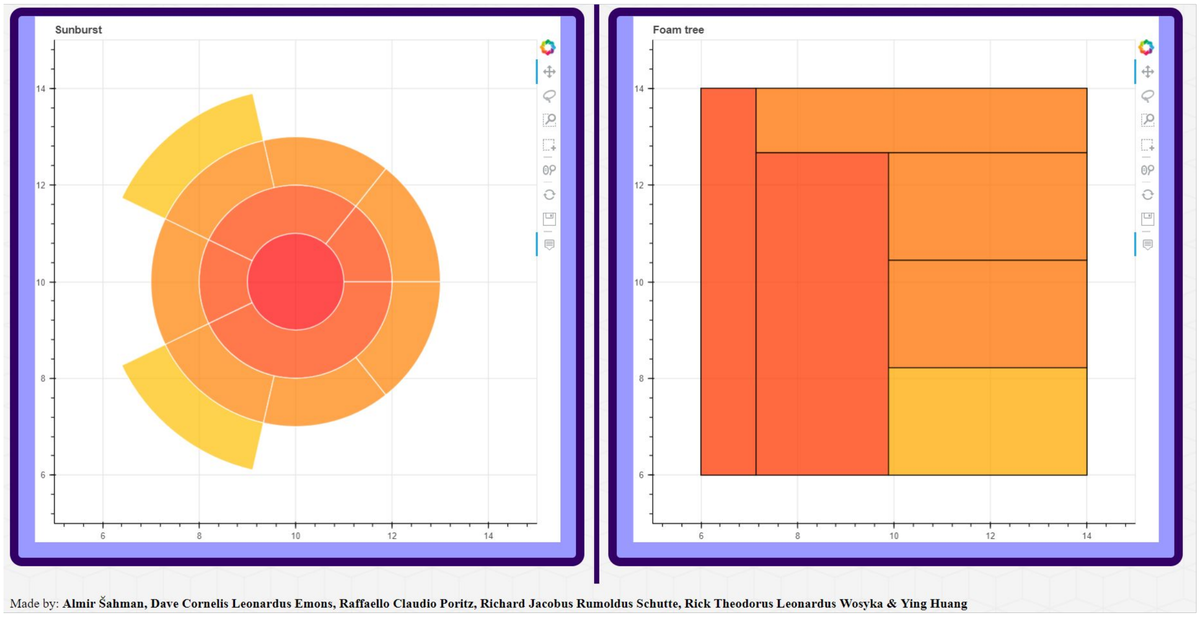
DBL HTI + Webtech: Report

D. C. L. Emons, Y. Huang, R. C. Poritz, A. Šahman, R. J. R. Schutte, and R. T. L. Wosyka



Made by: **Almir Šahman, Dave Cornelis Leonardus Emons, Raffaello Claudio Poritz, Richard Jacobus Rumoldus Schutte, Rick Theodorus Leonardus Wosyka & Ying Huang**

Fig. 1. Sunburt and Foamtree on custom data set

Abstract—In this report we will discuss, describe, and explain our work in creating and applying various hierarchy visualizations, each showcasing different aspects of a given data set. We implemented Sunbursts and FoamTrees to incorporate different views. In order to give data analysts from around the globe access to our tool providing multiple visualizations, we have used Python as our programming language, along with the Bokeh library, to make it web-based, allowing access to anybody anywhere. Furthermore, we provide an example of our visualization applied to our own custom data set.

Index Terms—Hierarchy visualization, Interaction techniques, Multiple coordinated views, Web-based visualization

1 Introduction

In this report, we will describe the two visualization techniques we have chosen, as well as explain what went into making our visualization tool. In order to help give perspective on hierarchical data to whomever may need it, be it data analysts or data enthusiasts, we have created a tool that creates two different visualizations of any hierarchical data set as they, individually, do not provide a full image of the data (pun intended), but together, each makes up for many of the discrepancies of the other. Out of the many different visualizations (over 300 just on `www.treevis.net` alone), we decided to use Sunburts and FoamTrees, as not only to they compliment each other nicely, but they also provide with aesthetically pleasing images.
One of the requirements for out tool was that it be accessible to anybody around the world, for them to simply upload a data set, in the Newick format, which is then parsed, and then have both visualizations created and shown. And so our tool is readily available on the internet, and makes use of the Bokeh library for Python, which facilitates the creation and integration of graphical visualizations on the web. We also used NumPy for the visualization, and the Biopython library for our parsing tool.
We give an example of our tool, used on the *INSERT DATA SET HERE*, + *INSERT QUICK STUFF*.

