Special Section on Data Engineering

Page History Explorer: Visualizing and Comparing Page Histories

Adam Jatowt[†], *non-member*, Yukiko Kawai[††] *and* Katsumi Tanaka[†], *members*

SUMMARY Due to the increased preservation efforts, large amounts of past Web data have been stored in Web archives and other archival repositories. Utilizing this data can offer certain benefits to users, for example, it can facilitate page understanding. In this paper, we propose a system for interactive exploration of page histories. We demonstrate an application called *Page History Explorer* (PHE) for summarizing and visualizing histories of Web pages. PHE portrays the overview of page evolution, characterizes its typical content over time and lets users observe page histories from different viewpoints. In addition, it enables flexible comparison of histories of different pages.

key words: *page history visualization, web archives, page evolution, change degree, comparative histories*

1. Introduction

During browsing users create mental images of pages based on encountered content. However, Web pages often undergo various changes over time. For example, the current content of a page may be quite different from its usual topics. In addition, current views of the page may not provide clear answers to the question whether the page is frequently updated or rather stale. Although, users can better understand pages and their temporal characteristics after frequently revisiting them, it is still difficult for first-time or rare visitors.

Clearly, providing additional knowledge about pages and their characteristics should be advantageous to users as it could improve various tasks on the Web such as page authoring, browsing or bookmarking. For example, page authors could better estimate the relevance and usefulness of pages to which they plan to make links or which they wish to bookmark for later use. However, as this knowledge is distributed over time and, thus, often hidden from the current view, one should revert to the analysis of page histories. Yet, browsing manually the content of past page versions requires much time and effort.

In this paper, we demonstrate Page History Explorer - an application for providing aggregated overviews of page histories and their long-term characteristics. In Figure 1, we show the visual summary generated by PHE for an example page.

The proposed system works as follows. First, samples of past page content are mapped on 2D space that represents time and change degree. This allows for roughly portraying various aspects of page evolution such as changes in outlook and for indicating time periods when large content changes occurred. Second, PHE summarizes historical content of pages by detecting prevailing and active content occurring over time. This kind of a temporal summary is visualized as clouds of salient terms that characterize historical page content. Users are also allowed to input arbitrary keywords in order to view page histories from keyword-related viewpoints. It is then possible to observe the periods when the page was updated with related content and find other terms that were frequently co-occurring with the specified keyword. Finally, Page History Explorer provides also a history comparison function. Users can contrast the characteristics of selected pages, compare different parts of the history of the same page or find the common aspects in a group of different pages.

In this work, we assume a realistic situation, in which past data on pages is stored by third party repositories, such as Web archives. As these repositories provide only fragmentary data, we have to cope with incomplete historical evidence. PHE downloads sample data from Web archives and uses it for reconstructing page histories in accurate way.

Analyzing page histories can be useful for various reasons. We discuss them below.

Characterizing and comparing pages. The overview of typical content and characteristics of pages can serve as a kind of temporal context that allows better understanding the current page state and its current content by shedding new light on it. For example, one can investigate, whether the current page topics were often discussed on the page in the past or whether they are rather novel. In addition, comparing histories of related pages makes it possible to determine their similarities and differences. Also, as pages often represent certain real-world objects (e.g., organizations, companies, people), hence, investigating their past could provide additional insights into the historical analysis of these objects.

Manuscript received June 04, 2010.
Manuscript revised March xx, 20xx.

†The author is with Kyoto University, Yoshida-Honmachi, Sakyo-ku, 606-8501 Kyoto, Japan.
††The author is with Kyoto Sangyo University, Motoyama, Kamigamo, Kita-Ku, 603-8555 Kyoto, Japan

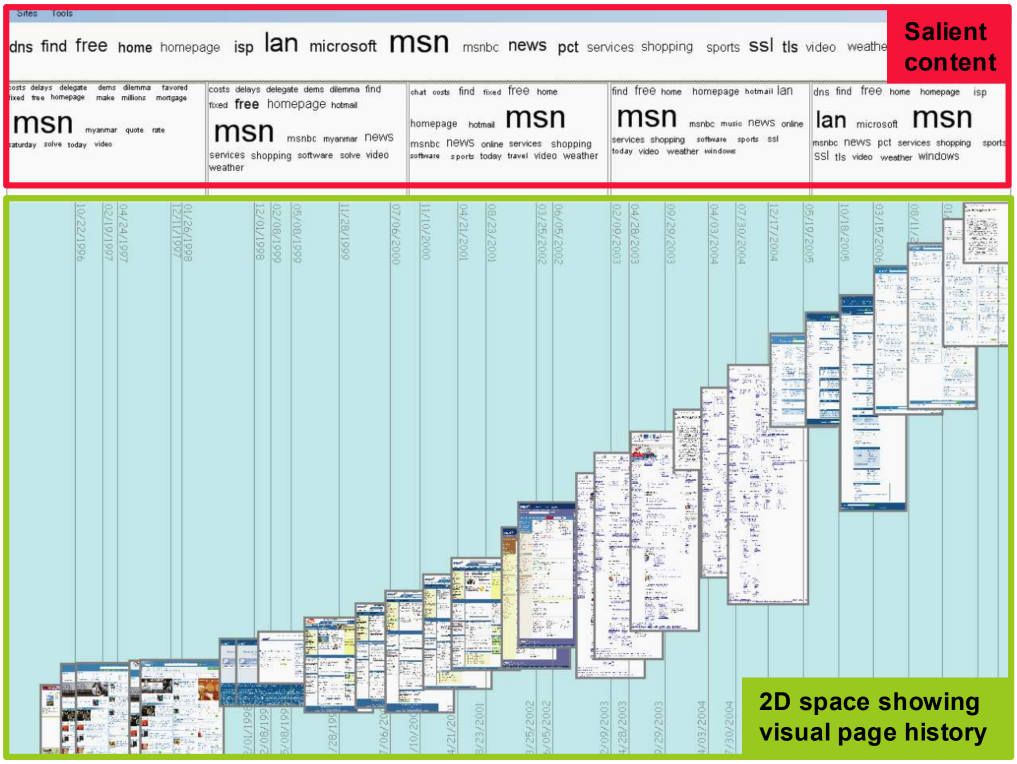

Sites Tools
dns find free home homepage isp lan microsoft msn msnbc news pct services shopping sports ssl tls video weathe
Salient content
costs delays delegate dems dilemma favored fixed free homepage make millions mortgage msn myanmar quote rate saturday solve today video
costs delays delegate dems dilemma find fixed free homepage hotmail msn msnbc myanmar news services shopping software solve video weather
chat costs find fixed free home homepage hotmail msn msnbc news online services shopping software sports today travel video weather
find free home homepage hotmail lan msn msnbc music news online services shopping software sports ssl today video weather windows
dns find free home homepage isp lan microsoft msn msnbc news pct services shopping sports ssl tls video weather windows
10/22/1996
02/19/1997
04/24/1997
12/11/1997
01/26/1998
12/01/1998
02/08/1999
05/08/1999
11/28/1999
07/06/2000
11/10/2000
04/21/2001
08/23/2001
03/25/2002
06/05/2002
02/09/2003
04/28/2003
09/29/2003
04/03/2004
07/30/2004
12/17/2004
05/19/2005
10/18/2005
03/15/2006
2D space showing visual page history

Figure 1 Example of a summary of MSN homepage history (R=5, zoom=0.12, activity term clouds, mm/dd/yyyy format).

Estimating usefulness and quality of pages. Temporal context of pages can be also helpful for evaluating their usefulness and trustworthiness. This is because summarizing pages' histories can answer questions about their long-term relevance and average freshness. It is thus possible to check whether the topical scopes of pages were consistent over longer time periods and whether the page is, in general, devoted to topics that are interesting to users. In addition, one can roughly judge whether the page is usually frequently updated. Freshness is an important quality measure as there are many stale and abandoned pages in the Web [2].

Exploring page histories. Users may want to explore past content of pages for a variety of purposes[1]; for example, to look at the histories of favorite pages that they frequently revisited. Such a "journey to the past" may offer sentimental and informative values. Recently, multiple use cases of Web archives have been identified by the International Internet Preservation Consortium [7] and some researchers [13,14]. For example, in one possible case, a local journalist wishes to quickly overview the content of a municipality homepage over a selected time period. In general, aggregated views on page histories should be useful in many potential use cases of Web archives, considering that temporal search is still unavailable in Web archives, and, that one has to analyze large amounts of historical data[2].

The remainder of this paper is as follows. In the next two sections we provide the background and survey the related work. Section 4 discuses data accumulation and processing, while Section 5 explains the methodology behind detecting salient terms from page histories. The next section describes the way in which PHE visualizes page histories and their comparisons. Section 7 explains the system implementation, demonstrates some experimental results and provides the discussion of related issues. Lastly, in Section 8, we conclude the paper and outline our future plans.

