Personalized Detection of Fresh Content and Temporal Annotation for Improved Page Revisiting

Adam Jatowt[1], Yukiko Kawai[2], and Katsumi Tanaka[1]

[1] Kyoto University
Yoshida-Honmachi, Sakyo-ku, 606-8501
Kyoto, Japan
{adam, tanaka}@dl.kuis.kyoto-u.ac.jp
[2] Kyoto Sangyo University
Motoyama, Kamigamo, Kita-Ku, 603-8555
Kyoto, Japan
kawai@cc.kyoto-su.ac.jp

Abstract. Page revisiting is a popular browsing activity in the Web. In this paper we describe a method for improving page revisiting by detecting and highlighting the information on browsed Web pages that is fresh for a user. Content freshness is determined based on comparison with the previously viewed versions of pages. Any new content for the user is marked, enabling the user to quickly spot it. We also describe a mechanism for visually informing users about the degree of freshness of linked pages. By indicating the freshness level of content on linked pages, the system enables users to navigate the Web more effectively. Finally, we propose and demonstrate the concept of determining user-dependent, subjective age of page contents. Using this method, elements of Web pages are annotated with dates indicating the first time the elements were accessed by the user.

Keywords: page revisiting, fresh information retrieval, change detection.

1 Introduction

The Web is a very dynamic environment, with many changes occurring frequently. Several studies have confirmed this by measuring the frequency of Web changes (e.g. [2], [3] and [8]). This volatility makes the Web attractive and is one of the reasons for its great popularity. Users have access to the freshest news and information at any time. This is in contrast to traditional media like newspapers where readers have to wait certain periods of time for each new edition. Many Web users have favorite pages that they frequently revisit [4], [12]. Usually, such pages not only contain high quality content, but are also frequently changing since otherwise they would soon become uninteresting or even obsolete to users. This is because Web site administrators not only want to attract new users to their pages but also to encourage return visits. In general, the revisiting frequency is dependent on the page quality and the overlap of the page content with users' interests, but it is also related to the page-updating frequency.

S. Bressan, J. Küng, and R. Wagner (Eds.): DEXA 2006, LNCS 4080, pp. 832 – 841, 2006.

However, revisiting pages can sometimes be costly or can be a waste of time. Users coming to the page in search of new content may have problems noticing it especially if the page is large and changes are not easily visible. Often, top pages of popular Web sites (e.g. some news sites) introduce only short sentences containing links which lead to the novel content being published on separate pages. Noticing all such changes in the page may be troublesome and take time. Also, fresh information may be hidden in lower levels of a Web site's topology and thus be difficult to find. Users who wish to obtain new content have to access them one by one to check for new information. Such navigation is usually based on users' intuition and guesswork whether linked pages are worth revisiting, and may result in incurred costs and waste of time.

Sometimes users may even be misled to visit unchanged pages when expecting new content. For example, a page might have a link labeled "new," causing an average user to think that new content has been added since his last visit. The user might thus access this link only to find that the content is still old from his perspective. The "new" label is actually aimed at first time or infrequent visitors. Frequent visitors are expected to remember all the places they previously visited if they do not want to waste time viewing the same content twice. However, in contrast, some of the content on such a page may actually be new from the revisitor's perspective, but, having already visited the page, he or she may not visit it again since he or she may believe that the page content has not changed. In both cases, page viewing is ineffective, and the user can become frustrated. In the first case, the user might revisit the page too often, thereby losing time and incurring costs, while in the second case, he or she might miss content updates. One solution is a personalized, freshness-oriented browsing style due to which the user is clearly informed, without much effort, about the location and amount of fresh for him or her content on visited pages or sites.

