Visualizing Web Navigation Data with Polygon Graphs

Jiyang Chen, Tong Zheng, William Thorne, Daniel Huntley, Osmar R. Zaïane and Randy Goebel
Department of Computing Science
University of Alberta, Edmonton, Alberta, Canada
{jiyang, tongz, thorne, dhuntley, zaiane, goebel}@cs.ualberta.ca

Abstract

As the volume of digitally accessible information grows, there is increasing pressure on the development of data visualization methods to enable humans to interpret that data. We provide a description of our WebViz system, as a tool to visualize both the structure and usage of web sites. We illustrate the use of our visualization paradigm by introducing polygonal graphs layered on top of our adaptation of radial disk trees. In our system, the structure of a web segment is rendered as a radial tree, and usage data can be extracted and layered as polygonal graphs. By interactively creating and adjusting these layers, a user can develop real time insight into the data. We present the system, show the idea of interactive visual operators, and provide some examples that help show the value of the specific visualization techniques, as well as the interactive use of those techniques.

Keywords—**Web Visualization, Visual Data Exploration.**

1 Introduction

The rapidly developing field of data visualization includes some interesting connections with machine learning, especially in the context of drawing inferences from large datasets like segments of the World Wide Web (WWW) (e.g., [9, 15]). In contrast to a visualization paradigm where visualization techniques are applied to interpret large datasets or the results of learning from those data sets (e.g., [2, 14]), we have been developing an integration of visualization techniques *within* the learning process. Our work considers a combination of theoretical and practical approaches, which include the ongoing development of a web visualization system called WebViz [12]. Our motivation is to use visualization more interactively, as a companion to various data mining methods, to provide visual guidance to interesting results. In this paper, we describe an aspect of our work that builds and applies a new kind of visualization object we call a *polygon graph*.

The polygon graph is a new general visualization object that we use as a visual data mining aid. The method can be applied on most of the tree structure visualization to help discover knowledge patterns of implicit relations between data variables. Here we show how we incorporate the polygon graph method in our visualization system, and demonstrate its advantages in several interactive analysis examples on web navigational data.

The remainder of the paper is organized as the following: we first discuss pertinent related work in Section 2. In Section 3, we present our visual data paradigm and mining system. We then introduce the polygon graph visualization technique in Section 4 and illustrate the use of polygon graphs to find patterns during the visual data exploration and mining process, before concluding in Section 5.

2 Related Work

Realizing the importance of visualization in understanding the connectivity structure of a web site, information visualization researchers have developed a variety of tools to aid web structure mining and further web usage analysis. Early systems [6, 8] were designed for web administrators and designers to create a dynamic view of the web site structure. Although these systems provide good methods for navigating the visualized structure, for example, by selecting expected pages to show their detail in the focused area, they have not considered the integration of web usage data visualization and knowledge pattern extraction. In order to localize any usage information vis-à-vis the web site, researchers began the development of techniques to visualize the web structure as a background of the usage records. For example, Ed Chi et al.'s notable DiskTree [3] visualized web usage information together with the site structure. In that system, a breadth-first search algorithm is used to map the graph representing the web structure into a radial pattern of rings or "disk tree" layout, which can be used to visualize web site evolution, web usage trends over time, and evaluation of information "foraging" [2]. Crowds Dynamics [7] is another usage analysis system that treats the web site as a social space and illustrates how the web user navigate the web by dynamic animation as a documentary movie for real-time user behaviour. Unfortunately, Minar's system is only practical for small web sites. 3D represen-