2. Background

In the early hypertext systems version management was often used for enabling source tracking or for preventing content duplication between document versions. On the other hand, Web pages, with few exceptions (e.g., wikis),

[1] The scale of interest in page histories is portrayed by the usage statistics of the Internet Archive's Web collection, which is currently the largest repository of page historical data. Its homepage (http://web.archive.org) had about 10 million hits or ~4 million retrieval requests (URL + exact date) per day according to data on June 2007 [15].

[2] For example, the Internet Archive's Web collection contains more than 6,000 past copies of Yahoo! homepage (http://www.yahoo.com) accumulated since 1996.

are generally not subject to any automated versioning. In most cases, external, third-party data sources such as Web archives have to be exploited in order to collect any evidences on past content of pages.

The Internet Archive is the most popular Web archive containing more than 95 billion of archived Web documents (over 2 Petabytes of data) crawled since 1996 [15]. For a rough comparison, the size of the indexable Web was estimated to be about 11.5 billion pages according to the research done in 2005 [5]. Of course, we have to account for the fact that the Web has grown considerably since that time and that the Internet Archive contains multiple copies of the same pages. Nevertheless, the amount of data stored only in the Internet Archive is large enough for providing a reasonable Web-scale service.

The current interfaces to Web archives usually have rather limited interaction and content processing capabilities, and they do not provide any aggregate knowledge about past page content. For example, Wayback Machine - the "gateway" to the Internet Archive's collection provides a directory page listing links to time-stamped snapshots of pages. Exploring the history of a page means thus accessing its snapshots one by one. This is troublesome, especially, for pages, for which many snapshots were preserved. In addition, Web archives often do not provide textual search capabilities as content indexing and ranking inside temporal collections are far from trivial.

Lastly, we have to note that the freedom of access to the archived content varies from country to country, as it depends on content copyrights and legal deposit legislations. In this paper, however, we neglect any legal issues assuming unrestricted access for all interested users.

3. Related Work

Despite nearly two decades of the Web history, still rather limited solutions have been offered to users for interacting with the past Web.

Visual Knowledge Builder (VBR) [18] was one of the first proposals for navigating histories of hypertexts. The objective was to allow readers viewing the way in which hypertext was constructed, the author's writing style and, in general, the context of created content. VBR was designed for private hypertexts for which all the previous versions were locally stored.

Zoetrope [1] is an interactive interface for manipulating and overlaying the evolution of DOM elements over time. The system allows users to visualize changes in selected content elements through the metaphor of lenses. Teevan at al. [19] implemented *DiffIE*, a browser plug-in that indicates content changes on revisited pages. To support effective user revisiting, annotation of revisited links with their change degree have been proposed in [11]. Differently to the above works, PHE visualizes the macro-history of an entire page and computes several kinds of content summaries.

In our previous work we demonstrated *Past Web Browser* [8] - a browsing application for Web archives that presents consecutive page snapshots as a slideshow to visualize content transitions. The browser indicates changes between the snapshots using appearing or disappearing animation effects depending on a change type. Past Web Browser was designed for enabling the detailed exploration of page histories and for enhancing the navigation of the past Web. In contrast, the objective of PHE is providing summarized overviews of page histories or their parts.

In another work [10], we have proposed using Web archive data for determining the age of content on pages. For an arbitrary page, the system approximately estimates the creation dates of the content that a user encounters on a page. This is done by detecting the oldest page snapshots that contain the same content elements as the current page version. This system and the Past Web Browser can be used together with PHE in a complementary way in order to provide a unified framework for historical exploration of pages and their relation to the current states of Web pages.

Viégas et al. [21] proposed the *History Flow* system for portraying the evolution of Wikipedia pages and for studying the authors' cooperation. It indicates the contributions of individual editors to a given article and their persistence over time. There are several important differences between their system and ours mainly due to different objectives and data used. First, PHE aggregates historical page content indicating dominant topics and common changes over time. In contrast, the History Flow system does not produce any temporal summary of past content focusing rather on visualizing contributions of different authors and their role in the process of page development. Second, there is no issue of incomplete data for Wikipedia pages as all their past versions are preserved and, in addition, these pages are quite simple and well-structured. Third, PHE provides several novel interaction mechanisms such as keyword-based history overview or visualization based on different change types. Lastly, our system enables also the detection of common and different aspects between the histories of multiple pages.

Dubinko et al. [4] demonstrated an application for efficiently visualizing interesting tags of images stored in an image sharing online community over time. Due to the large size of the dataset the authors focused mainly on providing efficient solutions for indexing and pre-computing the data. Interesting tags were found by detecting bursts of tag occurrences for every single day of the year and shown as animation using the river and waterfall metaphors. PHE differs from this work in the character of data that is used and in the functionalities it provides. For example, the appearance of page content has certain duration over time, while assignment of tags to images is an instantaneous process.

Some research was done for effectively visualizing the evolution of link structures on the Web [3,20]. Chi et al. [3] investigated ways for showing the changes in the usage and structure of Web sites based on site usage logs. Toyoda and Kitsuregawa [20] observed the evolution of Web communities over time on the example of Japanese archive. Like in the case of the above discussed works, the objectives of these researches and the data used are different from ours. While they display the changes in the structure and popularity of Web objects, our focus is on detecting and comparing dominant topics and characteristics of page content over time.

McCown et al. [14] measured the availability of page copies inside the repositories of major search engines and the Internet Archive. Their research was motivated by the need to provide efficient methods for reproducing the latest versions of Web sites. The objective was to help Web authors with retrieving Web data in case of its sudden loss, for example, due to server crashes.

Some researchers attempted to aid users with the comparison of current page structures [17,12]. Nadamoto and Tanaka [17] developed a comparative browser. When a user visits any page using their system then another similar page is automatically retrieved from the Web for providing complementary content. Similar paragraphs are also marked in both pages. Liu et al. [12] proposed a method for visualizing differences and similarities between Web sites of two competing companies. The idea was to combine pages from two sites as a single pool of data for hierarchical clustering. Cluster memberships revealed then the types of pages that are more common in one site compared to the other site and indicated the kind of pages that are similar for both Web sites. Although the above approaches are useful in comparing different Web objects, nevertheless there has been no tool proposed so far for comparing the histories of Web pages.

4. Data Preparation

In this section we overview the data preparation process. First, we provide several definitions.

Def. 1. Page lifetime - time period when a page was accessible on the Web.

Def. 2. Page snapshot - a copy of content that a page had at a time point t_i during its lifetime. t_i is a timestamp of the snapshot.

The term page snapshot should be distinguished from page version, as the latter implies the occurrence of changes. Two page snapshots with timestamps t_i and t_{i+1} may be thus exactly same.

Def. 3. Web archive - a collection containing past snapshots of selected Web pages.

Def. 4. Page history reconstruction - the process of reproducing page history using data obtained from Web archives.

PHE offers two modes of data accumulation, offline and online. In the offline mode, data is stored locally, for example, it can be past copies of pages frequently visited by a user. Since, in this case, the cost of data collection is negligible; hence, all page snapshots can be used for the history reconstruction. On the other hand, in the online mode, data is downloaded from archival repositories that provide online interfaces. The cost depends then on the amount of fetched data. For a selected URL and a time period *T*, PHE downloads *N* past snapshots[3] from the collection of Web archives. The snapshots should be possibly uniformly distributed over time. In order to do so, first, PHE fetches the temporal metadata of past snapshots from Web archives. The time period *T* is then divided into *N-1* time segments of equal lengths. Next, the system downloads page snapshots that have timestamps closest in time to the boundaries of the segments. Note, that the fetched snapshots may not always be evenly distributed, especially when available data is scarce.

4.1 Change Detection

As pages, normally, cannot be crawled in a continual way only fragmentary data is available in Web archives, and, thus, the history reconstruction process will unavoidably contain some errors. Intuitively, there is a trade-off between the amount of data and the accuracy of the history reconstruction, and indirectly, the precision of the generated summaries.

Suppose there is a sequence of seven snapshots, $s_0,...,s_6$ of a given page with their corresponding timestamps, $t_0<...<t_6$ (see the top timeline in Fig. 2). Suppose also that the snapshots contain partially different content except for the snapshots s_2, s_3 and s_4 which are same. In order to reconstruct the complete history of the page we could simply “interpolate” the content of all the snapshots as shown in the middle timeline. This means assuming the persistence of content of a known page snapshot over half of the distance to the subsequent page snapshot. Obviously, such naïve method does not consider the parts of content that occurred in the page in the past, yet were not recorded in any available snapshot. This could be, for example, a short text that appeared on the page briefly enough to remain undetected by the crawler. Let us call it *transient content*. The time span during which there was any transient content will be called *transient period*. The bottom timeline in Fig. 2 displays the actual page history with white areas indicating the transient periods. In general, not only the transient content is unknown, but also its timing is uncertain. There can be any number of changes occurring inside the transient period and their boundary time points are usually unknown. Intuitively, for any pair of

[3] By default, *N* is set to *30*. It can be also set by users.

snapshots, the uncertainty related to the transient content depends on the length of the time distance between the snapshots, the amount of the content difference obtained from their comparison and the average change frequency of the page. Note that, although the transient content is unknown during transient periods, the remaining (static) content that was on the page at that time still can be found by comparing neighboring page snapshots.