In this paper we propose an approach to support browsing by automatically detecting the content that is fresh for the user. This user-dependent freshness determination is made possible by storing and analyzing the pages previously viewed by the user. We have built a browsing system that not only indicates the page elements that are novel for the user but also calculates the freshness of links occurring on the page and displays them in different colors based on their freshness values. The browser enables users to easily find content that is new to them and the links worth visiting. It can be especially helpful for users who have problems remembering previously seen versions of pages and who cannot easily spot changes in the page content. In addition, such browsing and navigation aid could be used for mobile browsing scenarios, where screen limitations may not enable viewing the whole page and limited bandwidth may not allow for unrestricted browsing of Web sites.

We also propose the concept of user-oriented temporal annotation of page content. Using this method, a user can know when he or she viewed certain content on a page for the first time and thus can determine how obsolete or how new it is for him or her. It is possible to recall the time when a given part of the page has been seen for the first time, hence putting a temporal constraint on it. If a part of the page content was seen by the user for the first time at a certain time point t_s, then during later visits to the page the user can be informed that the content is at least not younger than t_s. Thus the user can treat the particular content differently based on the date it was first encountered. Additionally, considering the number of page revisits since t_s, it is also possible to approximately assess how much the content is already known to the user.

In general, we provide a new kind of contextual information to users that is determined by considering their browsing histories, informing them on what they have not yet viewed and on what and when they have seen. The enhanced browsing is provided to reduce the cost and time spent revisiting pages without requiring much effort from the user. The user-oriented temporal annotation of page content enables users to obtain the information about the subjective age of page objects and allows one to better understand and orient themselves in the present content of pages. The proposed concepts are designed for users who do not have much time for browsing but frequently visit favorite pages that are highly volatile.

After first discussing related research in Section 2, we will describe in Section 3 our proposals for the user-oriented freshness detection and temporal annotation of page contents. In Section 4, we discuss their implementation. Finally, we conclude in Section 5 with a summary of the key points.

2 Related Research

Browsing Web pages is one of the most popular activities on the Internet. Users browse not only for relevant information but also, in many cases, for fresh content. However, there has been little research so far, of which we are aware, into combining change detection with browsing to facilitate fresh information retrieval. The exception is WebGuide system [7], which enables users to compare differences between pages with respect to two dates and to visualize changes in Web sites. Our approach is different in that we focus on integrating change detection with browsing. We propose a novel visualization method of fresh content on pages by utilizing link color changes and by displaying numerical values of freshness degrees of Web pages. This enhances navigation in revisited pages with minimal user interaction. The user is also informed about the overall freshness degree of the page content. Finally, we propose a new time-based annotation method with the user-oriented dates of content viewing.

Improving browsing by utilizing a user's browsing history has been already researched before (e.g. [9]). The pages that a person has visited constitute an easy to use set of data and are quite effective at measuring person's interests. However, rather than searching for content similar to previously viewed content that the user might be attracted to, we try to detect the fresh content for the user. Additionally, we utilize the browsing history for determining the user-oriented age of page objects. This differs from previous proposals of browsing history visualization in that it maps the user browsing activity directly onto the present content of the page.

So-called current awareness systems (e.g. [1], [6] and [11]) are tools for informing users about page updates. A popular one, Rich Site Summary (RSS), is a Web feed that provides summaries of the new content on Web sites. This enables users to track updates to sites. Other systems detect content changes in some pre-determined sets of resources and notify users about such changes. For example, a work by Qiang et al. [11] contains several effective approaches into detecting changes and measuring change importance in Web structures. Usually, current awareness systems require users to specify beforehand pages of interest. However, users may have trouble listing all such pages. Additionally, these systems may cause an information overload by continuously sending information about new content appearing on the specified

pages, especially in cases of highly volatile pages. While the user could theoretically specify the exact types of content that he or she is interested in, in order to minimize this overload, doing so would be difficult and impractical. Content filtering cannot be done effectively as users are often interested in novel content, and which content will be interesting for them is hard to predict. Furthermore, many users are accustomed to actively viewing the Web, and they often actively revisit pages without waiting for an alert from change detection systems. In our approach, we integrate change detection with browsing, thus eliminating the burden of registration of interesting Web pages by the user. Consequently, the proposed system detects browsing-derived changes extracted between consecutive visits of pages by the user.

