverse document frequency $IDF(t)$, if D is the document collection and D_t is the set of documents containing t, $IDF(t) = \log \frac{|D|}{1+|D_t|}$.

Web page source is in a relatively structured format. HTML tags and metatags indicate different significance of terms. So we assign weights to them. Title, keywords and description in metatags weigh the highest. The first level headings have a medium weight, and bold words, headings in second level and below weigh the lowest. Hence, in vector space, the coordinate of document d in axis term t is given by $d_t = w_t TF(d,t) IDF(t)$, where w_t is the weight of t. Let $\vec{d}$ represent document d in vector space, then each Web page can be represented as $vec(d) = \{i : w_i TF(d,i) IDF(i)\}$.

After calculating TF-IDF value of an incoming Web page source, we apply single-pass incremental clustering to it. Although for each calculated Web page, its D and D_t may change by 1. This doesn't affect the clustering result greatly. So we don't re-calculate IDF values for the consideration of efficiency. The similarity of incoming Web page with existing pages in each cluster is computed. We calculate inter-document similarity $s(\vec{d}_i, \vec{d}_j)$ using cosine measure, which is $s(\vec{d}_i, \vec{d}_j) = \frac{\vec{d}_i \bullet \vec{d}_j}{|\vec{d}_i||\vec{d}_j|}$.

If the similarity value is above a given threshold, this Web page belongs to this cluster. If more than one cluster pass the threshold, all these clusters will contain this Web page so as to increase the hit rate of revisitation. A new cluster will be created when no existing clusters go beyond the threshold.

Clustering Procedure

When users input a URL or touch a link, a request for a Web page is issued. After getting the requested Web page, AutoWeb extract Web source and parse it in several steps.

Figure 2 shows the detailed procedure of clustering. We parse the source for text components using Document Object Model (DOM). Given the limited computing resources on mobile phones, we choose the most significant parts of text components, which are titles, keywords, description, bold words and headings. We remove stop words in these components. Porter Stemming algorithm [10] is employed to map words to the same stem. Then we assigned different weights to different components and calculate TF-IDF values for each selected terms in this Web page. After single-pass clustering, results are stored and ready for revisitation. Users interact with the system by a designed UI as illustrated in Figure 1.

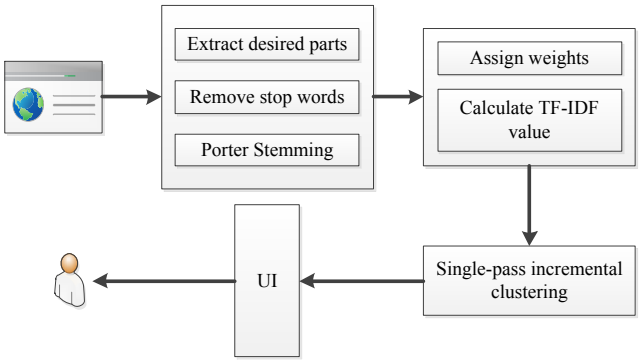

Figure 2. The Procedure of Clustering Web Pages

SYSTEM EVALUATION

We evaluated the system by processing Web pages related to four kinds of most frequent activities [2], that is *local services*, *travel*, *trivia* and *work/studies/hobbies*. We did these activities in AutoWeb with at least 50 related Web pages opened for each activity. We recorded the numbers of Web pages that were clustered into one correct group, more than one correct group, and wrong group. In the case that the Web page was clustered into several groups but not all these groups should hold this Web page, we considered it was clustered into wrong group.

Accuracy of Clustering

Figure 3 shows the results of evaluation. In every activity, about 80% of Web pages can be correctly clustered into one group of topic. Plus more than one correct group clustering, the accuracy is 92.4%. It also indicates that the kinds of activities don't exert a great influence on clustering accuracy. But, there are 7.6% of Web pages that were not placed into correct groups. We analyzed these Web page sources and found that they contained few feature words and less descriptive titles and headings, which were difficult for clustering. This is the drawback suffered by content-based analysis algorithms.