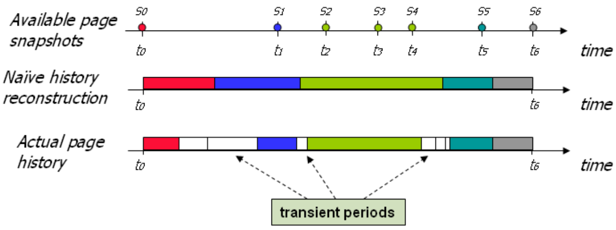


Figure 1 The result of a naïve history reconstruction compared with the actual page history.

In order to approximately reconstruct page histories we will need to estimate the probability of transient content. For this purpose we first detect changes in page content over time by comparing consecutive page snapshots. We use here an efficient difference computation algorithm proposed in [16]. Added content of snapshot s_i is defined as the content part that occurs in s_i but that does not occur in s_{i-1}. Analogically, the deleted content of s_i is defined as the part of the content of s_i that does not occur in s_{i+1}. In this way, we find content that was added to or deleted from each known past snapshot.

We define also static and changed content for the time spans between snapshot timestamps. In other words, we associate content changes with the periods of page history rather than with the timestamps of individual snapshots as it was done above. Static content for a time period $[t_i,t_{i+1}]$ is defined as the content that occurs in snapshots s_i and s_{i+1}. On the other hand, the changed content for this period is the union of deleted content of snapshot s_i and the added content of snapshot s_{i+1}. The changed and static content for the time spans between any pairs of consecutive snapshots will be later used for estimating the probability and the amount of transient content. In Figure 3, we demonstrate the change estimation on the example of four snapshots.

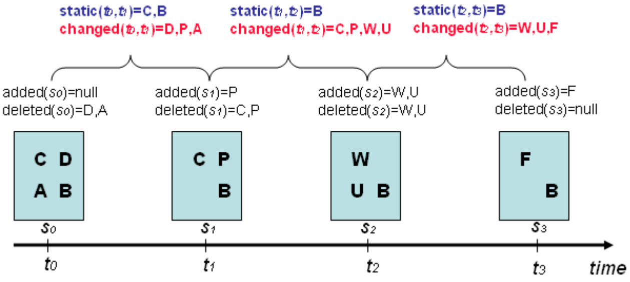


Figure 3 Change detection on the example of four snapshots.

5. Summarizing Past Content

PHE uses term clouds, which are similar to tag clouds, in order to visualize past content of pages. Tag clouds have become a common method for representing popular content in many Web 2.0 applications where the size of tag fonts indicates the levels of content popularity. In the case of a single document, term clouds can be used to provide a quick overview of document's content. We extend the usage of term clouds as a means for providing aggregate information on past content of pages.

We propose two types of temporal term clouds, *prevalence* and *activity term clouds*. The former is used to represent common content appearing in a page over time. The latter indicates the content that frequently appeared inside the changing (active) parts of the page. Both term clouds can be also adapted to user-selected keywords in order to capture most frequently co-occurring terms. In addition, we also propose a series of short-term clouds that enable viewing the distribution of salient terms inside shorter time segments. The following subsections describe the way to construct all the term clouds.

5.1 Prevalence Term Clouds

Prevalence term cloud is constructed based on the prevalence scores of terms which reflect how commonly terms occurred over time. The prevalence score of a term is estimated as the weighted average of the term's frequency function over time.

$$S^{pr}(a;T)=\frac{1}{T}\sum_{i=0}^{N-1}(t_{i+1}-t_i)*TF_a([t_i,t_{i+1}]) \qquad (1)$$

$TF_a([t_i,t_{i+1}])$ is the estimated frequency of a given term a in the time period $[t_i,t_{i+1}]$ where t_i and t_{i+1} are the timestamps of the pair of consecutive page snapshots. For presentation clarity, the explanation of modeling frequency function of terms over time is deferred to the Appendix.

According to Equation 1, the longer are the time frames during which a given term had a relatively high occurrence frequency in the page, the higher is its prevalence score. Thus, if the term occurred frequently in the page over lengthy time frames, it is deemed to be prevailing over time.

The above calculation does not account for additional factors affecting terms' visibility inside snapshots. That is, one could assign additional weights to the occurrences of terms depending on such factors as term locations, font sizes or other visual features.

5.2 Activity Term Clouds

Prevalence scores do not distinguish between static and changing content in pages as they reflect only the general frequency of the occurrence of terms in page histories. However, a term (e.g., "copyright") that appears always in the static content parts of the page may not be of much interest to users. On the other hand, terms that are often added and deleted may indicate significant content and may be interesting for users. We thus calculate the activity scores of terms in order to construct the activity term cloud.

The activity score of a given term *a* is computed as the probability that the term occurred in added or deleted content parts in the past. This likelihood is estimated as the combination of the probability of a change occurrence in *T* and the probability of the term *a* appearing in this change.

$$S^{ac}(a;T)=\frac{M}{N}*TF_a^c(T) \qquad (2)$$

M is the number of snapshots in which we observe any content changes and $TF_a^c(T)$ is the average frequency of term *a* inside the changed parts of all snapshots (see Equation 8 in Appendix).

5.3 Temporal Term Co-occurrence

In some cases, a user may also want to learn how a page discussed certain topics over time. To accommodate this need, we propose the calculation of *temporal term co-occurrences* in page histories.

We adapt here a standard Jaccard coefficient measure that captures non-casual association between two terms in a document or a collection of documents. We convert this space-centric co-occurrence measure into a time-centric one by treating page snapshots as unit segments.

$$Coocc(a,b;T)=\frac{S^{pr}(a,b;T)}{S^{pr}(a;T)+S^{pr}(b;T)-S^{pr}(a,b;T)} \qquad (3)$$

$S^{pr}(a,b;T)$ is a prevalence score of the common occurrence of terms *a* and *b* in *T* (Equation 4). It can be considered as the probability of seeing both terms in a randomly extracted snapshot of the page within *T*.

$$S^{pr}(a,b;T)=\frac{1}{T}\sum_{i=0}^{N-1}(t_{i+1}-t_i)*TF_{a,b}([t_i,t_{i+1}]) \qquad (4)$$

$TF_{a,b}([t_i,t_{i+1}])$ is the term frequency function of the common occurrence of both terms *a* and *b* during $[t_i,t_{i+1}]$ (see Appendix for details).

Temporal term co-occurrence can be calculated analogically using the activity scores of terms upon user's request. This allows for detecting most frequently co-occurring terms inside the active content of pages. Although, we have demonstrated here the case of a single keyword, the calculation can be also expanded to accommodate cases of multiple keywords.

5.4 Unit Term Clouds

In addition, PHE computes term clouds for a series of unit time segments within *T* in order to enable finer granularity analysis of past content. We call such term clouds *unit term clouds*.

The calculation of term scores in somewhat similar to *tf*idf* weighting scheme which is adapted here to time series data. First, page history is divided into a series of equal time segments. Scores are then computed for all terms appearing in each time segment. In this way, we effectively treat a unit segment of page history as a single "virtual" document. The sequence of such segments corresponds then to the collection of documents. In general, we score terms in each unit segment according to the following rule. Salient terms in a given, target time segment T_w are terms that have high scores inside T_w and, at the same time, have low scores inside other time periods. The second condition is implemented by calculating the rate of the term score in T_w and the corresponding scores in other time segments (Equation 5).

$$S_{unit}^{X}(a;T_w)=X(a;T_w)*\log\left(\sum_{j=1}^{R}\left[\frac{X(a;T_w)}{X(a;T_j)+1}\right]+1\right) \qquad (5)$$

Depending on user's choice, $X(a;T_w)$ denotes either prevalence or activity term scores. *R* is a user-specified number of unit time segments. T_j is any unit time segment within *T* ($1\leq j\leq R$).

6. Visualizing Page Histories

6.1 Visualizing History of a Single Page

Snapshots are converted to images and overlaid on 2D space where the horizontal axis represents time distance and the vertical one indicates the cumulative degree of added change (Figure 4). Added change degree is calculated as the number of terms inside content that was added to past page snapshots. The distance of snapshot s_{i+1} from snapshot s_i is thus the combined distance in temporal and change-degree dimensions. Note that the vertical axis shows the cumulative change of the page since the beginning of the analysis period.