3 System Overview

3.1 Freshness Degree

Let us suppose that a user has a favorite page, which he or she frequently revisits. For each visit, we store a view $V(t_i)$ of the page which is a snapshot of the page, where t_i denotes the timestamp of the view. The snapshot reflects the state of the page as it was perceived by the user and contains elements j ($j \in V(t_i)$) that occur on the page view $V(t_i)$. After several revisits of the page $V(t_1),\ldots,V(t_n)$, at time points $t_1,\ldots,t_n$, three attributes $(\tau_{ins}(j), \tau_{del}(j), \omega(j))$ are assigned to each element j at each page view; $\tau_{ins}(j)$ is the time point when the user saw the particular element for the first time, $\tau_{del}(j)$ denotes the time point when the user saw it for the last time and $\omega(j)$ specifies the element's viewing frequency. Representing each element occurring in a certain view $V(t_i)$ of the page by the triple $(\tau_{ins}(j), \tau_{del}(j), \omega(j))$ enables the system to determine how obsolete or well-known the element is for the user. The perception of the page contents from the perspective of the user-oriented freshness is thus changing every time he or she accesses the page.

The freshness degree of the page is computed as the ratio of the amount of fresh content to the total amount of the page content. We represent it as a linear combination of freshness degrees of text and images:

$$\text{TotalFreshness} = \alpha * \text{TextFreshness} + (1- \alpha) * \text{ImageFreshness} . \quad (1)$$

The amount of textual content is expressed as the number of words while the amount of image content is estimated by the dimensions of images on the page.

In addition, we can try to estimate the level of surprise of the user upon not seeing certain content on the page that is related to the long-term perception of the page content:

$$Surprise = \frac{\sum_{j=1}^{M} \left[size(j) * \left(t_{n-1} - \tau_{ins}(j) \right) * \omega(j) \right]}{\sum_{j=1}^{N} \left[size(j) * \left(t_{n-1} - \tau_{ins}(j) \right) * \omega(j) \right]} . \quad (2)$$

The surprise is related to the expectation of the user that no change will happen on the page. This can be expressed by using the time period that elapsed since the first view of the page content and the number of times it has been viewed by the user. Equation 2 is computed using the content of the current page version $V(t_n)$ and the last-visited page version $V(t_{n-1})$; $size(j)$ is the size of an element j and N is the number of distinct elements on $V(t_{n-1})$, M is the number of elements that were deleted since the last view of the page. The surprise level calculated in this way could be used to modify the total freshness degree of the page in order to represent more personalized, subjective freshness degree of the page.

A reader should note that the relevance of content is not taken into consideration in the proposed system. We assume that the content on frequently revisited or favorite pages is most likely interesting to the user. However, a relevance-aware freshness degree could be computed, for example, by considering the frequency of query words in the novel content or its overlap with the model of user interest.

3.2 Fresh Content Detection and Indication

When the user revisits a page, the present version of the page is compared with the recently visited one. Any added content is marked to draw the user's attention. For example, the background color of the text may be changed. Except for content comparison of the currently viewed Web page, there is a mechanism provided that estimates the degree of freshness for each page to which the current page has links. Then each link has assigned the degree of freshness which is shown next to the text in the anchor tag of the link. Additionally, link color is changed to indicate its freshness degree. Link-color changing has been used on the Web for some time. According to some estimate [10], 74% of Web sites use a link-color changing mechanism. Its prime aim is to inform users about the links that have been already visited by them. Hence, its main objective is to assist users with Web navigation rather than to inform them about the freshness of the content on the previously visited pages. Even if the contents of a previously visited page have been changed since the last visit, the corresponding link will still be displayed in the visited-link color. Therefore, this mechanism just informs the user about the pages he or she has visited without providing any information about changes to their contents. Thus, a user will not become aware of any changes in the content for a page already visited if he or she deems the page not worth revisiting by simply judging its freshness by the changed color of the link. Moreover, the marking process is an either/or process, meaning that only two colors are used to differentiate between visited and unvisited links. Our user-oriented freshness detection approach with link freshness visualization overcomes this problem. Below is the summary of the whole algorithm.

