A Visualization Technique for Access Patterns and Link Structures of Web Sites

Makiko Kawamoto, Takayuki Itoh
Ochanomizu University
{makiko, itot}@itolab.is.ocha.ac.jp

Abstract

There have been two types of Web visualization techniques: visualization of Web sites themselves based on such as link structures or lexical contents, and visualization of browsers' behaviors. We think that integration of such two visualization techniques is very useful for Web site management, and therefore we are currently studying on visualization of access pattern and link structure on a single screen.

This paper presents a Web visualization technique using our own multiple-category-embedded graph visualization technique. The presented technique constructs link structures using crawler software, and access patterns from access log files. It then integrates them and visualizes by our graph visualization technique. We expect that users can visually understand the relationship between access patterns and link structures, and utilize the knowledge for design and management of Web sites. This paper shows our case study and discusses typical access patterns we observed by the technique.

1 Introduction

Visualization of Web information is an active research topic for over ten years. Here, Web information to be visualized is divided into the following two types: 1) Web site contents information including link structures and lexical contents, and 2) browser information including access statistics. We think it is interesting to simultaneously visualize such two kinds of Web information, because we can expect to discover fruitful knowledge to be used for design and management of Web sites.

This paper presents a visualization of access patterns and link structures of Web sites, applying a hybrid force-directed and space-filling graph layout technique [5]. Here, this paper defines an access pattern as a set of Web pages which are commonly accessed by multiple browsers. We think the technique can contribute to visually observe the adequateness of the access patterns and link structures. Also, we think it can contribute to discuss reconstruction of link structures reflecting access patterns, and redesign of contents of Web pages.

This technique first constructs a hierarchical graph; nodes correspond to Web pages, edges correspond to hyperlinks, and hierarchy corresponds to the directory structure of the Web site. The technique then calculates adequate positions of the nodes on the display space, where it satisfies the following four conditions:

Condition 1: Web pages which have the same access patterns are placed closer on the display space.

Condition 2: Web pages connected by hyperlinks are placed closer on the display space.

Condition 3: Web pages and their clusters do not overlap each other on the display space.

Condition 4: Utilization of the display space is maximized.

This paper shows our case study to demonstrate the effectiveness of the presented technique, and discusses typical access patterns discovered by using the technique.

2 Related Work

Web-related visualization has been an active research topic since 1990s, and several survey papers and Web sites have been published [1]. Early works [3] [8] simply applied generic tree or graph visualization techniques for single domain, mainly for interactive navigation of moderately-sized Web sites. Recent works apply various graph mining techniques to visualize huge-scale link structures, mainly for analysis of cyber worlds. Not only visualization of link structures, but also visualization of access statistics of Web sites [11] is also useful for analysis of Web sites.

Behavior of browsers is very interesting information for Web designers and administrators, and therefore analysis and mining of such behavior are also active research topics. Nasraoui et al. defined a similarity calculation scheme between behavior of two browsers, and applied fuzzy clustering to them [9]. Pitkow et al. presented a prediction model of user surfing paths based on Markov models [10]. Davison presented a technique to predict actions of browsers by detecting their interest from textual contents [4].

Such behavior of browsers can be effectively visualized by mapping the information onto Web visualization techniques. Several studies have visualized Web access pat-