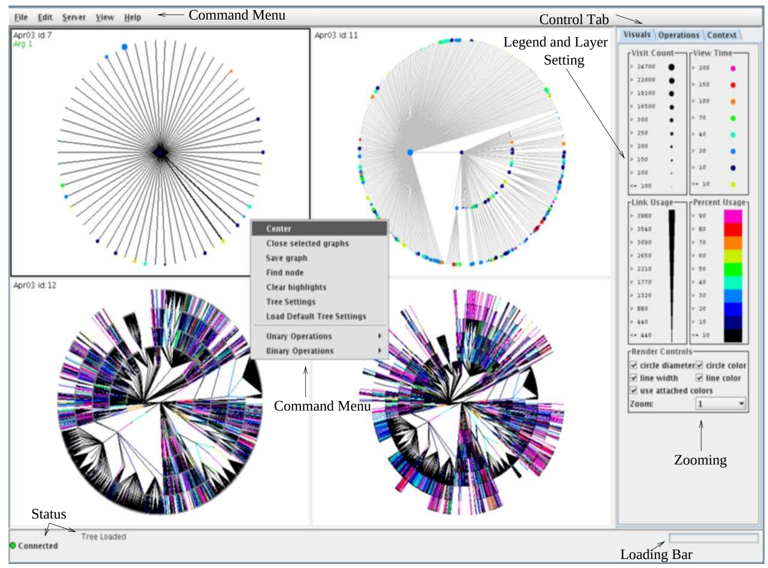


Figure 1: WebViz System

tation and layout techniques have been applied to visualize the web structure and usage in [13]. However, the results have severe occlusion problems and are thus difficult to understand, which limits its value as an information visualization tool.

Any network, especially the WWW, is composed of nodes and links between them. Usually we can consider the structure of any website as a graph, with each node representing a web page, and each edge representing the hyperlink that connects two pages. Researchers propose many representations to help understand the structure of a website. Most of them depict the structure as an acyclic connected graph, in other words, a tree. Chi's radial disk tree remains one of the most popular methods to visualize web structure [3]. The root of a disk tree is selected as a starting page for the breadth-first rendering algorithm. The resulting visualization is a concentric circular hierarchical layout which has the root node in the middle of the circles. Nodes in outer circles are children of the inner circle's nodes, which means that the corresponding pages of the outer nodes can be connected from that of the inner nodes in the web structure. The radial tree layout uses screen space more efficiently than a vertical layout, and easily conveys a visual sense of distance between web pages. Several visualization systems have adapted the disk tree layout to represent various aspects of web structure [1, 9, 10]. Other similar approaches include ConeTree [11], which starts from a node and display the children in the base of a cone that has the father as its vertex and Hyperbolic tree [6], which visualize the web structure by means of a hyperbolic space. These representations are promising in showing the web structure, nevertheless, none of them can be exempt from problems such as occlusion or scaling for a large amount of nodes.

An interesting related contribution is the representation of Parallel Coordinates [4] to visualize data in high-dimensional space and model relations between data variables by transforming the search for relations among the variables into a 2-D pattern recognition problem [5]. In that system, lines linking high-dimensional coordinates form a kind of polygon. The major difference with our work is that these lines, in the Parallel Coordinates, link parallel axes representing the dimensions. Regarding the visualization objects (cf. interaction with their use), our polygon graphs are formed by linking at different levels of hyperlinks originating from a same page. Polygon graphs may look similar to radar plots (or spider plots). However, radar plots are histograms that have been bent into circles with each individual spoke representing a variable. They are essentially used to visually and approximately compare polar histograms (i.e. compare a number of results to some set target without regard to a specific numeric scale). The polygon represents values for a set of variables. Our polygon graphs instead visualize the value of a same variable for a set of children pages linked from a given page. They combine variable values with web structure (i.e. context). Our polygon represents the values of one variable for different repeated entities (i.e. web objects).

Here we present a data visualization technique to discover web navigational patterns, which can be applied on most of the tree representation of web structure. A unique feature of our work is the inclusion of dynamic interactive