1. The pages viewed by the user are stored in a cache or database during browsing.
2. The current version of the accessed page is compared with the latest version viewed by the user.
3. Any added content is color-coded.
4. If any links on the current version of the accessed page have been already visited by the user then the current versions of the linked pages are compared with the latest versions viewed by the user.

5. For all linked pages that the user has viewed, the added contents are identified, and the pages' degrees of freshness are calculated. The links to these pages are shown in different colors depending on their freshness degrees. Additionally, freshness degrees and dates of their last visits may be displayed. The links for pages that have not been viewed by the user are left unchanged.

Figure 1 illustrates the steps involved in freshness-based annotating of the content and links. The links are shown in different colors depending on their corresponding degrees of freshness. Additionally, the text boxes indicating values of the freshness degrees are attached to the links. The user can spot which parts of the page are new to him or her and also see which linked pages contain large amounts of new content. The system thus facilitates the detection of fresh information in currently viewed pages and, at the same time, directs the user to fresh content on the linked pages.

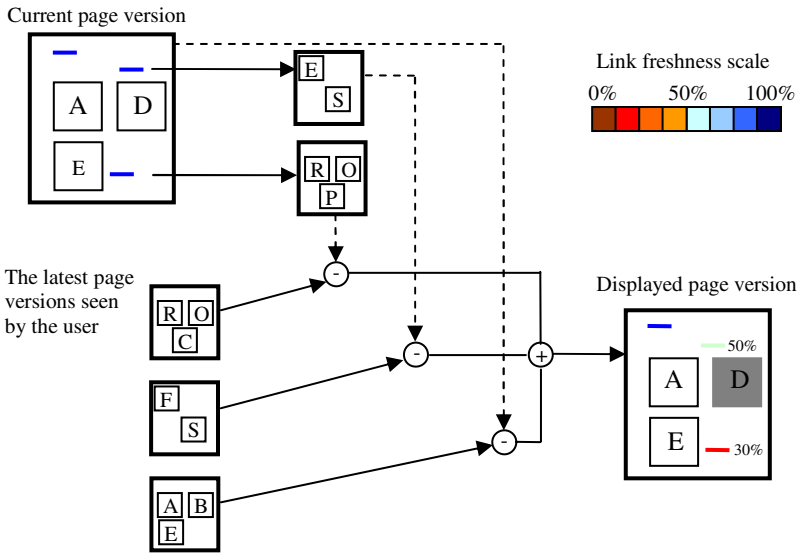

Fig. 1. Personalized, freshness-oriented annotation of page contents and links

The process can be fine tuned, for example, by setting a minimum time for considering the page content to have been viewed; similar to the process used in some mail clients, such as for example Microsoft Outlook. A scrolling-aware mechanism can also be implemented to categorize page parts into those seen and unseen by the user. In addition, page content viewed more than a certain period of time ago can be considered as new or partially new again by assigning time-dependent weights to the stored page versions. Finally, freshness degree can be propagated between pages and thus the freshness rate of larger page structures can be determined. For example, we may calculate the total freshness rate of the site or its part. In our implementation, however, we limit the change detection down to one level.

3.3 Detection and Visualization of Subjective Age of Page Content

Upon request the system visualizes the user-oriented, subjective age of different objects of page content. Elements on the current view of the page have dates $\tau_{ins}(j)$ attached denoting the timestamps of the particular page views when the user saw the elements on the page for the first time. To compute these dates the system does a search in the stored sequence of page views to find the earliest page view when particular elements first appeared. It compares the stored page views with the current version of the page for detection of overlapping content. The timestamp of the oldest page view containing a particular element is considered as the origin date, $\tau_{ins}(j)$. Two kinds of search algorithms can be used here: sequential and binary search. Sequential search is more effective for relatively short browsing histories of pages with few past page snapshots while the binary method works better for larger amounts of past data.