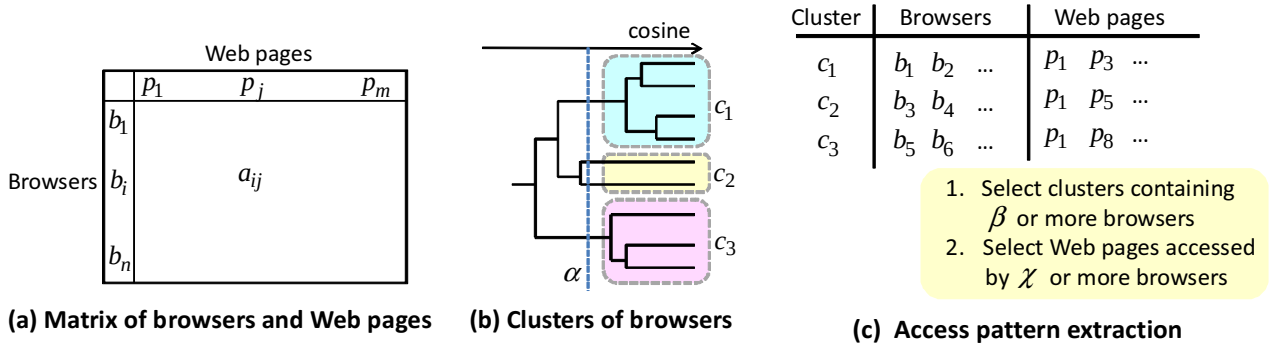

Figure 1: Processing flow of access pattern extraction.

terns [2] [7]; however, they represented just a few access patterns after extracting a small portion of the link structures of the Web. Generally, it is a difficult problem to realize visualization satisfying good visual properties for both link structures and additional information (access patterns in this case). Our multiple-category-embedded graph visualization technique [5] has a good visual property to represent both link structure and additional category information.

3 Presented Technique

This section describes the processing flow of the presented technique. The technique constructs the input data by integrating the following information,

- access patterns extracted from Web access log files, and
- link structures constructed using Web crawler software.

This technique supposes to use standard Web access log files, which record IP addresses of browsers, times of the accesses, filenames which are accessed, URLs where the accesses came from, and so on.

Here, we cannot always assume that we can retrieve every browsing operation of the browsers. For example, browsing software may display cached Web pages when the browsers click the reverse button. In this case, no communication with Web servers happens, and therefore their actions are not recorded to the Web access log files. The presented technique supposes the following three stances:

Stance A: Do not visualize any access paths of browsers.

Stance B: Visualize access paths extracted only from Web access log files.

Stance C: Visualize access paths, after completely tracking actions of browsers.

By applying [Stance A], we can assume the access paths of browsers by visualizing Web access patterns using the presented technique; however, it is difficult to validate the assumptions. By applying [Stance B], we can visually recognize the access paths; however, the visualized information misses some operations of browsers. On the other hand, [Stance C] has a big limitation; we may need to prepare special environment to completely track the actions of browsers, for example, tracking by Web analysis services while embedding tracking codes to every Web page, or using special browsing software. We implemented the presented technique supposing [Stance A] and [Stance B]; however, the technique is also available while supposing [Stance C].

3.1 Access Pattern Extraction

This section describes our implementation of Web access pattern extraction. Figure 1 is an illustration of the processing flow of Web access pattern extraction.

The implementation first parses a Web access log file, and constructs lists of IP addresses of browsers and accessed URLs. Here, our implementation records URLs which do not point to multimedia contents files (e.g. images, sounds) to the list. It then constructs a matrix where rows correspond to n IP addresses of browsers b_1 to b_n, and columns correspond to m accessed URLs of Web pages p_1 to p_m. It also fills elements of the matrix a_{ij} by the total number of accesses from the i-th IP address to the j-th URL.

The implementation then applies a hierarchical clustering algorithm to divide the browsers based on the similarity of the sets of accessed Web pages. Here, it treats numbers of accesses to each Web page by a browser as m-dimensional vectors, and calculates the cosine between every pair of the browsers [9]. It then constructs a dendrogram by recursively coupling the browsers. The process first treats the browsers as nodes, and combines the pair of nodes which has the largest cosine value into a single