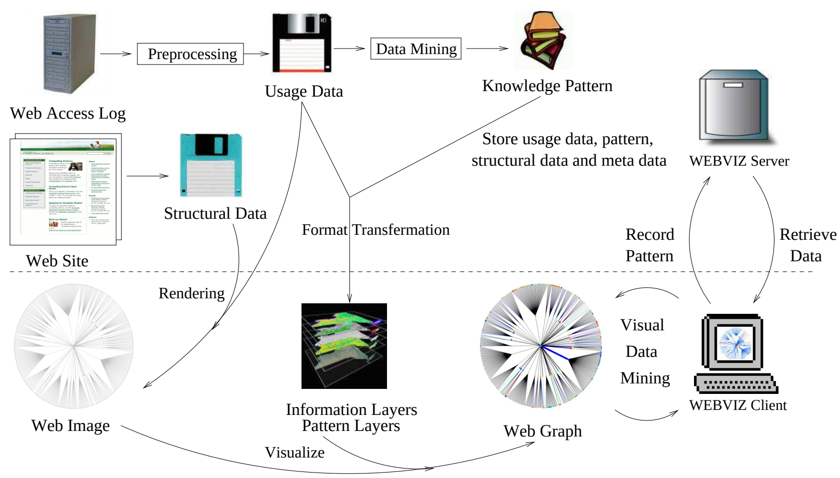


Figure 2: WebViz System Architecture

visualization methods, where visual operations can be used to improvise during the process of visual data exploration and mining.

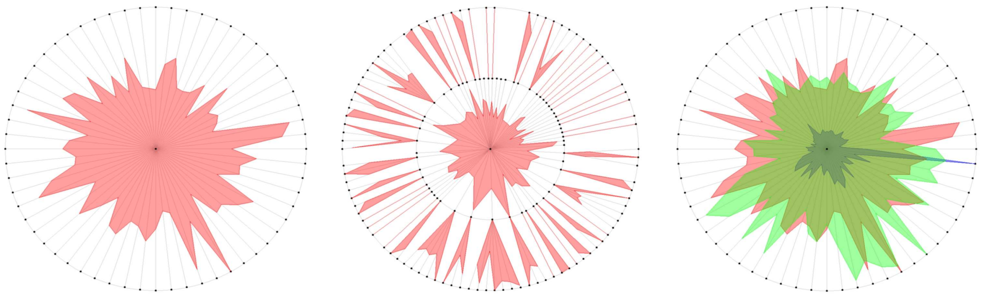
3 Visualizing Web Navigational Data

In order to visualize the usage data extracted from web access logs and to interactively discover navigational patterns in an ad-hoc manner, a prototype framework WebKVDS was proposed in [1]. WebViz, our new system that also adapts the radial tree algorithm to visualize the web structure and usage data, is based on WebKVDS but implements more functionalities, in particular, polygon graphs that we present herein.

We refer to our adaptation of the disktree visualization as a “web image,” which is displayed as the background of our visualization graph. The application of our WebViz system is based on web access logs. We extract usage data and use our system to find knowledge patterns, which are categorized by their attributes. These knowledge patterns are refereed to as “information layers” since they can be represented by different visual cues that can be visually overlaid on the base web image. These layers are extracted from web access logs as usage statistics or patterns, e.g., page visit count. By combining interesting layers with the web image, we visualize web usage data together with their structure context and create a “web graph” as a result.

Figure 1 shows a screenshot of the WebViz system. In that example, each radial tree is a visualization of the web structure and usage of our department website based on access log data in one month. As a default, we use node size to represent the page visit count, node colour to show the average viewing time, edge thickness to indicate the link usage count and edge colour to represent the usage percentage of this link compared to other links that have the same head page. All these legends can be edited: users are able to change the cue-layer mapping, add, insert or change the value range, colour, size of the visual cue setting and save their preference in a meta data file. By allowing the user to change the root and depth of the radial tree, WebViz can be used to visualize the whole web site picture, as well as to depict the detail of small clusters of web pages. In brief, as a visual web data mining system, WebViz implements most of the interactive exploration methods that existing web visualization tools provide, including focus+context, zooming, rotation, visual cue re-mapping and edit, object filtering based on data and object search by name. An online demo version can be found at [12].

WebViz contains two major components: the WebViz server and the WebViz client, as shown in Figure 2. On the server side, We construct the structure of the given website using web crawling tools, and then extract usage data by preprocessing provided web access logs. That prepared usage data is used as input to a variety of data mining algorithms to create patterns, e.g., association rules. The base visualization is the web image, which is the visualization for web structure, and is rendered on the client screen after receiving data from the server. The user can select from a variety of overlay options, to select and display usage data, and extract patterns with learning methods. The resulting knowledge patterns are transformed into information layers, which can be overlaid on the web image to generate a web graph. WebViz uses a client-server architecture so that all source data, including web structural and usage data, discovered knowledge patterns and meta data, can be stored on the server side. Users are able to adjust their personal interface and further save their preference in meta files and record new found patterns as data source files on the server for access by other clients.