With this kind of visualization, a user can roughly grasp the outlook of a page in the past thanks to comparing thumbnails of snapshots from different time points. Also, since the relative sizes of pages are retained, users can compare page sizes over time. More interesting, however, is the capability to portray the frequency and the amount of changes occurring over time. For example, a relatively long vertical distance between two consecutive snapshots accompanied by a rather short horizontal distance implies a sudden burst of new content added to the page. By looking at a change distribution over time users can spot the time periods during which large additions of new content occurred.

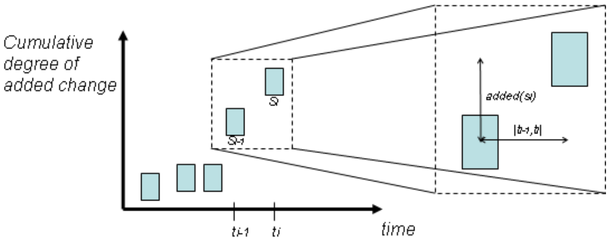


Figure 4 Visualization of page snapshots on 2D space.

Although the added change is the main focus here, a user can also visualize page history from the viewpoint of deleted changes or the viewpoint of combined added/deleted changes made to page. When the deletion option is chosen, the vertical axis will indicate cumulative degree of deletion type changes in order to demonstrate the way in which content was erased from the page over time. On the other hand, the combined view of added/deleted changes shows the general pattern of content movement making it possible to quickly judge whether, on average, the page was frequently updated or rather stale. Note, however, that although, rather unlikely, content reverts could happen, such as subsequent additions and deletions of the same content. Nevertheless, in general, the lack of any content changes in longer time frames translates to the higher probability of content obsoleteness or even page abandonment.

In order to characterize topics of the past content Page History Explorer displays top 20 terms calculated according to their prevalence or activity scores (see Figure 1). The font sizes of the terms reflect their relative scores calculated in Section 5. In addition, the sequence of unit term clouds is visualized directly over a 2D space. Each such unit term cloud contains top 20 terms calculated according to Equation 5. Unit term clouds indicate the top terms over shorter time frames and can be contrasted with the main term cloud computed over the entire time frame. Users can change the number of unit term clouds in real time by adjusting parameter *R* (*R* is set to *N/4* by default).

Upon inputting any keyword, a user obtains the overview of page history from the keyword-based viewpoint. Any arbitrary terms can be chosen here, for example, they may be selected from current page content or from term clouds returned by PHE.

In the keyword-based history overview the system displays prevalence or activity based temporal term co-occurrences of provided keywords. At the same time, the vertical axis in 2D space represents the frequency of the keywords inside content additions or deletions depending on the user's choice. Thus two consecutive snapshots will be located farther away from each other in the vertical dimension, the higher is the keywords' frequency inside the content change between these snapshots. Users can thus see the pattern of changes that contain given keywords over time.

Figure 5 shows the history of Yahoo! homepage summarized from the viewpoint of "iraq" keyword. One can notice that the new content related to "iraq" appeared on the page around 22nd January, 2003, several weeks before the start of the Iraq war. The last time there was any change related to this topic was around May 2006.

6.2 Comparing Histories of Multiple Pages

In this section we discuss the adaptation of PHE for detecting differences and similarities between histories of different pages. Naturally, pages chosen for the comparison should be related to each other in some way in order to provide meaningful results. For example, they may represent objects belonging to a certain real-world structure or a group, such as computer laboratories in Japan or competing companies. Note that more than two pages can be compared at once as shown in Figure 9.

Comparison of Page Outlook. PHE displays together snapshots of the selected pages on a 2D space. However, in order to make the page histories comparable some normalization has to be used. Compared pages are shown in relation to the page that had largest changes within *T*. The system calculates the cumulative change degree for all the compared pages within the selected time range. The change degree embraces added, deleted or both content change types depending on the user's selection. Let us suppose that C_i denotes the cumulative change degree of page *i* and that there are *n* compared pages. Then, the maximum value of the cumulated change degree among all the compared pages is used as a normalization factor ($C_{max}=max_{1<i<n}(C_i)$). For a page *i* the relative position of its last snapshot on the axis Y is thus equal to C_i/C_{max} and its remaining snapshots are arranged on the scale from 0 to C_i/C_{max} according to their change pattern. The vertical scale of the entire 2D space has thus a range from 0 to C_{max} making the page with the largest change degree start in the bottom left-hand side corner and end in the top right-hand side corner of 2D space.

With this kind of history presentation the users can contrast certain aspects of page histories. For example, it is possible to determine which page changed more, which page had, on average, larger size or which page had more fresh content containing certain keywords over time.

In Figure 6 we display the historical views of two pages, Open Directory Project[4] (ODP) and Yahoo! Directory[5]. Both pages are Web directory homepages. When visualizing histories of both pages in 2D space, we can see that the Yahoo! Directory page is nearly twice as

[4] http://www.dmoz.org

[5] http://directory.yahoo.com

young as the ODP homepage, it has much higher change degree and it usually contains larger and more diverse content. The Yahoo! Directory homepage displayed rich, frequently updated content in the past, rather than only showing top-level directory hierarchies as in the case of the ODP homepage. ODP is mostly a static page, with minor content changes. In fact, changes in the ODP homepage were often due to the updates in the count of visitors. The above characteristics of the Open Directory Project's homepage are consistent with the results that we have obtained using our age detection tool [10] mentioned in Section 3. We have found that June 2000 was the average creation date of the whole content. We could also determine that the estimated creation date of the oldest element on the page was some time before January 25th 1999 (this is actually the timestamp of the oldest available snapshot).

Comparison of Term Clouds. In order to compare temporal term clouds, users can activate a comparison window (Figures 6 and 8). They should select pages to be compared and decide a comparison viewpoint by choosing between prevalence, activity, or keyword-based prevalence/activity scores of terms. Then, for each compared page, its corresponding term cloud is displayed in the frame designated for the page. However, the simple contrasting of temporal terms clouds is not effective for detecting common terms among the pages. In order to enable more efficient comparison, PHE detects terms that have relatively high scores in all the compared pages and highlights them in the comparison window. That is to say, it looks for similar topics among histories of different pages. To find common terms among all the compared pages, PHE calculates their average scores over all the compared pages called *commonness scores*. Note that this calculation is done for all the terms of compared pages, hence, not only for the top terms of each page. This is because a term may not have a high score in a particular page, yet it can have a high commonness score within the set of compared pages.

For each compared page, PHE inserts up to 20 top terms according to their commonness scores into the frame designated for the page. These terms are marked by red color. Since already a term cloud of 20 top terms have been placed in the frame, thus, in total, there could be a maximum of 40 terms displayed for each page. However, the number of displayed terms is often lower than this, as duplicate terms are eliminated.

For a more efficient visualization, the system also re-arranges terms so that common terms (i.e. terms marked with red color) are shown before the other terms inside each frame. Thus, users can see two groups of terms for each page, the terms that are popular in the history of all compared pages and those that are common for the particular page. Terms in both groups are also ordered according to their font sizes. Users can thus spot terms that indicate the most similar content for the compared pages and, at the same time, they can easily see how high scores these terms have for each particular page.

Figure 6 shows the term comparison for the example of ODP homepage vs. Yahoo! Directory homepage. When contrasting the top prevailing terms in both pages one can notice several terms representing same categories, such as, “game”, “science” and “art” that frequently occurred on both pages during their lifetimes. We discuss other comparative examples in the Section 7.2.

Besides the comparison of terms that are representative for histories of different pages, users can also contrast top terms for different time periods or different content types of the same page. For example, we can compare top prevailing vs. top active terms in the same page, top co-occurring terms for keyword *a* vs. top co-occurring terms for keyword *b* or top terms in the first vs. top terms in the second half of page's lifetime.

7. Experiments

7.1 System Implementation

The system has been implemented in C# using the Microsoft .Net Framework. We describe its work mode below.

When a user inputs page URL, time period and the number of snapshots, PHE downloads data from the Internet Archive and stores it in a local cache. Data could be also concurrently downloaded from other repositories [8,14]. If there is error page encountered the system tries to re-download the snapshot for which error messages were returned instead. If it fails again the snapshot that is nearest in time is retrieved. After data download, PHE converts page snapshots to images. The images are then arranged on 2D space according to the positions of their top-left corners starting from the bottom left-hand side with a certain offset left in order to make space for showing the first and last snapshots on the screen. Each displayed image has a vertical line attached with its timestamp in a selected date format displayed at both the upper and the lower ends of the line.