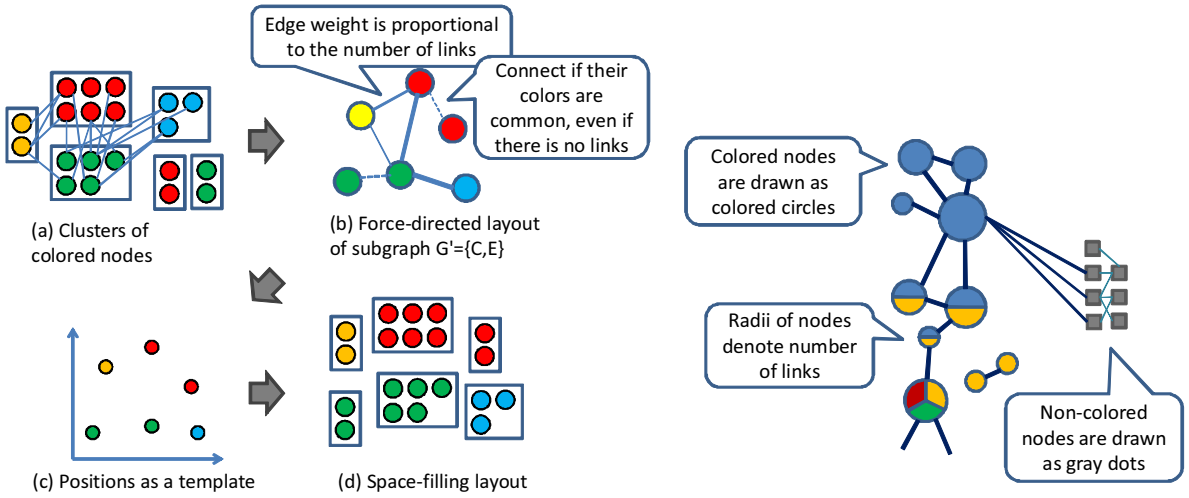

Figure 2: (Left) Processing flow of the hybrid layout technique. (Right) Drawing style of our implementation. It distinguishes nodes by their colors, and links by their thicknesses and transparencies.

node. Recursively finding the pairs of nodes which have the largest cosine values and combining into the single nodes, the process constructs the dendrogram. The implementation then generates clusters of the browsers c_1 to c_l, where l is the number of clusters, by cutting the dendrogram by a user-defined threshold α.

Finally, the implementation extracts sets of Web pages as Web access patterns. It selects the clusters which contains β or more browsers, where β is a user-defined number. It then extracts a set of Web pages which are accessed by γ percent or more of the browsers, where γ is also a user-defined value.

It may happen that hundreds of clusters are generated and therefore hundreds of Web access patterns are generated by using the implementation. Currently we select 10 or 20 meaningful patterns from the extracted patterns subjectively; however, we would like to implement an automatic Web access pattern selection technique, or sophisticated GUIs so that users can intuitively and interactively select meaningful sets of Web access patterns.

3.2 Link Structure Construction

The technique constructs link structures by using Web crawler software. We suppose to construct link structures of Web domains that we can obtain Web access log files. Our current implementation recursively accesses Web pages in the domains starting from the top page of the domains, using JSpider [6], a famous open source Web crawler software.

3.3 Data Integration

The technique integrates the link structures and Web access patterns. It first matches URLs in the link structure and Web access pattern data. It then constructs a tree structure based on directory structures of the URLs, where branch-nodes of the tree correspond to directories, and leaf-nodes correspond to Web pages. Finally, the technique embeds hyperlinks and Web access patterns to the leaf-nodes.

The technique then parses the Web access log file again, and extracts the pairs of URLs, where one of them is the accessed URL, and the other is the URL where the access came from. It then matches the pairs with hyperlinks, and counts the numbers of extracted pairs for each hyperlink. The counting result denotes the frequency of passage of browsers for each hyperlink.

3.4 Visualization

The technique applies our own hybrid force-directed and space-filling graph layout technique [5] for the integrated link structure and access pattern data.

Figure 2(Left) shows the processing flow of the hybrid graph layout technique. Figure 2(Left)(a) is an example of a set of clusters of colored nodes. Figure 2(Left)(b) is an example of the subgraph. Here, edges of the subgraph connect clusters if at least one pair of the nodes in the clusters is linked, or the clusters share same colors. The technique then calculates the position of the clusters by applying a force-directed layout technique that can deal with weighted edges. It then applies our Treemap-like rectangular packing algorithm to realize space-filling layout. The algorithm refers the positions of clusters calculated by the force-directed layout as a template, as shown in Figure 2(c). Figure 2(d) illustrates the result of the space-filling layout process. The technique realizes preferable cluster layout, because the force-directed algorithm satisfies [Condition 1] and [Condition 2], while a space-filling algorithm