A special approach may be applied for processing links. Links can have two kinds of user-oriented dates: one is the date of seeing the anchor text of the link for the first time on the page while the other is the date when the linked page was actually accessed by the user.

By the user-oriented age visualization, the system makes it possible for a user to re-order chronologically page content based on its viewing history. The user can know how old the page content is from his or her point of view. Consequently, a kind of temporal context is assigned to the current page content that can shed new light on content elements.

4 System Implementation

We have built a prototype browser in C#. In addition to having all the standard components of a traditional Web browser, it has a freshness mode and age display buttons. The system stores the contents of page versions visited by the user in the local cache. The dates of the page accesses are also recorded. When the freshness mode button is on, the system takes the URL of the current page and compares it with the previously viewed version of the page. We use the diff algorithm [5] to detect textual changes as it is a commonly used and easy to implement change computation algorithm. The detection of new images is done by comparing their *src* and *alt* attributes inside image tags. The fresh content on the current page is highlighted by a different background color that can be specified by a user, and the total freshness rate of the page and the date of its last access are shown in the bottom bar of the browser. The freshness mode is automatically switched off when the page is accessed for the first time.

In addition to comparing the content between the current and previously viewed versions, the system also fetches all the URL addresses of the links present on the current page and determines whether the linked pages have been previously accessed. No action is taken for those that have not been visited. For the previously accessed one, the system compares the content of its current version with that of the one recently visited. Depending on the amount of new content, a certain color is associated to the link based on the defined color scale, and the freshness degree of the page and the date of its last access are displayed in small font to the right of the link's anchor text. We have used the color scale that to some extent resembles the currently used link-color changing style in the Web. According to this scale, a page with completely new content for the user will have a link in the standard blue color, while a page with

completely known content ($V(t_{n-1})= V(t_n)$, where t_n is the present moment) will have a dark red color. Pages with other freshness degrees will have links displayed in colors that are between blue and red depending on their amounts of new content (see Figure 1). Freshness degree was calculated using Equation 1.

If one of the links is clicked, the browser loads the requested page, indicates its new content and displays the freshness degree of the page. If the user then returns to the page from where the link was followed, the system displays the page with the same markings as before. The only change is the update of the visited link markings. The system does not re-compute the freshness of the page as the user may not have finished viewing the fresh content and links from before and would likely have forgotten which ones they were. The user likely needs more time to view all of the fresh content on the page and those on the linked pages before the page freshness is recomputed. Thus, in the current implementation, only when the user switches off and on the freshness mode button is the page freshness recomputed and the fresh content marked. When the freshness mode button is off, the system works as a traditional browser. Figure 2 shows an example of an annotated page. The parts in yellow indicate the fresh content for the user.

When the age display button is pressed then the age computation and visualization process is triggered for the currently viewed Web page content. The system compares previous views of the page with the current view to find the earliest page snapshots containing content elements. The comparison sequence depends on whether sequential or binary searches are used. The former is used when the number of stored past page snapshots is below a pre-defined threshold; otherwise the latter is utilized. The system tries to group and embrace by visual frames the neighboring parts of the page content that have the same dates. Each frame has a date added in the bottom-right corner in a small font. This is done in order to minimize the amount of additional content introduced into the page so that the original layout and outlook of the page are changed as littler as possible. Figure 3 shows an example of an annotated page using the user-oriented age determination. Upon pressing the button again the system comes back to the original outlook of the page.