2 Related Work

Visualization of hierarchical data has been a central problem of information visualization for more than half a century. The first research paper on the matter in the Association for Computing Machinery's (the premier US scholarly society for computer scientists) Digital Archive dates back to the 1950's [5], but visualizations have been around for far longer, such as Johann Christian Lange's "Formal Logic Representation" which dates back to 1714 [2].
Despite there being over 300 different visualizations on `www.treevis.net` alone [6], no single visualization gives a perfect representation of

- *Dave Cornelius Leonardus Emons, E-mail: d.c.l.emons@student.tue.nl*
- *Ying Huang, E-mail: y.huang.2@student.tue.nl*
- *Raffaello Claudio Poritz, E-mail: r.c.poritz@student.tue.nl*
- *Almir Šahman, E-mail: a.sahman@student.tue.nl*
- *Richard Jacobus Rumoldus Schutte, E-mail: r.j.r.schutte@student.tue.nl*
- *Rick Theodorus Leonardus Wosyka, E-mail: r.t.l.wosyka@student.tue.nl*

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

any and all hierarchical data set. For example, the Walker Layout [9] suffers from the same problem as other node-link diagrams: due having the root alone at the top, with the leaves flooding the bottom, leaving them cramped together, not making use of the free space at the top of the visualization, near the root and the first few levels of the tree. To solve the problem of unused space, one could use a Nested Pie Chart [7], but then another problem arrises: comparing the sizes of various shapes is inherently difficult, so estimating whether a certain subtree is larger or smaller than another is not something that can be done quickly and without extra information.
Three-dimensional visualizations have also been created, some as early as the 1960, such as Jacques Bertin's Stereogram [3]. More recently, Mulitvariate Hierarchic Plots, which which closely resembles a three-dimensional version of our chosen Sunburst visualization, was created to help ”analysts wish to interpret the structure of the data not only at a single point in time, but examine the changes in the data categories through time” [8]. To that end, another, third dimension was added, to represent the changes of the data through time. Unfortunately, while this visualization technique still provides a lot of information to the data analyst, it suffers from the same problem of comparing region sizes as all other radial representations.

3 Data Model

In general terms, data hierarchy gives relations between groups, subgroups, and individual elements of a data set. The data has a tree-like form to it, with every element other than the root (in tree terms) having exactly one parent element, not more and not less, but every element can have any number of children elements, be it zero, one, or twelve.
The Newick format (also known as the Newick Standard, or New Hampshire Tree Format) is a way of representing trees in a computer-friendly fashion with parentheses and commas. It uses relation parentheses and trees described by Arthur Cayley in 1857 [4]. Nodes that are children of the same parent node appear within the same set of parentheses, and nodes that have the same depth are incased in the same number of parentheses, although they are not necessarily the same parentheses. In other words, nodes that result from the same split of a branch appear in the same pair of closed parentheses, separated by commas.
There are two main limitations of the New Hampshire Standard. The first is the uniqueness, or lack thereof, of trees. In some fields, the order of descendants does not matter, while the format does take into account the order of inputed nodes. So for example, the tree $(A_1,(B_1,B_2),A_2)$; and $(A_2,(B_2,B_1),A_1)$; represent different trees, but to analysts in fields that do not care about left or right children of nodes, the two trees are the same.
The second limitation is that the format represents rooted trees. If the root of a tree is irrelevant, or simply not inferable, the Newick Standard does will simply arbitrarily root the tree. For example, an example given by Joe Felsenstein [4], one of the creators of the format, is the trees $(B,(A,D),C)$;, $(A,(B,C),D)$, and $((A,D),(B,C))$ represent the same unrooted trees.

4 The Visualization Tool

Our tool creates a multi-view representation of the uploaded dataset. We chose to use Sunburts and FoamTrees.

4.1 Graphical User Interface

- Show here the GUI of your tool. Make a screenshot from a browser and show how the hierarchy visualizations look like. Just an example screenshot...
- Explain all the features of the GUI in detail
- What are the different views of the GUI?
- Are they already linked?

