TreeVersity: A Tree Structures Comparison Framework on Topology and Node's Attributes Differences

John Alexis Guerra Gómez
Human Computer Interaction Lab
University of Maryland at College Park *

ABSTRACT

Classifying data in hierarchies is a common technique, they provide a comprehensible way of understanding big amounts of data. From budgets, to organizational charts or even the stock market, trees are everywhere and people find them easy to use. However when we need to compare two versions of the same tree structure, or two related taxonomies, the task is not so easy. Much work has been done on this topic, but almost all of it has been restricted to either compare the trees by topology, or by the node attribute values. With this project I'm proposing TreeVersity a framework for comparing tree structures, both by structural changes and by differences in the node attributes. I'm basing this research on my previous work on comparing traffic agencies using LifeFlow [1, 2] and on a first prototype of TreeVersity, and I'm proposing to develop a generalizable framework to compare common tree structures like different years of the USA Federal Budget.

Index Terms: E.1 [Data Structures]: Trees—; H.5.2 [User Interfaces]: Graphical User Interfaces (GUI)—

1 INTRODUCTION

Hierarchies help us to understand and categorize information, by describing responsibilities, aggregations, precedence or any other father-to-child relationship. Examples like the US Federal Budget, the closing prices of a Stock Market, the tree of species, or even one season's NBA player statistics, can be viewed and organized as a hierarchy. Much research has been done to help us understand, visualize and navigate tree structures. Techniques like node link representations, TreeMaps[3], Radial representations [4]and even Icicle trees [5] are commonly used even in non scientific publications like newspapers and websites.

Once one understands one single hierarchy, the next question would be how to compare multiple trees. Examples of this type of problem are tasks like understanding what changes in a budget proposal compared to previous years, which species are grouped differently on two different taxonomies or which stock's prices have significantly changed on the market. As will be explained in section 2, this type of question has led to a rich group of research projects, that most commonly have been narrowed for specific domains. Because of this they either focus on finding topological differences (e.g. nodes that have been deleted, created or relocated), or comparing node attribute values (e.g. finding budget cuts between years).

With this work I'm proposing a tree comparison framework that addresses a richer set of problems, including: Positive and negative changes in leaf node's attribute values, with and without changes in topology, and positive and negative changes in leaves and interior node's attribute values, with and without changes in topology.

*jguerrag@cs.umd.edu

2 RELATED WORK

This section focuses on research that has been done on comparing, visualizing and analyzing multiple tree structures. There is substantial work on single tree structures, but since they are not relevant to the objective of comparison, it won't be described in this document..

I have categorized the related work in three areas, according to the project's focus. First I will describe the projects that emphasize comparing hierarchies by their topological differences, then the projects that focus on comparing node attribute values, and finally those that have a bigger theoretical component.

2.1 Topological Comparison

Most of the work on tree comparison has been done on comparing topological changes between tree structures. This might have been influenced by the well known problem of comparing taxonomies of species. TreeJuxtaposer by Munzer et al.[6] is one of the best examples of this field, they presented an efficient algorithm to comparing hierarchies. It uses a node link representation with side by side comparison and a focus+context technique with guaranteed visibility. TreeJuxtaposer escalates very well with the number of nodes on the tree, but not so much with the number of trees. A different approach is the one presented by Graham et al. [7] which uses an Icicle[5] like representation with interconnected small multiples for each hierarchy that prove to escalate better on the number of trees, but that still required splitting the screen space among the different hierarchies. In later work [8] Graham address this issue switching from the small multiples to an aggregated representation using directed acyclic graphs (DAG). Others have used before the concept of aggregation of multiple trees in one view, starting by Furnas et al. [9] that proposed the concept on 1994 and CandidTree[10] that uses a node link representation that uses color, shapes and dotted lines to represent uncertainty. Also on the topic of taxonomies, but using a different approach, Amenta and Klingner's TreeSet [11] allows the comparison of a big number of taxonomies by calculating a bi dimensional metric representing each tree and plotting them in a scatter plot.

The InfoVis2003 contest [12] represents a peak point in the development of this field. That year the competition focused on tree comparison, more specifically in finding topological differences between trees. Some of the winning submissions presented innovative solutions for the problem, like TreeJuxtaposer [6] already described. Others include Zoomology [13] which used radial representations combined with zooming interfaces, InfoZoom [14] which used condensed side by side tables, EVAT[15] with radial side by side comparisons and TaxoNote with condensed Windows Explorer like representation. Sadly many of these projects didn't evolve to anything beyond the competition's 2 pages submission.