The users can click on any snapshot in order to activate the descriptor window for the page (see Figures 5 and 9). Users can then change any parameters related to the page in the descriptor window, such as, time period, number of snapshots, change type used in 2D space visualization, zoom level, transparency level, types of displayed terms, etc. Clicking on any snapshot also highlights all the other snapshots of the same page and brings them up. This is useful in case when histories of several pages are visualized together. In such a case, the snapshots belonging to the same page are also framed by the same color in order to better distinguish them from the snapshots of other pages.

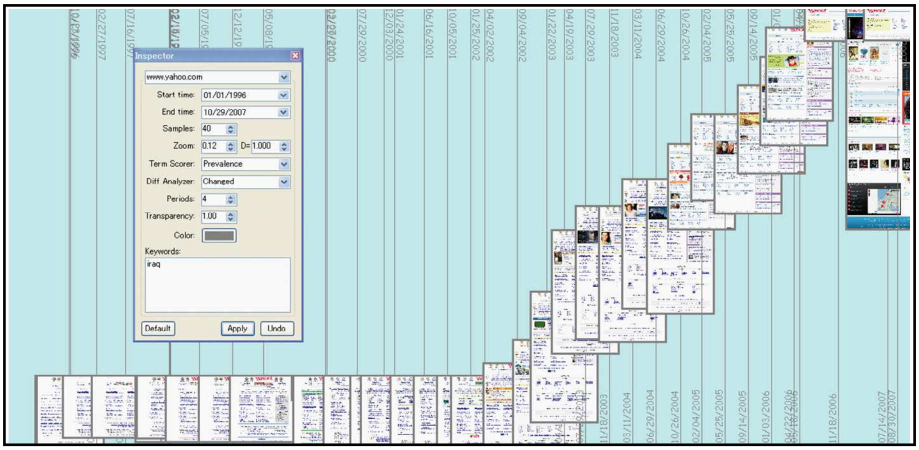
When zooming in the users should see the past content of snapshots in more detail. However, in the case of large *N*, the snapshots can be covered by the neighboring ones and thus become only partially visible. Of course, zooming out will prevent this, however, then, the content cannot be seen in detail. Thus, if necessary, the users can double click on any snapshot to see its content in a separate window and, optionally, to follow its links. In addition, transparency level of snapshots can be adjusted on the scale from *0* to *1*.

In order to calculate term clouds, PHE extracts textual content of page snapshots and discards stop words. The list of stop words, besides containing common terms, includes also terms specific to the Web, such as "click" or "link". We have also implemented stemming function for a more efficient calculation of term frequencies; however, stemmed terms create certain difficulty for users in understanding results.

7.2 Demonstration

Figure 1 shows the results for MSN homepage through its all recorded history (we set 01/01/1996 as the start date and 29/10/2007 as the end date in all the examples in this paper). The displayed tag cloud contains top active terms for the page for the whole history and for 5 unit time segments.

Figure 6 shows the comparison between ODP and Yahoo! Directory homepages which was discussed in Section 6.2.



Inspector

www.yahoo.com

Start time: 01/01/1996

End time: 10/29/2007

Samples: 40

Zoom: 0.12 D= 1.000

Term Scorer: Prevalence

Diff Analyzer: Changed

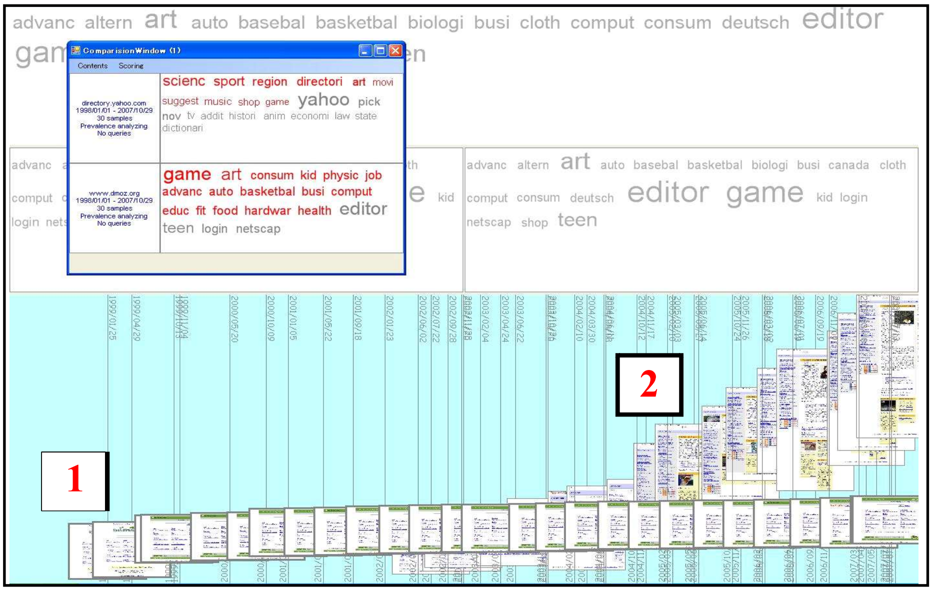
Periods: 4

Transparency: 1.00

Color:

Keywords:
iraq

Figure 5 Yahoo! homepage history from the viewpoint of "iraq" term (R=4, zoom=0.12, prevalence term cloud, mm/dd/yyyy format, descriptor window shown).



ComparisionWindow (1)

directory.yahoo.com 1998/01/01 - 2007/10/29 30 samples Prevalence analyzing No queries	scienc sport region directori art movi suggest music shop game yahoo pick nov tv addit histori anim economi law state dictionari
www.dmoz.org 1998/01/01 - 2007/10/29 30 samples Prevalence analyzing No queries	game art consum kid physic job advanc auto basketbal busi comput educ fit food hardwar health editor teen login netscap

advanc altern art auto basebal basketbal biologi busi cloth comput consum deutsch editor
advanc altern art auto basebal basketbal biologi busi canada cloth comput consum deutsch editor game kid login netscap shop teen
1
2

Figure 6 Comparison of the ODP (1) and Yahoo! Directory (2) homepages (ODP page highlighted, R=2, zoom=0.1, prevalence term clouds, stemming, comparison window shown in the top-left hand-side, yyyy/mm/dd format).

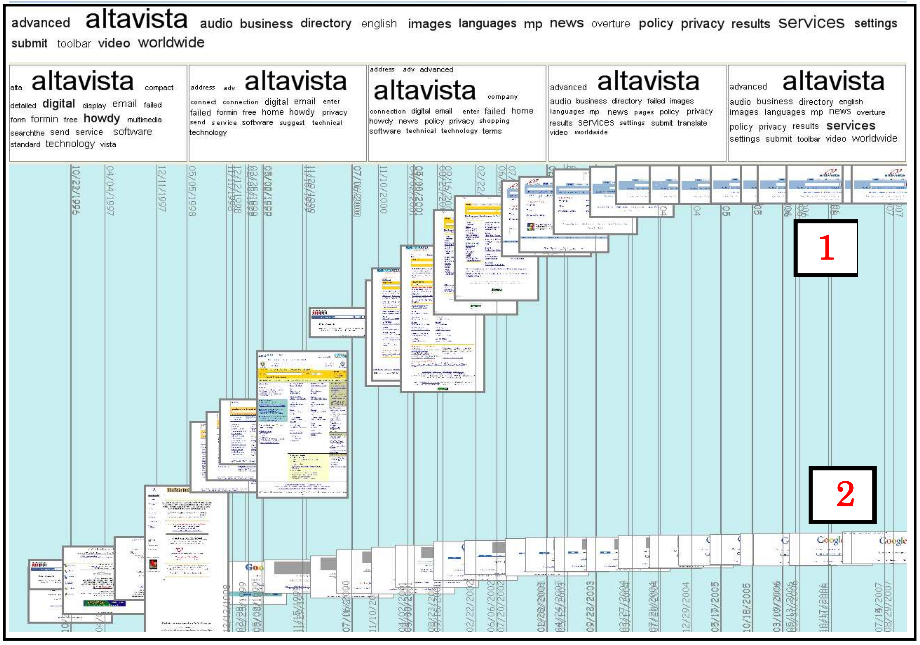
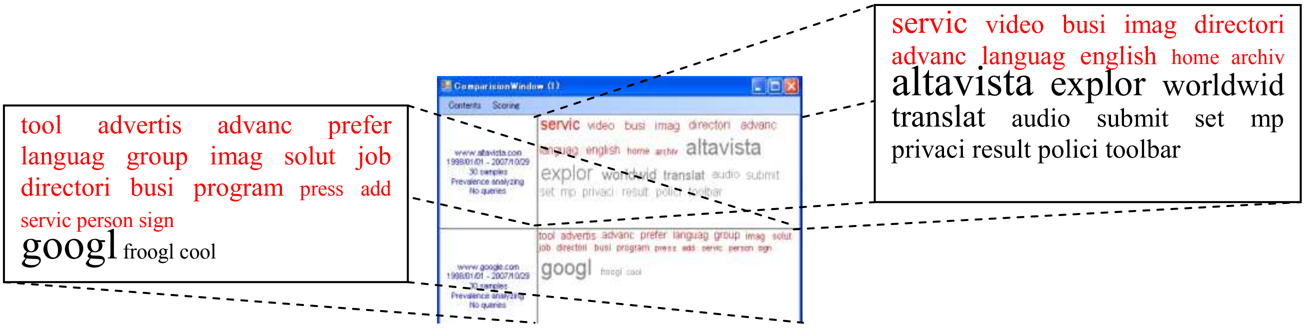