4.2 Visualization Techniques

- Explain your visualization techniques here
- You can use the literature from www.treevis.net to get all the information you need
- Provide lots of screenshots from your visualization techniques here to illustrate them

For our tool, we implemented both Sunburts and FoamTrees. Whilst

4.3 Implementation Details

We chose to use Python as our programming language for ease of use. Given that we all took Data Analytics for Engineers last quartile, in which we learned and used Python, as well as the NumPy and Pandas library, we thought it would be best to work in a programming language we were all familiar with, and so we could all work in tandem on the project, without having any issues of swapping back and for between languages for various parts of the tool and website. Coupled with the fact that Python was one of the recommended languages, with resources such as Bokeh mentioned on the class site on Canvas, it was the natural choice.
The data is uploaded to the website as a text file of any type, with the data in the Newick format. We then used the Biopython library to parse given dataset. The Bokeh library was used in making both the visualization and the website, as ”its goal is to provide elegant, concise construction of versatile graphics, and to extend this capability with high-performance interactivity over very large of streaming datasets” [1]. The visualization also made use of the NumPy library for Python.

5 Application Example

- Show one dataset example, maybe the NCBI taxonomy in a browser
- Explain what you see
- Are there any visible structures?
- Are there any outliers or anomalies?

6 Discussion and Limitations

- What are the limitations of your approach?
- Is there a visual scalability issues like how many elements can be displayed on screen?
- Is there an algorithmic limitations, for example, for computing the hierarchy visualization and the layout?

7 Conclusion and Future Work

This Design-Based Learning course and project taught us a lot. We were given what seemed to be at first a large and unsurmountable task, which then when divided into smaller individual pieces with the help of our tutor and scrum master, no longer seemed to be a daunting and impossible assignment. We learned how to divide large wishes for our project into smaller, indepdenant tasks, so that any one member of our group could work on something at any time, whilst not having to rely on the work of others to complete their own. Through division of labour, the small subtasks came together like the different pieces of a puzzle, working in unison to work toward the final goal.
We still have plans to improve our tool even further. We want to add further interactions with the visualizations, as well as maybe adding a new visualization to our tool altogether.

Acknowledgments

We would like to thank our group members for their support in this DBL project. Moreover, we are thankful to our tutor who gave us a lot of feedback to find the right decisions in this project.

References

[1] Bokeh library. `https://bokeh.pydata.org/en/latest/`. Accessed: 23-05-2018.

[2] M. E. Baron. A note on the historical development of logic diagrams: Leibniz, Euler and Venn. *The Mathematical Gazette*, 53(384):113–125, 1969. doi: 10.2307/3614533

[3] J. Bertin. *Semiologie graphique. Les diagrammes, les reseaux, les cartes*. Editions Gauthier-Villars, 1967.

[4] J. Felsenstein. `http://evolution.genetics.washington.edu/phylip/newicktree.html`. Accessed: 13-05-2018.

[5] A. for Computing Machinery Digital Library. `https://dl.acm.org`. Accessed: 21-05-2018.

[6] H. J. Schulz. Treevis.net: A tree visualization reference. *IEEE Computer Graphics and Applications*, 31(6):11–15, Nov 2011. doi: 10.1109/MCG.2011.103

[7] A. Sukla and M. Sauhta. Nested pie chart, 2005. retrieved 26-07-2011.

[8] T. Tekusova and T. Schreck. Visualizing time-dependent data in multivariate hierarchic plots – design and evaluation of an economic application. In E. Banissi, L. Stuart, M. Jern, G. Andrienko, F. T. Marchese, N. Memon, R. Alhajj, T. G. Wyeld, R. A. Burkhard, G. Grinstein, D. Groth, A. Ursyn, C. Maple, A. Faiola, and B. Craft, eds., *IV'08: Proceedings of the International Conference on Information Visualisation*, pp. 143–150. IEEE Computer Society, 2008. doi: 10.1109/IV.2008.51

[9] J. Q. Walker, II. A node-positioning algorithm for general trees. *Software – Practice and Experience*, 20(7):685–705, 1990. doi: 10.1002/spe.4380200705