Finally there have been some other approaches using zooming interfaces like MoireTrees [16] that allows navigation of multi hierarchies (different trees that categorize a share group of leaf nodes) using zooming and radial displays, and DoubleTree [17] that uses two connected, side by side SpaceTrees [18] to highlight topological differences between taxonomies.

Despite the substantial work on topological differences between trees, to the best of my knowledge, none of these solutions address the problem of comparing changes in node attribute values. I propose to build upon this knowledge on topological differences and add more information to tackle problems that go beyond finding moving nodes in hierarchies.

2.2 Node Attribute Values Comparison

The work on comparing node attribute values isn't as rich as comparing tree topologies. The only projects I have found that work on this topic are mainly based on TreeMaps. The original TreeMap tool [3]had a feature that allowed the display of changing values on the hierarchy but it was limited and wasn't planned to be used on comparison. The project Animated TreeMaps [19] explored the concept of representing changes in the nodes attribute values using animation, especially focusing in stabilizing the layout. On more recent work, SmartMoney's Map of the Market [20] represents stock market prices changes using colored TreeMaps. Finally the project Contrast TreeMaps [21]modified the base visualization and used color to represent changes in nodes, as well as structural differences, however their work was restricted to leaf nodes only.

2.3 Theory oriented

The final approach for tree comparison makes use of tree metrics, which usually are algorithms that calculate distances between two or more trees. This metrics can be classified by the type of comparison they make, and Bille [22] presents an excellent survey of them. According to him the most important classes of metrics are Edit Distance, Alignment Distance and Inclusion (a.k.a. subtrees). In his work he describes efficient algorithms for each of this areas that could be used to compare many trees at once. Part of this work is to evaluate the possibility of including some of this in TreeVersity.

Another common related strategy for analyzing multiple trees is the consensus tree [11, 23, 24, 25]. This a technique used in Phylogenetic analysis for summarizing many trees into one. I'm planning on using this to some extent to analyze multiple trees and generate one aggregated version that contains historical information on each node.

3 Current Work

3.1 LifeFlow comparison

My initial work on comparison and original inspiration for this research topic comes from our work on LifeFlow [1, 2]. LifeFlow is a visual analytics tools for temporal categorical data. It allows the understanding of large datasets by the creation of a summary of all the possible temporal sequences present on the data, and represent them using a modified Icicle tree, that can display both the temporal component and the number of records on each sequence. As an example of the type of analysis that can be performed on LifeFlow, it has been used to analyze all the different patterns that patients follow in a emergency unit in a hospital. This information includes hundreds of patients and several temporal events per patient describing their movement through the different areas in the hospital, like the Intensive Care Unit (ICU) and the Floor (normal care unit). One of the questions the doctors were able to answer using LifeFlow, was how many patients "bounce back" to the ICU, this is a sequence of ICU->Floor-> ICU.

My contribution to the project was the incorporation of non temporal attributes to the visualization, that allows the categorization of the data by different parameters. After this I developed new algorithms for sorting the different trees, according to the average or maximum time between events in them. Using these two improvements I was able to compare 8 different traffic agencies, in their efficiency of clearing traffic accidents. I used a dataset from the National Cooperative Highway Research Program (NCHRP) that includes 203,214 traffic incidents from 8 agencies. I were able to

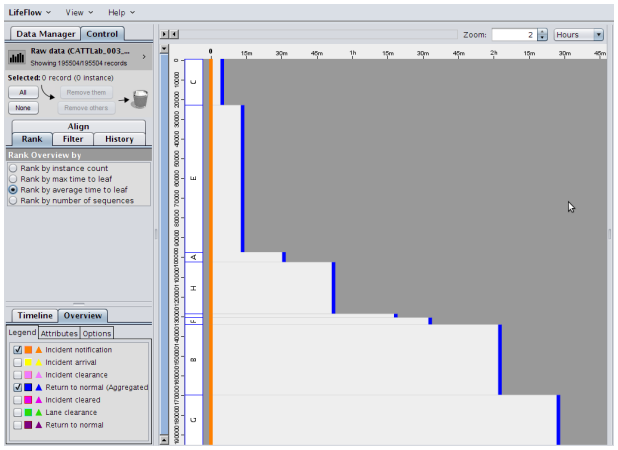

Figure 1: Comparison of 8 different traffic agencies on their performance on clearing 203,214 traffic incidents across the USA.