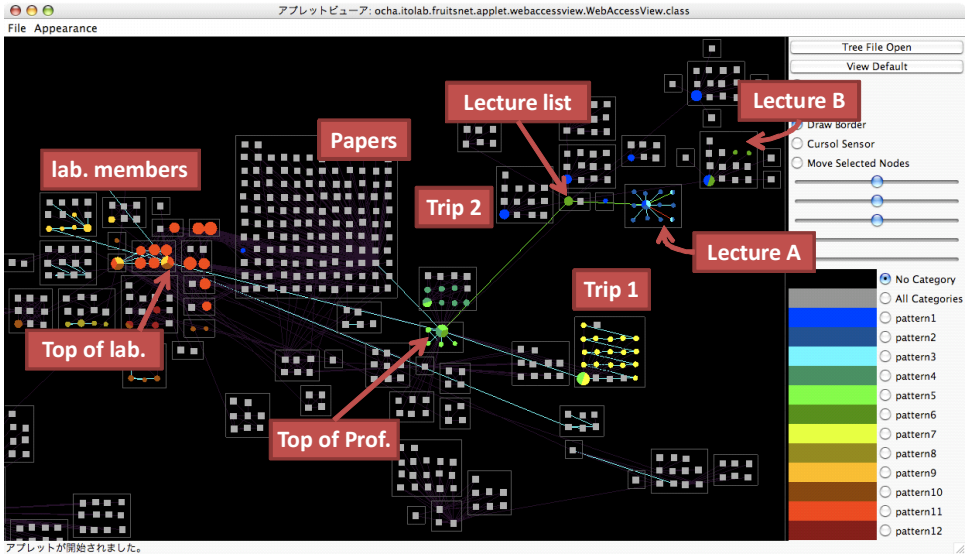

Figure 3: Visualization of access patterns and paths.

satisfies [Condition 3] and [Condition 4].

Figure 2(Right) shows how to draw nodes and links with our current implementation. Our implementation assigns independent colors to the access patterns, and draws nodes which belong to one or more access patterns as colored circles. When a node belongs to multiple patterns, the implementation divides the circle into multiple fans, similar to a pie chart. It calculates radii of nodes from the number of connected links because we would like to emphatically represent hub nodes. For nodes which do not belong to any access patterns, however, our implementation simply draws smaller gray dots.

At the same time, the implementation supports two schemes to assign colors and transparencies to links. The first scheme divides links based on the following three levels:

- Links are drawn relatively thick and bright, if they connect two colored nodes.
- Links are drawn moderately thick and bright, if they connect a colored node and a non-colored icon.
- Links are drawn relatively thin and transparent, if they connect two non-colored nodes.

The second scheme assigns colors to links based on the numbers of browsers passage; red links denote most frequent paths, green or blue links denote moderate paths, and transparent links denote rarely accessed paths.

4 Case Study

This section introduces a case study that visualized the link structure and access patterns of Web sites of authors' laboratory (http://itolab.is.ocha.ac.jp/), where the data is constructed from the Web access log file in July 2009. We implemented the technique with Java JDK 1.5.0, and tested on an Apple MacBook (CPU 2GHz, RAM 1GB) running Mac OS 10.4.11.

Figure 3 shows the window of our implementation. It displays GUI widgets on the right end of the window, for viewing and focus+context operations, and selection of specific Web access patterns. The lower-right part of the GUI is the list of colors and buttons that correspond to Web access patterns. Pressing one of the buttons, the implementation highlights the corresponding nodes, and links connected to the highlighted nodes. Also, the implementation can locally modify the node layout so that the highlighted nodes are concentrated on the display.

4.1 Visualizing Access Patterns and Paths

The visualization result in Figure 3 shows major access patterns and paths in the Web site of our laboratory.

The left side of the drawing area displays Web pages of projects and students of our laboratory. An orange access pattern denotes that browsers accessed to Web pages of projects. A dark yellow access pattern denotes that browsers simultaneously accessed to Web pages of several students. A brown access patterns denotes that browsers independently accessed to Web pages of specific students.