(a) (b) (c)

Figure 3: Polygon Graph Examples

4 Visual Exploration using Polygon Graphs

In this section, we first describe our method for visualizing web usage data or navigational patterns as polygon graphs on a radial tree, then illustrate examples of how interesting facts of web usage emerge by using polygon graphs in WebViz. Based on these examples, we claim that our polygon graph method helps users understand the web navigational data, and their structured context with a radial tree. It is thus a new useful component of the general web data visualization system.

4.1 Visualizing Usage Data as Polygons

In our radial tree representation, the screen space is used more efficiently than many other structure layout methods. This is not only because of the disktree algorithm, which distributes object location based on their level and usage count [1], but also because the system can combine many visual cues that can be used to represent data attributes, including the colour, size of nodes and the colour, and thickness of edges, so multiple attributes can be visualized at the same time. However, the information that is attached to edges as their thickness or colour are often too hard to recognize in deep trees, and the space along edges between parent node and its children is usually wasted in single or double level trees. (see Figure 1). In order to better use the available visual space, we propose the idea of a polygon graph as a visual cue. It can be used to visualize any usage data on top of the tree structure visualization, and is easy to manipulate interactively to help understand navigational pattern as well as compare differences between multiple data attributes. Given a type of data attribute to visualize, and a scale formula (explained below), there are two steps to build a polygon graph based on the disktree visualization.

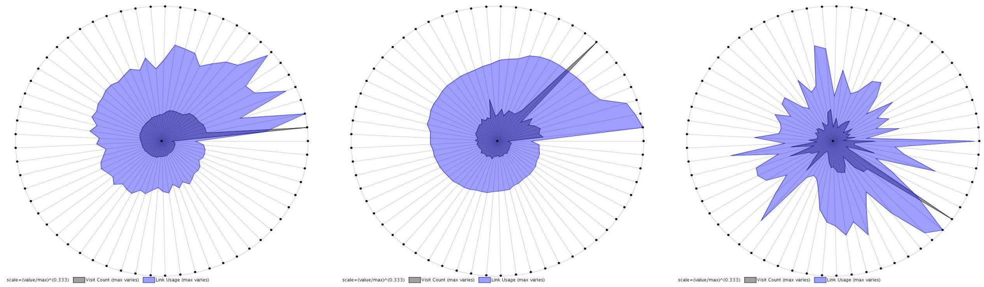
- Step 1: For each node (except the root), find a representative point along the edge that connects the node and its parent, based on the scale formula, the data value of the node, and the maximum data value of the nodes that have the same parent.
- Step 2: For each parent node, connect all representative point of its children, which creates a closed polygon around the parent node.

Figure 3 (a) shows a polygon layer on a single level radial tree and (b) displays a polygon layer on a multiple level disk tree. For subsequent levels of multiple level trees, a parent node is connected to its leftmost and rightmost children to form fan like polygons. Figure 3 (c) shows a polygon graph with multiple layers. The layers are represented in different colours and the possible relations of corresponding data attributes can be found by comparing different polygon shapes.

In polygon graph generation, the scale formula is used to control locations of representative points. Since the possible location of a representative point is from the parent node to the child node along the edge, given the maximum data value of children and data value of the current child, the scale formula serves as a function that takes the above two values as input and generates a floating number between 0 and 1 as output. We can then compute the position of the representative point by multiplying the result value by the length of the edge. However, the usage data associated with children nodes are usually very different (e.g., view time, visit count, etc.) so the maximum values can be highly variable. Therefore our scale factor provides a simple kind of spatial normalization, so that we constrain the polygon to be within the space of the base tree visualization. Otherwise, if we use simple scale function, e.g., percentage, representative points would be all very close to the parent and the generated polygon would be too small to see. Thus, the scale formula we used is set to $scale = \sqrt[3]{\frac{value}{max}}$ to attenuate the effect of the maximum value that could be an outlier.