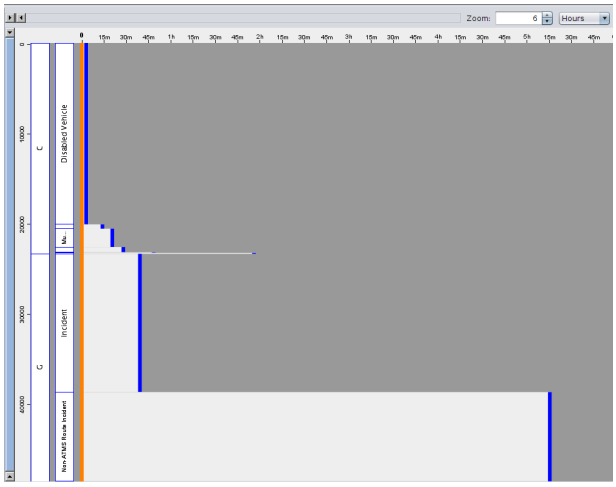

Figure 2: Detailed comparison of agencies C and G. The LifeFlow shows that most of the accidents attended by C are reported cleared right after they were notified which can explain the good overall result obtained by the agency, while in agency G the type "Non-ATMS Route Incident" is taking more than 5 hours to be cleared, which is affecting the agency average.

rank the agencies by the data that I had, and I found some possible causes for the best and worst performers. Figure 1shows an screen capture of the LifeFlow analysis of the dataset, on it the orange bar represents when the agencies were notified and the blue one when they cleared the scene. The x axis represents the time and the y axis the number of incidents attended by each agency. From the image is easy to see that agency C is the fastest agency, clearing the accidents in less than 10 minutes, while agency G is taking more than 2 hours in average to resolve their incidents. Further analysis of the type of incidents cleared by both agencies revealed that agency C had such a good average because they were most of their incidents as cleared at the exact same time they were reported (which might indicate a data entry problem), while for agency G they reported only two types of accidents, and while they were performing very well in one of them, they were taking more than 5 hours to clear the second type, as can be seen on Figure 2.

3.2 TreeVersity

The type of comparison capabilities of LifeFlow was useful for the task at hand, but at the same time it reveals some limitations of the technique. For example comparing more than two events at a time is very difficult in LifeFlow because of screen overcrowding. Because of this I decided to begin building TreeVersity, a tool that specializes in comparing tree structures, both on topological changes and in node attribute values.

I am designing TreeVersity to support the following use cases: 1) Given two trees, compare them and explore the differences by user interaction. 2) Same as the previous case, but provide a ranking of the most significant differences and similarities, and guide the user through them. 3) Given a forest (a group of trees) and one selected tree as a base, find the most similar (and the most dissimilar) hierarchies, and display their similarities (and differences). 4) Given a forest, create an average tree that represents the most common characteristics among the trees. 5) Given a forest, find the most different trees, that is, trees that are the most different to the average.

Of those use cases, I have started to implement the first one, interactively comparing two tree structures. My main goal is to create an interactive visualization that allows the comparison of two trees by looking at: 1) Created and removed nodes. 2) Absolute and relative differences of the node attributes values. 3) Cardinality of the differences. 4) Differences in attributes of leaf nodes only or differences in attributes of interior nodes also. 5) Amount of change compared with the other nodes on the tree (or compared with the siblings).

For this I have implemented a prototype of TreeVersity that uses a mixed approach for comparison. First it presents a connected side by side comparison of the trees, that allows synchronized navigation and identification of unique versus created/removed nodes. Second I display an aggregation of both trees that represents the differences between the node attribute values of the trees, I call this the *diffTree*. For this I have been experimenting with two different visualizations, I have called them informally the "slope" and the "gas tank" approaches. In the next sections I describe them in more detail.