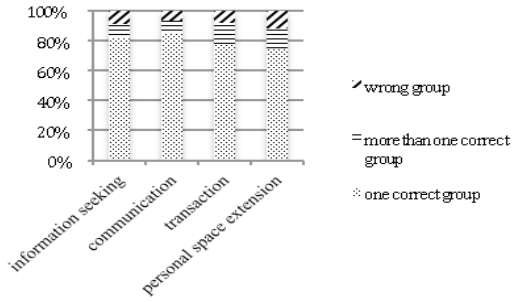

Figure 3. Clustering Accuracy in Each Activity

Computing Resources Consumption

We also validate the time and space consumption of AutoWeb. We use debug tool Instruments in Xcode to trace the memory usage at each load of Web pages. The elapsed time along with memory consumption is illustrated in Figure 4. In iPhone 3GS, time consumption is 119ms at the beginning and grow up to 181ms when 50 Web pages were opened. iPhone 4 has a faster CPU frequency so the time consumption started at 87ms and ended in 131ms. This latency isn't large enough to be noticed. At beginning, the system takes up 11300KB (11.0 MB) memory space and memory consumption increases with 80KB per Web page approximately. With 50 Web pages opened, the system used 15500KB (15.1MB) memory. So the system latency introduced by clustering and memory consumption of AutoWeb is acceptable.

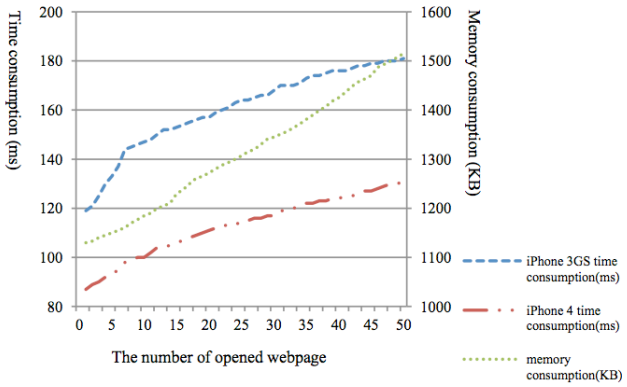

Figure 4. Time and Memory Consumption

USER STUDY

We compare AutoWeb with a conventional mobile browser called PlainWeb to log events. It only has back button and traditional history list for revisitation. We recruited 26 participants (9 females). Ages ranged from 21 to 32 years (*mean* = 22.4, *SD* = 4.5). They are active mobile Web users with iPhone. Before experiments, all participants were instructed how to use AutoWeb and PlainWeb. They practiced browsing Web pages with these browsers for at least half an hour until they got used to them.

To conduct the experiment in a controlled way, we designed three typical sets of tasks according to existing studies about mobile browsing content [2]. All participants fulfilled tasks using both AutoWeb and PlainWeb. 13 participants used AutoWeb first, and the others used PlainWeb first to receive counterbalanced results.

We investigate the average revisitation time in different tasks and browsers. The average revisitation time is the average time participants spent in re-finding each opened Web page. Figure 5 illustrated every participant's average revisitation time in two passes. There is a significant difference in AutoWeb and PlainWeb in terms of average revisitation time ($F(1, 24)=6.53$; $p<.001$). The result suggests that participants were able to revisit previous Web pages more quickly using AutoWeb. The average revisitation time of AutoWeb is 5.52sec (SD=.96sec), while the average revisitation time of PlainWeb is 7.83sec (SD=1.10sec). AutoWeb saves up to overall 29.5% revisitation time.

