

InfoVis 2003 Contest Submission: Visualization and Pair Wise Comparison of Trees

Treemap, Radial Tree, and 3D Tree Visualizations

Nihar Sheth
School of Informatics Indiana
University, Bloomington
nisheth@indiana.edu

Katy Börner, Jason Baumgartner & Ketan Mane
School of Library and Information Science
Indiana University, Bloomington
{katy, jlbaumga, kmane} @indiana.edu

Eric Wernert
Computer Science Department
Indiana University, Bloomington
ewernert@cs.indiana.edu

1 Introduction

This paper presents and discusses data analysis and visualization results as part of our InfoVis 2003 contest submission. Two in-house developed approaches – a coupled dual radial tree layout interfaces and a three-dimensional tree viewer (Stewart, Hart et al. 2001) – were employed to compare phylogenetic trees. An in-house developed radial tree layout was applied to visualize tree topology. The treemap algorithm (Bederson, Shneiderman et al. 2002) developed at HCIL, University of Maryland was utilized to visualize tree attributes.

2 Phylogenies

The test data comprised two small binary phylogenetic (evolutionary) trees. The task required to design an interactive tool that supports the alignment of the tree topologies. The trees are unrooted and leaf-labeled. Hence, mapping between leaf nodes is straight forward while the mapping between internal nodes is not obvious. Two solutions are proposed.

The first tool is a 3D tree viewer developed at the Advanced Visualization Laboratory, Indiana University (Stewart, Hart et al. 2001). It visualizes the hierarchical structure of both trees and interconnects matching leaf nodes by straight, color coded lines. Matching sub-trees are color coded as well to support the interactive alignment process, see Figure 1.

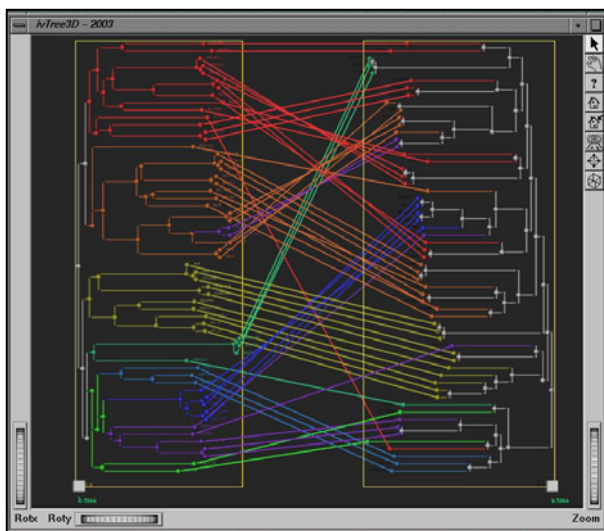


Figure 1: 3D tree viewer showing homology between two phylogenies

Users can search for specific node labels and interactively align trees by changing the layout of trees. Exploiting the third dimension more than two trees can be visualized and compared.

Individual taxa or groups of taxa can be traced across multiple trees. The viewer uses the Open Inventor graphics API to generate the 3D visualization.

The second tool uses two tightly coupled radial tree visualizations to support the semi-automatic alignment of trees. The tool takes a correspondence matrix for two phylogenies as input. This matrix can for example be computed using the popular consensus tree approach (Adams 1972) which establishes mappings of intermediate nodes according to the similarity between their respective leaf sets.

The layout algorithm generates two radial tree visualizations – one for each phylogenetic tree (see Figure 2) – and two control panels (see Figure 3)

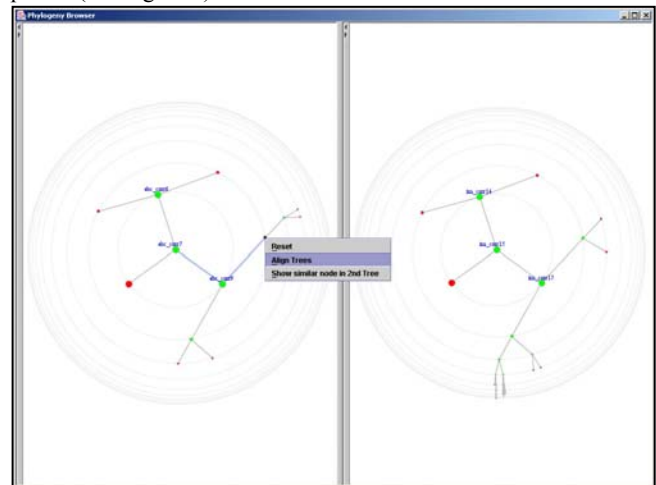


Figure 2: Dual radial tree viewer showing the aligned topologies of two phylogenies

The two radial tree visualizations are tightly coupled. A pop-up menu enables users to automatically align trees or to query the 'other' tree for best matching nodes. Automatic alignment selects the node in the 'other' tree that best matches the currently selected node and moves selected and best matching node into the middle of the display. The search for best matching nodes uses the correspondence matrix to determine nodes that are embedded in a similar topology as the selected node and colors them black. Both tree view have the full radial tree functionality (search, details on demand, etc.) explained in the next section. The program was implemented in Java and runs as applet or application.

3 Classification

This task required to visualize very large trees with large fanouts.

The radial tree visualization introduced in the previous section was applied to visualize the tree structure and to support search and browsing. The radial tree is a focus and context technique developed at Indiana University and is available in the

Information Visualization Repository¹. A query submission interface was added to support search. An interface snapshot showing the “Mammal” sub-tree is given in Figure 3.