Fig. 2. Example of an annotated page with fresh content

Fig. 3. Example of an annotated page with user-oriented, subjective age of the content

5 Conclusion

Incorporation of the mechanism for the personalized freshness detection into a browser enables easy identification of content that has not yet been viewed by the user. This improves the browsing experience by making the user aware of content that is fresh from his or her viewpoint. It extends the already widely accepted mechanism of link color changing to using an array of colors to indicate the degree of content freshness of linked pages. This approach is proposed to facilitate browsing and navigation by decreasing the cost and time needed to find fresh content.

In addition, the system is equipped with the mechanism for the user-oriented detection of the subjective age of page content. It displays the information about dates when the user has seen certain content on the page for the first time and hence determines the age of the page content from the point of view of the user. This may help the user to better understand the current content on the accessed pages and the distribution of added changes in time.

Acknowledgements. This research was partially supported by the Japanese Ministry of Education, Culture, Science and Technology Grant-in-Aid for Scientific Research in Priority Areas entitled: Content Fusion and Seamless Search for Information Explosion (#18049041, Representative Katsumi Tanaka), and by the Informatics Research Center for Development of Knowledge Society Infrastructure (COE program by the Japanese Ministry of Education, Culture, Sports, Science and Technology) as well as by the Japanese Ministry of Education, Culture, Science and Technology Grant-in-Aid for Young Scientists B (#18700111).

References

1. Boyapati, V., Chevrier, K., Finkel, A., Glance, N., Pierce, T., Stockton, R., Whitmer, C.: ChangeDetector™: A Site Level Monitoring Tool for WWW. In Proceedings of the 11th International WWW Conference. Honolulu, Hawaii, USA (2002) 570-579
2. Brewington, B.E., Cybenko, G.: How Dynamic is the Web? In Proceedings of the 9th International World Wide Web Conference. Amsterdam, The Netherlands (2000) 257-276

3. Cho, J., Garcia-Molina, H.: The Evolution of the Web and Implications for an Incremental Crawler. In Proceedings of the 26th International Conference on Very Large Databases (VLDB). Cairo, Egypt (2000) 200-209
4. Cockburn, A., McKenzie, B.: What Do Web Users Do? An Empirical Analysis of Web Use. International Journal of Human-Computer Studies 54(6) (2001) 903-922
5. Diff Algorithm: http://www.codeproject.com/cs/algorithms/diffengine.asp
6. Douglis, F., Ball, T., Chen, Y., Koutsofios, E.: AT&T Internet Difference Engine: Tracking and Viewing Changes on the Web. World Wide Web Journal 1(1) (1998) 27-44
7. Douglis, F., Ball, T., Chen, Y.-F., Koutsofios E.: WebGUIDE: Querying and Navigating Changes in Web Repositories. In Proceedings of the 5th International World-Wide Web Conference on Computer Networks and ISDN Systems. Amsterdam, The Netherlands (1996) 1335-1344
8. Fetterly, D., Manasse, M., Najork, M., Wiener, J.L.: A Large-scale Study of the Evolution of Web Pages. In Proceedings of the 12th International World Wide Web Conference. Budapest, Hungary (2003) 669-678
9. Lieberman, H.: Letizia: An Agent That Assists Web Browsing. In Proceedings of the International Joint Conference on Artificial Intelligence. Montreal, Canada (1995) 924-929
10. Nielsen, J.: "Change the Color of Visited Links". Jakob Nielsen's Alertbox, (2004), http://www.useit.com/alertbox/20040503.html
11. Qiang, M., Miyazaki, S., Tanaka, K.: WebSCAN: Discovering and Notifying Important Changes of Web Sites. In Mayr, H.C., Lazansky, J., Quirchmayr, G., Vogel, P. (Eds.): Proceedings of the 12th International Conference on Database and Expert Systems Applications. Lecture Notes in Computer Science, Vol. 2113. Springer-Verlag, Berlin Heidelberg New York (2001) 587-598
12. Herder, E., Weinreich, H., Obendorf, H., Mayer, M.: Much to Know about History. In Proceedings of the Adaptive Hypermedia and Adaptive Web-based Systems Conference. Dublin, Ireland (2006)