advanced altavista audio business directory english images languages mp news overture policy privacy results services settings submit toolbar video worldwide
alta altavista compact detailed digital display email failed form formin free howdy multimedia searchthe send service software standard technology vista
address adv altavista connect connection digital email enter failed formin free home howdy privacy send service software suggest technical technology
address adv advanced altavista company connection digital email enter failed home howdy news policy privacy shopping software technical technology terms
advanced altavista audio business directory failed images languages mp news pages policy privacy results services settings submit translate video worldwide
advanced altavista audio business directory english images languages mp news overture policy privacy results services settings submit toolbar video worldwide
1
2

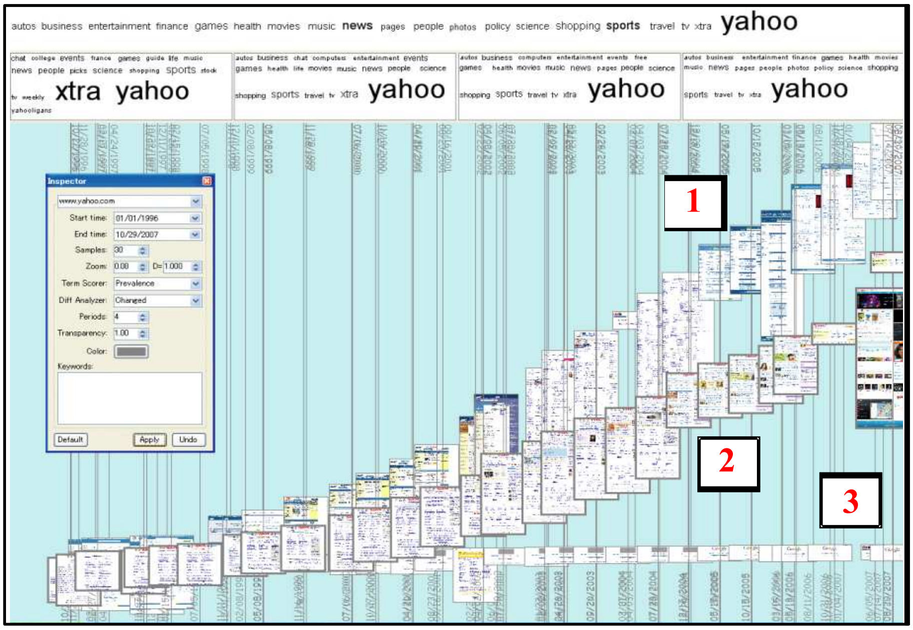
Figure 7 Comparison of the Altavista (1) and Google (2) homepages (AltaVista page highlighted, R=5, zoom=0.1, prevalence term clouds, mm/dd/yyyy format).


ComparisonWindow
Contents Scoring
servic video busi imag directori advanc languag english home archiv altavista explor worldwid translat audio submit set mp privaci result polici toolbar
tool advertis advanc prefer languag group imag solut job directori busi program press add servic person sign googl froogl cool

www.altavista.com
1998/01/01 - 2007/10/29
30 samples
Prevalence analyzing
No queries

www.google.com
1998/01/01 - 2007/10/29
30 samples
Prevalence analyzing
No queries

Figure 8 Comparison window for the prevalence scores of the AltaVista and Google homepages (stemming used).


autos business entertainment finance games health movies music news pages people photos policy science shopping sports travel tv xtra yahoo
chat college events finance games guide life music news people picks science shopping sports stock tv weekly xtra yahoo yahooligans
autos business chat computers entertainment events games health life movies music news people science shopping sports travel tv xtra yahoo
autos business computers entertainment events free games health movies music news pages people science shopping sports travel tv xtra yahoo
autos business entertainment finance games health movies music news pages people photos policy science shopping sports travel tv xtra yahoo
Inspector
Default
Apply
Undo
1
2
3

www.yahoo.com
Start time: 01/01/1996
End time: 10/29/2007
Samples: 30
Zoom: 0.08 D= 1.000
Term Scorer: Prevalence
Diff Analyzer: Changed
Periods: 4
Transparency: 1.00
Color:
Keywords:

Figure 9 Histories of MSN (1), Yahoo! (2) and Google (3) homepages (Yahoo! page highlighted *R*=4, zoom=0.08, prevalence term clouds, mm/dd/yyyy format, descriptor window shown).

In another example we compared histories of the homepages of two popular search engines, AltaVista[6] and Google[7] (Figure 7). We can see that the Google homepage was frequently changing with a stable change rate since the beginning of its lifetime. The outlook of the Google homepage, which is popular for its simplicity, has basically remained unchanged since the origin of the page. On the other hand, the AltaVista homepage was updated with relatively large amount of content in the first half of its lifetime. The size of content in the AltaVista homepage during the first half of its lifetime was also much larger than the one of Google's homepage. However, later, around June 2002, the Altavista homepage became much simpler, focusing mostly on providing search functionality and, thus, resembling the well-known appearance of the Google homepage. Since June 2002 the AltaVista homepage has generally remained unchanged.

Looking at Figure 8 we can see that "Google" and "Altavista" keywords are (not surprisingly) the most prevailing keywords in the Google and AltaVista pages, respectively. Common words that can be observed from the comparison of the prevalence term clouds are "service", "image", "video", "advanced", etc. These are terms representing concepts related to search technology on the Web, such as image search, video search or advanced search. More interesting, however, are the keywords unique for both pages. The Google homepage is characterized by "froogle" which is the product comparison service and by the expression "cool" that conveys the well-known Google's free style and its user-friendly approach. On the other hand, "translate" and "audio" are terms specific to AltaVista. The former is related to the popular Babelfish translation service[8], while the latter is due to MP3/audio search function offered by AltaVista. Note that Google did not provide audio search functionality.

Lastly, in Figure 9, we show MSN homepage history contrasted with other rival pages, Yahoo! and Google.

7.3User Studies

First, we have conducted a pilot questionnaire survey to analyze the level of interest in exploring page histories [9]. The questionnaire was administered from the 7th to the 12th of February 2008 to a group of 1000 Internet users in Japan. The subjects were divided into four groups depending on their ages: 20-29, 30-39, 40-49 and 50-59 years old. Each group consisted of 250 respondents, where half were males and the other half were females. Each respondent received a small amount of money to complete the survey. The survey was done in Japanese. Through this study we found that 49.4% users would like to revisit content that has already disappeared from the Web, 29.2% would like to see the content that they previously could not access and 66.8% users wish to know the age of information found on the Web. This confirms a reasonably high level of user willingness to interact with page histories.

Next, we have conducted a detailed user study in order to gather feedback on PHE system and identify the problems and bottlenecks. We have asked 8 subjects (in their 20s; with average computer skills) to perform two tasks lasting 20 minutes each; one for the purpose of gathering knowledge about the history of Yahoo! homepage (task1) and the other for the purpose of comparing histories of Google and AltaVista homepages (task2). As, to the best of our knowledge, there are no similar systems available with which we could compare PHE, we thus necessarily used the Wayback Machine (WM) for this purpose. The arrangement of tasks was organized in such a way that each user did one task with PHE and one with WM, and, the same numbers of users worked on each task using the same system. All users received the prior explanation and saw demonstration of both the PHE and WM systems before embarking on the tasks.

Later we asked the users to answer several questions related to tasks and, in addition, we asked them to provide feedback through a short questionnaire. There were 14 questions for task1 and 8 questions for task2. The questions on tasks ranged from ones on page's age, visual similarities of page histories, time points of sudden outlook or size changes, content's similarity and differences measured by the number of the same and different words in different pages, to the questions on first occurrences of given words.

The users generally, as expected, had problems with completing tasks using WM due to the large number of links to past page snapshots, the lack of visualization support, and the necessity for coming back each time to the directory page in order to visit the other snapshots. It was also hard to spot and remember the changes in visited snapshots. On the other hand, all of questions were successfully answered when using PHE.

Below we show the survey questions for gathering users' feedback on the system and we present their answers in Figure 10.