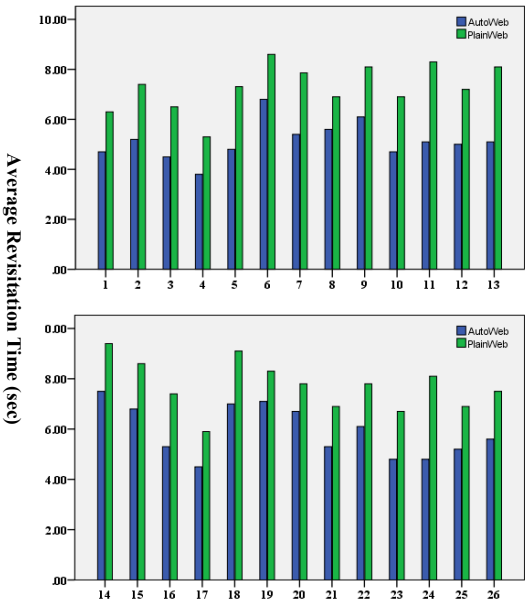

Figure 5. Average Revisitation Time for Each Participant

We also investigated participants' browsing experience with AutoWeb and PlainWeb. After finishing these three tasks, Participants filled out a questionnaire to rate AutoWeb and PlainWeb in a 5-point Likert scale.

As with ease of use, AutoWeb received an average rating of 4.13, while PlainWeb received 4.08. There is no obvious difference in ease of use between these two browsers. With the question, "Which browser do you like", AutoWeb received an average rating of 4.50, and PlainWeb receive 3.12. Mann-Whitney U Test show a significant differences between AutoWeb and PlainWeb (z=4.13, $p<.001$). It reveals that participant prefer AutoWeb.

CONCLUSION AND FUTURE WORK

Results presented in user study confirm that AutoWeb is easy to use and facilitates the revisitation for browsing mobile Web pages. AutoWeb introduces the concept of clustering according to Web topics into mobile browsers. This method proves effective and conducive to revisitation.

Future work includes adding the features that users can intervene in clustering process for more satisfactory results. Besides, a more extensive user study should be conducted.

ACKNOWLEDGEMENTS

This work is supported by National Natural Science Foundation of China, No. 61003005.

REFERENCES

1. Chakrabarti S. Mining the Web: Discovering Knowledge from Hypertext Data. 2002. pp. 56-57
2. Church, K., and Smyth, B. Understanding the Intent Behind Mobile Information Needs. In Proc. IUI 2009, pp. 247-256
3. Cockburn, A. and McKenzie, B. What do Web Users do? An Empirical Analysis of Web Use. In IJHCS, 2001(54), pp.903-922.
4. Device Essentials. http://www.comscore.com/Press_Events/Press_Releases/2011/6/comScore_Introduces_Device_Essentials
5. Dourisboure, Y., Geraci, F., et al. Extraction and classification of dense implicit communities in the Web graph. In ACM Trans. Web 2009;3(2)
6. Jhaveri, N. and Räihä, K. The Advantages of a Cross-Session Web Workspace. In Proc. CHI EA, 2005, pp. 1949-1952
7. Kellar, M., Watters, C. and Shepherd, M. A Goal-Based Classification of Web Information Tasks. In Proc. the American Society for Information Science and Technology, 2006, 43(1), pp. 1–22
8. MacKay, B. and Watters, C. Building support for multi-session tasks. In Proc. CHI 2009, pp. 4273-4278.
9. Nikolaev, K., Zudina, E., and Gorshkov. A. Combining anchor text categorization and graph analysis for paid link detection, In Proc. WWW 2009.
10. Porter, M., An algorithm for suffix stripping. 1980 http://tartarus.org/~martin/PorterStemmer/
11. Teevan, J., Cutrell, E.,et al. Visual Snippets: Summarizing Web Pages for Search and Revisitation. In Proc. CHI 2009, pp. 2023-2032.
12. Wang, Q., and Chang H. Multitasking Bar: Prototype and Evaluation of Introducing the Task Concept into a Browser, In Proc. CHI 2010, pp. 103-112