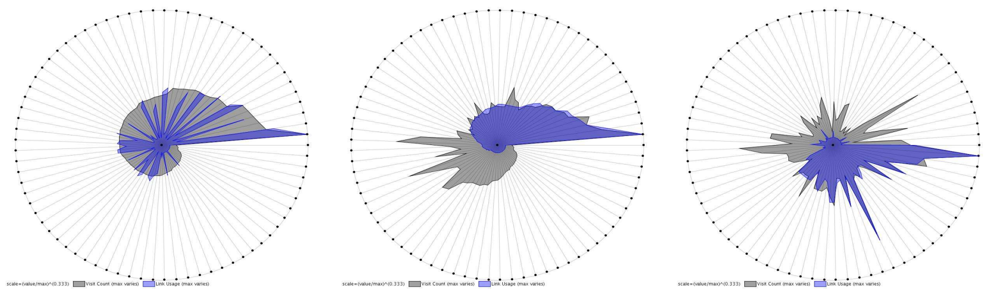
4.2 Interactive Analysis on Polygon Graphs

In order to provide an interactive environment to manipulate polygon graphs, we designed and implemented several visual data exploration operations for the user to discover knowledge patterns that the polygons may contain:



(a) Order by Visit Count (b) Order by Link Usage (c) Random Order

Figure 4: Similar Data Attributes in Polygon Graph



(a) Order by Visit Count (b) Order by Link Usage (c) Random Order

Figure 5: Dissimilar Data Attributes in Polygon Graph

- *Node reorder*. The order of the children nodes in a disktree is originally decided by the node scan sequence of the breadth search algorithm. We provide an operation that can order the children nodes that have the same parent by a selected data attribute and visualize nodes clockwise starting from 3 o'clock ascendingly. This operation reveals relations between different attributes and generates more understandable figures. The user can decide to order the whole tree or just a subtree by an attribute.

- *Change polygon graph root*. Double clicking on any node makes it the root of the disktree. This operation allows users to navigate the visualized web structure.

The user can also pick nodes to compare their detail information, modify colour mapping, select certain layers to visualize and view the whole visual data exploration process using the "back" and "forward" functions, which is similar to a web browser. In the following, we show examples of finding web navigational patterns using the tree reordering operation. Other functionalities can be tested in our online demo [12].

Two of the most important data attributes in web access logs are visit count and link usage, which record the frequency of visits to a page and the frequency a hyperlink is followed, respectively. Web visualization systems usually use node size to represent visit count and edge thickness to show link usage. However, this straightforward visualization method does not provide the ability to compare and analyze the relationship between these two attributes. In a radial tree context, the link usage of an edge connecting the child node and its parent node represents the visit frequency to the child from its parent, while the visit count of the child shows the overall access frequency. In other words, the link usage value actually represents the visit count attribute "locally." Therefore, by comparing the two attributes, we can discover whether the "local" navigational behaviour is similar to the overall visit pattern. For example, a page is considered normal if the local link usage distribution to its children is similar to that of the overall visit count frequency and is unusual and need more attention otherwise.

While traditional web data visualization methods can not easily compare the two attributes that may only look similar to each other proportionally, polygon graphs provide an interactive and effective way to analyze them. In Figure 4, the root of the radial trees represents a page whose local link usage pattern is similar to the global visit count frequency. Children nodes of the radial tree in Figure 4(a) are ordered by the visit count attribute and 4(b) is ordered by the link usage attribute. We can see that a child page usually has high "local" access frequency if it is visited more often overall, and vice versa. The phenomena can also be shown by the shape similarity of the two polygons representing two attributes on the random ordered radial tree in Figure 4(c). On the other hand, Figure 5 shows the characteristic of a page that need further investigation. The polygons are totally different whether ordered randomly or by selected attributes. The figures clearly show that the local navigational pattern of this page does not match the global access frequency pattern, in other words, people who view the page tends to visit pages that are not popular overall more often than those "hot" pages. Therefore, by visualizing and comparing navigational patterns in the polygon graph, our system can help web administrator to quickly locate possible places that need careful monitoring and design improvement.