1. Was it easy to understand PHE?
2. Was it easy to use PHE?
3. Which system was easier to solve the tasks?
4. How would you rate PHE in terms of usefulness?
5. How much were thumbnail images useful?
6. Were the co-occurring keywords useful?

[6] http://www.altavista.com
[7] http://www.google.com
[8] http://babelfish.altavista.com

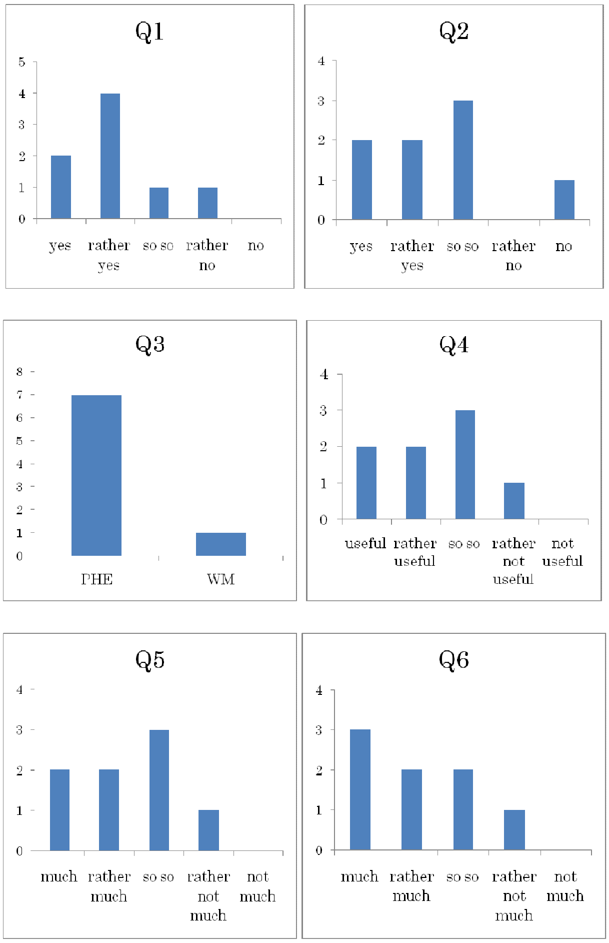


Figure 10 Answers to user questions.

Users confirmed that PHE is generally easy to understand yet sometimes hard to use, and it is better in solving the tasks when compared to WM. Thumbnail images and co-occurring words were considered quite useful. The changes of the interface design into a more intuitive and simple one and adaptation of the system for particular tasks are some objectives for our future work.

Besides these results, we have later asked users for explanation of their decisions in order to obtain more insight about the working of the system. For the question about particular problems in the usage of the system the users admitted that: a) they could not remember the way how term clouds work and are calculated, b) the system was too slow, c) thumbnail images were refreshed too quickly, d) it was unknown on how to look for specific information, e) there were too many options to use, f) it was difficult to operate the system, g) it was hard to find topic terms and f) the display of dates should be more clearer. In general, user friendliness was considered low, and especially the time delay was large due to downloading data from the remote archive and converting snapshots to images. Caching might be thus some partial remedy here.

The problems encountered when using WM were the difficulty in finding the differences among page snapshots, finding specific keywords, necessity for many clicks. When asked on what other functions users would like to have in PHE, they answered: a) showing differences in page content over time displayed on thumbnails, b) providing functions for keyword search, c) more timelines, d) comparison function for many pages, g) finding content by time periods and f) historical image search.

We then asked users for what tasks they would like to use PHE in the future. The answers were: a) for knowing pages, b) checking the trustworthiness of pages through watching their changes, c) checking page histories, d) looking for information on past events recorded in the newspapers, online articles, pictures and comments and e) keyword finding. One user said that PHE may be also useful in public libraries.

7.4 Discussion

Intuitively, the effectiveness of Page History Explorer is constrained by the availability of data and its character. Below, we discuss the issues that pose challenges for this research.

Temporal page analysis can be carried to the extent to which data can be obtained. Therefore, the proposed method may be less useful for relatively young pages, dynamically generated pages or pages for which only few snapshots were archived. In addition, if a page remained unchanged over its lifetime, then visualizing its history may not be very interesting to users.

PHE is based on a URL-based access, thus, users must know the URLs of interesting pages. Related is also the problem of finding pages that have been already removed from the Web, pages that changed their URLs or that borrowed content from other pages. The example of Web directories discussed in the previous section demonstrates the challenges in tracing content movements over time. In fact, Yahoo! homepage contained also simple page directory listings as a part of its content long time before the establishment of the Yahoo! Directory page. Although, some solutions based on searching for similar content on the Web have been already proposed [6,14], the problem of locating moved pages or their content parts is still unresolved.

Our system may be slow when working in the online mode, especially, when the number of snapshots is large. Ideally, PHE or some of its components could be implemented as a Web service associated with Web archives.

Finally, we would like to emphasize that, ideally, the complete historical analysis of pages should not be limited only to their content. For example, changes in the numbers of in-links could be utilized for better understanding the document's evolution and its relation to other pages. Unfortunately, such data is currently unavailable for arbitrary pages. In addition, historical analysis and comparison of page context, such as other pages within the same Web site, would be probably also interesting to users, provided, there is enough data available for this.

8. Conclusions and Future Work

Web pages are usually represented and judged by the content of their current versions. However, as Web pages are durable and changeable documents, they should be also described by their long-term topics and characteristics for providing a more complete image.

In this paper, we proposed effective interactive approach for visualizing summaries of page histories based on data extracted from Web archives. Using the proposed solutions, it is possible to get a grasp of page histories without resorting to manual analysis of their numerous snapshots. The snapshot visualization on 2D space reveals the change pattern over time, the general outlook and the characteristics of pages in the past, while the term clouds provide different kinds of aggregated views on their historical content. This kind of temporal representation offers contextual data that can be used for facilitating understanding of pages or even for predicting their future states. An important feature of PHE is its comparison function for determining similar and different aspects in histories of different pages. Overall, the proposed system helps users to effectively obtain overview of page histories and let them interact with page past in an easy way.

Our future work is related to integration of the proposed visualization approach into Web browsing. One proposal is to implement components of PHE into traditional browsers so that, for example, temporal term clouds or 2D space could be automatically projected next to the content of visited pages. In addition, we plan to extend this work for exploring and comparing the histories of whole Web sites.

Acknowledgments

The authors want thank Soungwoong Yoon for his help with the final version of the paper. This work has been partially supported by National Institute of Information and Communications Technology, Japan and by SCOPE (Ministry of Internal Affairs and Communications).

References

[1] Adar, E., Dontcheva, M., Fogarty, J., and Weld, D.S. Zoetrope: interacting with the ephemeral web. *Proceedings of UIST 2008*, ACM Press, 2008, 239-248.

[2] Bar-Yossef, Z., Broder, A. Z., Kumar, R. and Tomkins, A. Sic Transit Gloria Telae: Towards an understanding of the Web's decay. *Proceedings of WWW 2004*, ACM Press, 2004, 328–337.

[3] Chi, E. H., Pitkow, J., Mackinlay, J., Pirolli, P., Gossweiler, R. and Card, S. K. Visualizing the evolution of Web ecologies. *Proceedings of CHI 1998*, 1998, 400–407.

[4] Dubinko, M., Kumar, R., Magnani, J., Novak, J., Raghavan, P. and Tomkins, A. Visualizing tags over time. *Proceedings of WWW 2006*, ACM Press, 2006, 193-202.

[5] Gulli, A. and Signorini, A. The indexable web is more than 11.5 billion pages. *Proceedings of WWW 2005*, ACM Press, 2005, 902-903.

[6] Harrison, T.L. and Nelson, M.L. Just-in-time recovery of missing web pages. Proceedings of *ACM HT 2006*, ACM Press, 2006, 145-156.

[7] IIPC's Access Working Group. *Use cases for Access to Internet Archives*, 2006, http://netpreserve.org/publications/iipc-r-003.pdf

[8] Jatowt, A., Kawai, Y., Nakamura, S., Kidawara, Y. and Tanaka, K. Journey to the past: proposal for a past Web browser. *Proceedings of ACM HT 2006*, ACM Press, 2006, 134-144.

[9] Jatowt, A., Kawai, Y. Ohshima, H. and Tanaka, K. What can history tell us? Towards different models of interaction with document histories. *Proceedings of ACM HT 2008*, ACM Press, 2008, 5-14.

[10] Jatowt, A., Kawai, Y. and Tanaka, K. Detecting age of page content. *Proceedings of the 9th ACM WIDM 2007*, 137-144.

[11] Jatowt, A., Kawai, Y. and Tanaka, K. Personalized detection of fresh content and temporal annotation for improved page revisiting, *Proceedings of DEXA 2006*, Springer LNCS 4080, 2006, 832-841.