Browsing a radial tree is very similar to browsing a hyperbolic tree. Upon selection, a node is moved to the center of the tree and the surrounding tree is rearranged accordingly. A slow-in, slow-out animation technique (instead of a straight linear transition) was used to provide visual constancy and to reduce disorientation.

The layout shown in Figure 3 provides labels for the center and first level nodes exclusively to avoid clutter. Each rank category of classification hierarchy such as phylum, order, family etc. were color coded differently to provide navigational cues to the user. For example, family nodes in Figure 3 are given pink, order nodes are in orange, and species in green.

The path to the root node (Mammal) of the entire data is highlighted in red from the node at the center to aid the user in the navigation of the hierarchy. Node details can be requested on demand.

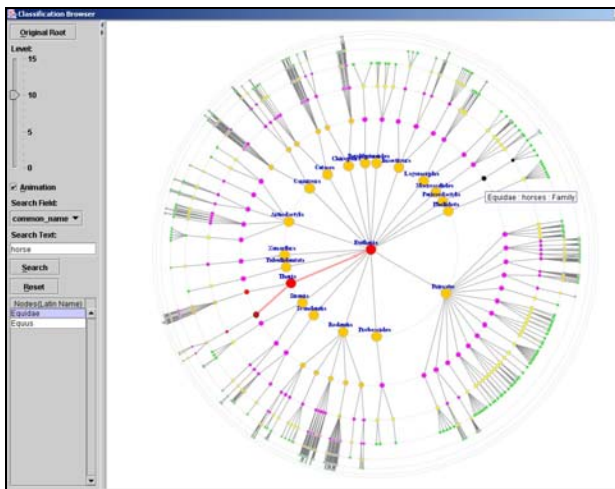


Figure 3: Radial tree Viewer displaying the classification data

The left panel provides a button that can be used to get the original tree layout with the tree root in the center. The slider lets users change the number of displayed levels. Selecting the animation check box leads to a smooth animation of tree layout changes. User can search for Latin or common name of nodes. Search terms are entered in the query field. Regular expression matching is also supported. Matching nodes are marked black providing instant visual feedback to the user. In addition, matching nodes are displayed in a list. Selection of a list item moves this particular node in the center of the display and aligns the tree accordingly.

In sum, browsing results and search results are shown in the context of the particular tree structure.

4 File System and Usage Logs

The third task required to determine and visualize topological or attribute value changes in large trees.

Subsequently we discuss a visualization that aims to visualize the attribute value changes for name, hit counts, title and others provided in the log files.

The treemap algorithm (Bederson, Shneiderman et al. 2002) implemented at the human computer interaction laboratory

(HCIL) at the University of Maryland was identified as the tool that would serve the task requirements. Figure 4 shows the data using the squarified partitioning method. Each directory/file is represented by a square label. Size, color, and label of each square can be used to represent three attributes of this directory/file. For example in the figure 4 below we have the color coding for the creation time of the directories/files. The green coloration is given to older pages, the blue color to entities that are created most recently. This gives us a very good overview of the creation time of the structure at a glance. The nesting of the directory squares corresponds to the original directory hierarchy.

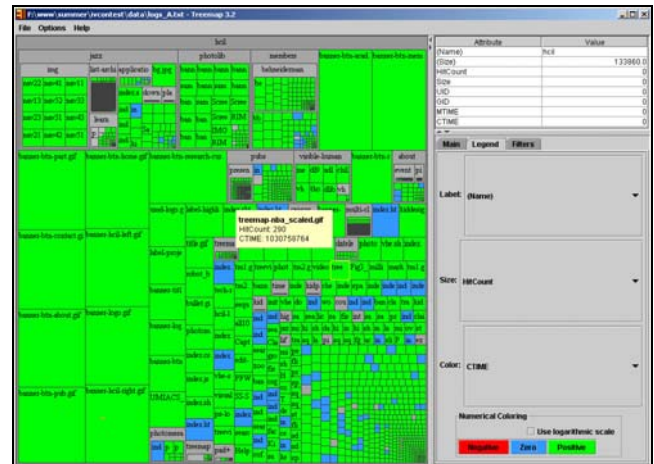


Figure 4: Treemap visualization showing the file system of the Computer Science Department at the University of Maryland

Placing the mouse pointer over a square brings up a panel with additional information such as file name, size, and number of hit count. If the user clicks on a square that represents a directory then the current treemap visualization is replaced by a zoomed in version of the selected square. Multiple combinations of individual attribute value can be visualized. The user can search for directories, do a comparative statistics on hit counts, and can visually identify files or folders with similar attribute values.

A web page accompanying this submission with large scale versions of the interfaces as well as animation sequences showing them in interaction is at:

http://ella.slis.indiana.edu/~kmane/katy/iv_contest/webpage/.

5 References

Adams, E. N. (1972). "Consensus techniques and the comparison of taxonomic trees." *Systematic Zoology* **21**: 390-397.

Bederson, B. B., B. Shneiderman, et al. (2002). "Ordered and Quantum Treemaps: Making Effective Use of 2D Space to Display Hierarchies." *ACM Transactions on Graphics (TOG)* **21**(4): 833-854.

Stewart, C. A., D. Hart, et al. (2001). Parallel implementation and performance of fastDNAmI - a program for maximum likelihood phylogenetic inference. Supercomputing Conference, Denver, CO.

¹ <http://iv.slis.indiana.edu>