5 Conclusions

It is undeniably advantageous for web designers and web application administrators to understand the trends and patterns in web navigation. The implicit relationship between the global and local visit frequency and other data attributes is an example of such useful web patterns. Visualizing these patterns in the web structure context further provides a means to explain the discovered knowledge in an interactive manner to make it easier to understand.

In this paper, we introduce our web data visualization system and elaborate on a new method that creates a new general visualization object that we call polygon graph. Our system displays web usage data in context and supports not only visualize discovered patterns but also improvised interactive manipulation, and promotes impromptu visual data mining. The idea of polygon graphs can be applied on any tree structure visualization and already shows its advantages to visualize and compare attribute similarity.

Our continuing work includes more evaluation of our methods across domains, further development of interaction techniques with the visualization objects, and the development of new visualizations as appropriate.

Acknowledgments

Our work is supported by the Canadian Natural Sciences and Engineering Research Council (NSERC), by the Alberta Ingenuity Centre for Machine Learning (AICML), and by the Alberta Informatics Circle of Research Excellence (iCORE). We would also like to thank Eric Coulthard and Jeffery Grajkowski for their remarkable contributions.

References

[1] J. Chen, L. Sun, O. R. Zaïane, and R. Goebel. Visualizing and discovering web navigational patterns. In *Seventh ACM SIGMOD International Workshop on the Web and Databases (WebDB 2004)*, pages 13–18, June 2004.

[2] Ed H. Chi. Improving web usability through visualization. *IEEE Internet Computing*, 6(2):64–71, March/April 2002.

[3] Ed H. Chi, James Pitkow, Jock Mackinlay, Peter Pirolli, Rich Gossweiler, and Stuart K. Card. Visualizing the evolution of web ecologies. In *Proceeding of CHI'98*, 1998.

[4] A. Inselberg. n-dimensional graphics, part i–lines and hyperplanes. In *Technical Report G320-2711, IBM Los Angeles Scientific Center*, 1981.

[5] A. Inselberg. Visual data mining with parallel coordinates. *Computational Statistics*, 13(1), April.28 1998.

[6] J. Lamping, R.Rao, and P.Pirolli. A focus+context technique based on hyperbolic gemetry for visualizing large hierarchies. In *ACM SIGCHI*, pages 401–408, May 1995.

[7] Nelson Minar and Judith Donath. Visualizing the crowds at a web site. In *Proceedings of CHI99*, 1999.

[8] T. Munzner. H3: Laying out large directed graphs in 3d hyperbolic space. In *1997 IEEE Symposium on Information Visualization*, pages 2–10, 1997.

[9] Y. Niu, T. Zheng, J. Chen, and R. Goebel. Webkiv: Visualizing structure and navigation for web mining applications. In *Proceedings of IEEE Web Intelligence Conference (WIC), Halifax, Canada*, Oct 13-17 2003.

[10] C Oosthuizen, J Wesson, and C Cilliers. Visual web mining of organizational web sites. In *IV '06*, pages 395–401, 2006.

[11] G. Robertson, J. Mackinlay, and S. Card. Cone trees: Animated 3d visualizations of hierarchical information. In *Proc. of ACM SIGCHI conference on Human Factors in Computing Systems '91*, pages 189–194.

[12] WebViz: http://kingman.cs.ualberta.ca/research/demos/content/webviz/demo/src/front/WebVizClientApplet.html.

[13] Amir Youssefi, D.Duke, M.Zaki, and E.Glinert. Toward visual web mining. In *Proceeding of Visual Data Mining at IEEE Intl Conference on Data Mining (ICDM), Florida*, 2003.

[14] K. Zhao, B. Liu, T. M. Tirpak, and W. Xiao. A visual data mining framework for convenient identification of useful knowledge. In *ICDM '05*, pages 530–537.

[15] T. Zheng, Y. Niu, and R. Goebel. Webframe: In pursuit of computationally and cognitively efficient web mining. In *PAKDD*, pages 264–275, 2002.