On the other hand, the center of the drawing area displays the Web page of the professor, who has Web pages of his technical papers, lecture materials, and trip reports. The visualization result denotes that many browsers passed the link between the top pages of the laboratory and the professor; however, these two top pages did not share any

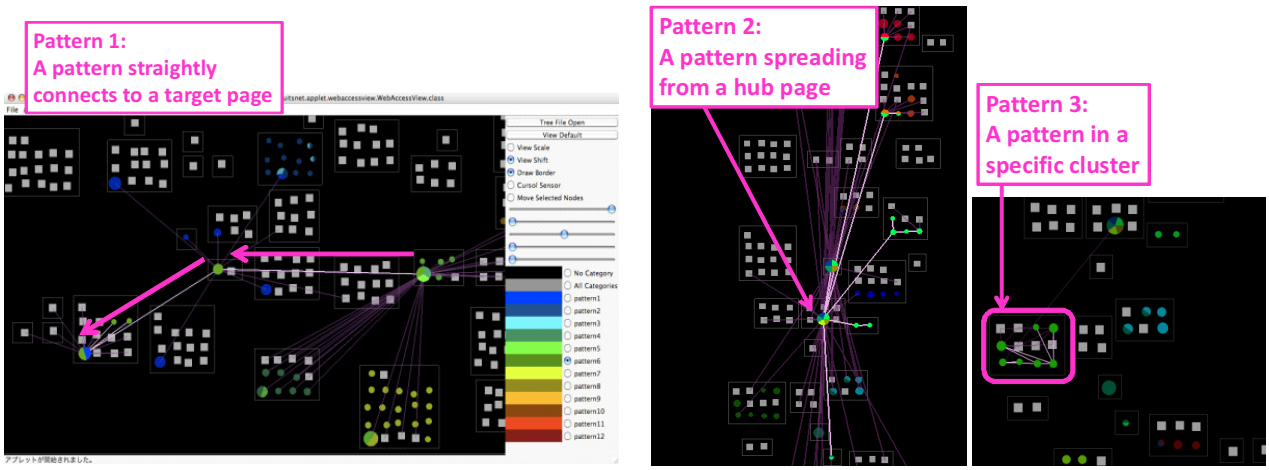

Figure 4: Three types of Web access patterns we discovered from the visualization result applying to the Web site of our laboratory.

access patterns. A bright yellow access pattern denotes that browsers sequentially accessed to one of his trip reports. The professor had two classes for undergraduate students in this semester, and many students accessed to Web pages of the lecture A from his top page, because the link between the top page and a page of lecture A is drawn in bright green. It seems that many students taking the lecture B also accessed starting from the top page of the professor; however, the number of the students is much smaller than students taking lecture A, because the links between the top page and pages of lecture B are not drawn in bright colors.

As above mentioned, this visualization result explained us active links and multiple access patterns over the Web site of our laboratory, and contributed to assist the design change of the Web site.

4.2 Typical Patterns

We discovered three typical Web access patterns during this case study, as shown in Figure 4. The following are the typical Web access patterns:

Pattern 1: Patterns straightly connected to the target pages.

Pattern 2: Patterns radially spreading from hub pages.

Pattern 3: Patterns closed in specific directories.

Figure 4(Left) shows an access pattern of students who accessed to the Web page of materials and sample programs of a specific class. The example shows that many of students did not directly access to the Web page. Instead of, they accessed started from the top page of the professor, and finally arrived at the destination page by accessing linked pages. Figure 4(Center) shows an access pattern spread from the Web page of the member list to the Web pages of the members. Figure 4(Right) shows an access pattern is closed in a specific directory, which is the directory of a specific member.

We think that discovery of such typical patterns will contribute to the improvement of Web sites. Access patterns categorized in [Pattern 1] denote that browsers are interested in a specific page. If we find such browsers needed to pass many pages to the destination page, Web designers may need to change the link structure so that the browsers can access to the destination page more easily. Access patterns categorized in [Pattern 2] denote that browsers access to many pages from the hub pages. We are often interested in the combination of Web pages contained in the access patterns. Web designers may think of changing the design of the hub pages so that all the pages supposed that browsers are interested in are accessed. Access patterns categorized in [Pattern 3] denote that browsers access to many pages related to specific persons or topics. We may need to pay attention to Web pages that are not included in the access patterns in the specific directory. Web designers may want to improve so that browsers will access to such Web pages.