[12] Liu, B., Zhao, K. and Yi, L. Visualizing web site comparisons. *Proceedings of WWW 2002,* 2002, 693-703.

[13] Masanes, J. (ed.). *Web archiving*. Springer Verlag. Berlin Heidelberg, Germany, 2006.

[14] McCown, F., Diawara, N. and Nelson, M.L. Factors affecting website reconstruction from the web infrastructure. *Proceedings of JCDL 2007*, ACM Press, 2007, 39-48.

[15] Mohr, G., Carpenter, K. and McCown, F. Academic Research and the Internet Archive's Web Archives. How to mine and analyze ten+ years of virtual activity. *Tutorial at JCDL 2007*, 2007.

[16] Myers, E. W. An O(ND) difference algorithm and its variations. *Algorithmica 1*, 1986, 251-266.

[17] Nadamoto, A. and Tanaka, K. A comparative web browser (CWB) for browsing and comparing Web pages. *Proceedings of WWW 2003*, ACM Press, 2003, 727-735.

[18] Shipman F. M. and Hsieh H. Navigable history: a reader's view of writer's time: Time-based hypermedia. *New review of hypermedia and multimedia*, Taylor & Francis, vol. 6, 2000, 147-167.

[19] Teevan, J., Dumais, S.T., Liebling, D.T., Hughes, R.L. Changing how people view changes on the web. *Proceedings of UIST 2009*, ACM Press, 237-246.

[20] Toyoda, M. and Kitsuregawa, M. Extracting evolution of Web communities from a series of Web archives. *Proceedings of ACM HT 2003*, ACM Press, 2003, 28-37.

[21] Viégas, F., Wattenberg, M. and Dave, K. Studying cooperation and conflict between authors with history flow visualizations. *Proceedings of CHI 2004*, ACM Press, 2004, 575-582.

Appendix

We explain here how to estimate term frequency in a change over time. First, we list heuristics that we are going to use for the prediction of transient content in page histories.

1. The larger is the changed content between any two snapshots (as defined in Section 4.1), the larger is the amount of transient content that could appear between these snapshots. This is because, intuitively, the area covered by this content is the likely area where the transient content could appear. Note that we neglect the possibility of content reverts here.
2. The longer is the time period between any two snapshots, the higher is the probability of transient content occurrence in this period. This is reasonable, since, in general, one can expect more changes occurring in longer time periods.
3. The higher is the overall change frequency of a page, the higher is the likelihood of transient content inside any time period during page's lifetime.

Suppose there is a sequence of snapshots, $s_0,\ldots,s_{N-1}$, with their respective timestamps $t_0<\ldots<t_{N-1}$ within the user-defined time period T. Let us focus on the pair of two consecutive snapshots, s_i and s_{i+1}. Figure 11 shows the snapshot pair with estimated term frequencies within $[t_i,t_{i+1}]$.

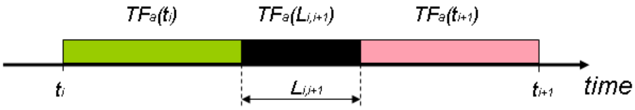


Figure 11 Term frequency between two consecutive snapshots with a supposed transient period marked by black color.

Let $L_{i,i+1}$ denote the supposed length of the transient period inside $[t_i,t_{i+1}]$, that is, the period of the occurrence of any transient content. Introducing $L_{i,i+1}$ helps us to represent the frequency of an arbitrary term a inside $[t_i,t_{i+1}]$ as a linear combination of term frequencies in the both snapshots and the estimated frequency of the term during $L_{i,i+1}$ expressed as $TF_a(L_{i,i+1})$.

$$TF_a([t_i,t_{i+1}])=\frac{1-\alpha_{i,i+1}}{2}*(TF_a(t_i)+TF_a(t_{i+1}))+\alpha_{i,i+1}*TF_a(L_{i,i+1}) \qquad (6)$$

Parameter $\alpha_{i,i+1}$ is used here to combine both estimates and its calculation will be explained later. $TF_a(L_{i,i+1})$ is estimated as a linear combination of the term frequency inside the static content, expressed as $TF_a^s(L_{i,i+1})$, and the expected term frequency inside the transient content, $TF_a^c(L_{i,i+1})$ within $L_{i,i+1}$.

$$TF_a(L_{i,i+1})=\beta_{i,i+1}*TF_a^c(L_{i,i+1})+(1-\beta_{i,i+1})*TF_a^s(L_{i,i+1}) \qquad (7)$$

$TF_a^s(L_{i,i+1})$ can be easily found after comparing the content of both snapshots (as explained in Section 4.1). $TF_a^c(L_{i,i+1})$, on the other hand, is estimated by considering the overall statistics of term a calculated over all the available snapshots in T. Hence, it can be actually written as $TF_a^c(T)$ as it does not depend on any particular sub-period inside T. $TF_a^c(T)$ is the average frequency of term a inside the combined added and deleted parts of all page snapshots in T (Equation 8). The idea here is that the term has a high chance to occur in the transient content if it often occurs inside changes within time period T.

$$TF_a^c(T)=\frac{1}{M}\sum_{i=0}^{N-1}\frac{c_a^{ad}(t_i)+c_a^{de}(t_i)+d}{size^{ad}(t_i)+size^{de}(t_i)+d} \qquad (8)$$

M is the number of snapshots in which any content changes have been observed. $C_a^{ad}(t_i)$ and $c_a^{de}(t_i)$ denote respectively the counts of term a inside the added and deleted content at snapshot s_i. $size^{ad}(t_i)$ and $size^{de}(t_i)$ denote the number of terms in added and deleted content at s_i, respectively. We use a constant d here in order to cope with the case when the term was not observed inside the changed part of any snapshot. In other words, we add some non-zero probability mass to unseen events in a similar fashion as smoothing is done in information retrieval (d is equal to 1 by default).

The weighting parameter $\beta_{i,i+1}$ is the only factor in Equation 7 that is still unknown. It is estimated based on heuristic 1 by computing the relative size of the changed content inside $[t_i,t_{i+1}]$. From Section 4.1 we remember that the changed content in $[t_i,t_{i+1}]$ is the union of added content at s_{i+1} and deleted content at s_i.

$$\beta_{i,i+1}=\frac{size^c([t_i,t_{i+1}])}{size^c([t_i,t_{i+1}])+size^s([t_i,t_{i+1}])} \qquad (9)$$

Lastly, we need to explain the procedure for estimating parameter $\alpha_{i,i+1}$ used in Equation 6. We use here heuristics 2 and 3. That is, the longer the time span between t_i and t_{i+1} and the higher the total change frequency of the page are, the higher is the probable length of the occurrence of transient content within $[t_i,t_{i+1}]$. This is because there is a higher likelihood of several consecutive changes occurring in $[t_i, t_{i+1}]$.

$$\alpha_{i,i+1} = \frac{t_{i+1} - t_i}{T} * \frac{M + d}{N + d} \qquad (10)$$

We use parameter d here again to account for the case when no change was observed from the comparison of all available page snapshots.

In order to determine $TF_{a,b}([t_i, t_{i+1}])$, which appeared in Equation 4, Equations 6, 7 and 8 are used mutatis mutandis. That is, frequencies of term a are multiplied with the corresponding frequencies of term b in Equations 6 and 7, and the counts of term a are multiplied with the corresponding counts of term b in Equation 8.

Adam Jatowt received MS in Electronics and Telecommunications from the Technical University of Lodz, Poland in 2001. In 2005 he received PhD in Information Science and Technology from the University of Tokyo, Japan. He has worked as a research fellow at the National Institute of Information and Communications Technology, Japan in 2005. From 2006 to 2009 he worked as an assistant professor and since 2010 he has been working as an associate professor at the Kyoto University. His research interests include temporal information processing, web mining, document comprehension and social bookmarking.

Yukiko Kawai received the MS and PhD degrees in Information Science and Technology from Nara Institute of Science and Technology, in 1999 and 2001, respectively. She has worked as a research fellow at the National Institute of Information and Communications Technology from 2001 to 2006. Since 2006 she has been a lecturer at Kyoto Sangyo University. Her research interests include data mining, information analyzing and Web information retrieval.

Katsumi Tanaka received the BS, MS and PhD degrees in Information Science from Kyoto University, in 1974, 1976 and 1981, respectively. In 1986, he joined the Department of Instrumentation Engineering, faculty of Engineering at Kobe University, as an associate professor. In 1994, he became a full professor at the Department of Computer and Systems Engineering Department, Faculty of Engineering, Kobe University. Since 2001, he has been a professor of the Graduate School of Informatics, Kyoto University. Currently, he is a vice-dean of the school. His research interests include database theory and systems, Web information retrieval, and multimedia retrieval.