As above mentioned, the visualization technique represents various access patterns and they will suggest Web designers how the Web sites are to be improved.

5 Conclusion

This paper presented a visualization technique for link structures and access patterns of Web sites. The technique constructs link structures by crawler software, and access patterns from Web access log files. It then integrates them into a hierarchical graph, and displays by applying our own hybrid force-directed and space-filling graph visualization

technique. This paper demonstrated that we discovered three typical access patterns by using the technique. The paper also discussed that the discovery can contribute to improvement of Web sites.

Our potential future work includes the following issues.

We need to overcome some issues of our hybrid graph visualization technique. It is currently difficult to represent hundreds of access patterns by the technique, because they are distinguished just by colors. We need to think of other design to represent more access patterns in a single visualization result. Also, we need to improve the scalability of the technique; we would like to speed up the technique for large scale Web sites containing ten thousands of Web pages and hundreds of access patterns.

We would like to test more variety in drawing the graph. Our current implementation of graph visualization does not support directed graphs, so we would like to support it so that we can display directions of the accesses. Also, we would like to represent frequency of accesses for each Web page by radii or heights of nodes, instead of representing the numbers of links. Color is another issue: it may be confusing for some people because colors of nodes and links denote independent meanings. We would like to have experimental tests how it is really confusing.

It is often interesting to observe access patterns that Web designers expected to be realized but they were not actually observed. We would like to extend the visualization technique so that users can set such expected access patterns.

Our current implementation of access pattern extraction process is so naive that we do not expect it can perfectly discover meaningful access patterns. Also, it requires users to manually and subjectively select small number of access patterns. We would like to improve the implementation.

A challanging interest for us is visualizing the correlation between access patterns and contents of Web pages. We would like to extract keywords that are highly correlated to any of access patterns, and visualize the distribution of keywords as well as access patterns.

This work has been partially supported by Japan Society of the Promotion of Science under Grant-in-Aid fir Scientific Research.

References

[1] An Atlas of Cyberspace, http://personalpages.manchester.ac.uk/staff/m.dodge/cybergeography/atlas/

[2] J. Cuqini, J. Scholtz, VISVIP: 3D Visualization of Paths through Web Sites, *10th International Workshop on Database and Expert Systems Applications*, 1999.

[3] D. Durand, P. Kahn, MAPA: A System for Inducing and Visualizing Hierarchy in Websites, *9th ACM Conference on Hypertext and Hypermedia*, pp. 66-76, 1998.

[4] B. D. Davison, Predicting Web Actions from HTML Content, *13th ACM Conference on Hypertext and Hypermedia*, pp. 159-168, 2002.

[5] T. Itoh, C. Muelder, K.-L. Ma, J. Sese, A Hybrid Space-Filling and Force-Directed Layout Method for Visualizing Multiple-Category Graphs, 2009 IEEE Pacific Visualization Symposium, pp. 121-128, 2009.

[6] JSpider, http://j-spider.sourceforge.net/

[7] N. Labroche, M.-J. Lesot, L. Yaffi, A New Web Usage Mining and Visualization Tool, *19th IEEE International Conference on Tools with Artificial Intelligence*, pp. 321-328, 2007.

[8] T. Munzner, H3: Laying Out Large Directed Graphs in 3D Hyperbolic Space, *IEEE Symposium on Information Visualization ' 97*, pp. 2-10, 1997.

[9] O. Nasraoui, H. Frigui, A. Joshi, R. Krishnapuram, Mining Web Access Logs Using Relational Competitive Fuzzy Clustering, *Eight International Fuzzy Systems Association World Congress*, 1999.

[10] J. Pitkow, P. Pirolli, Mining Longest Repeating Subsequences to Predict World Wide Web Surfing, *2nd conference on USENIX Symposium on Internet Technologies and Systems*, pp. 139-150, 1999.

[11] Y. Yamaguchi, T. Itoh, Y. Ikehata, Y. Kajinaga, Interactive Poster: Web Site Visualization Using a Hierarchical Rectangle Packing Technique, *IEEE Symposium on Information Visualization*, 2002.