3.2.1 The Slope

The slope is a tree visualization based on a node link representation. It uses shape, color and size to represent the (absolute or relative) amount and direction of change. The shape presents a slope that shows which node was bigger, then one of the tree of the left (decreasing slope) or the one on the tree of the right (increasing slope). The amount of slope is relative to the amount of change of each node compare with all the other nodes on the tree. The color represents the same amount of change using a gradient of tree user selectable colors.

The topological differences are also represented in this visualization. I used an special marker (a different color and a line) on the nodes to differentiate created and removed nodes from the others. An example of this visualization is shown on Figure 3.

I believe the slope visualization is good to recognize the changes in all the levels of the tree, to identify the topological changes and to understand the structure of the aggregated tree. However it works better with smaller trees, and can only represent one magnitude of change at a time (either absolute or relative). Because of this I developed a complementary visualization that is presented in the next section.

3.2.2 The Gas Tank

The "gas tank" representation uses an space filling approach based on TreeMaps [3] to represent changes in the leaf nodes of the diffTree, displaying at the same time the absolute and relative amounts of change. It is specially good to highlight the "biggest players" on the diffTree (nodes with the biggest values overall).

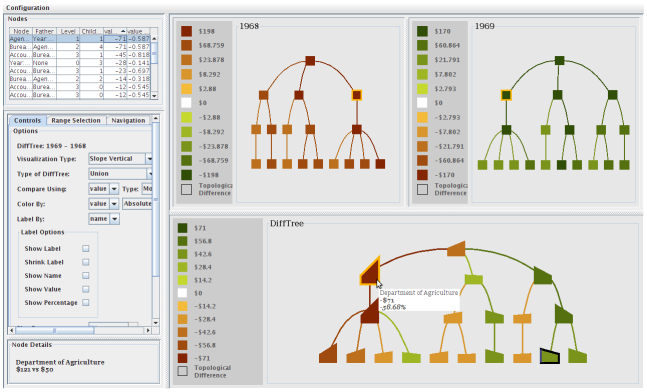

Figure 3: Example of the slope visualization representing a subset of a made up Federal Budget. The image shows the comparison of years 1968 vs 1969, where a cut of 58% was made on the "Department of Agriculture". Red nodes represent cuts, while green ones represent increases. The node with the black border represents a created node (topological difference).

Figure 4: Gas Tank Visualization representing the data from Figure 3. It shows the absolute amount of change using color and a label, and the relative difference using the amount of space filled in each node. It also shows the topological differences using again a black border. This view also allows the selection of a node along all the compared trees and the diffTree to see its details, like it was done with the "Receipts, Federal litigative and judiciary" that decreased only on $2 from 1968 to 1969.

To create the gas tank representation I first take the two individual TreeMap nodes representation, I combine them and obtain the difference, and finally I draw the difference as a filling portion of the area of the biggest of the original nodes. This way I avoid nodes of size zero. The process is illustrated on Figure 4 on the left. I then use this node representations of the leafs and aggregate them in the same way TreeMaps does to represent the hierarchy. Although the gas tank visualization only represent the changes on the leaf nodes, TreeVersity has a control that allows the selection of a level of the tree to represent, that way the gas tank diffTree is redrawn to represent only nodes in that level (or leaves on previous levels). An example of the gas tank representation with the same artificial budget data is shown in Figure 4 on the right.

Like the slope visualization, the gas tank also has its strong and weak points. It is specially useful to compare the absolute and relative changes in the biggest nodes, but it isn't so helpful to represent the hierarchy of the tree, and it hides information like the amount of change in the interior nodes.

3.2.3 TreeVersity current limitations

To calculate the node attributes differences, I assume the hierarchies to contain an uniquely identifier for each node, that allow us to make a matching between the trees, and calculate the difference. I have established out of the scope of this project problems like finding matching subtrees or partial matchings. In the same way of thinking I have chosen to work only on unordered trees, to simplify the problem domain.

4 Proposed Plan

I will build on the current work to develop a framework for tree comparison both on topology and in node attribute values. For this I'm proposing the next objectives:

- Design and build a first prototype of TreeVersity, using as a base the slope and gas tank analogies presented, that allows the interactive navigation and comparison of two trees at at time, representing absolute and relative changes in leaf or interior nodes of two trees. For this I am building on top of the proposed work presented on Section 3.2, and I am improving the design with the close collaboration of Audra Buck-Coleman Assistant Professor of Design at the Department of Art of the University of Maryland. **Expected outcome**: TreeVersity tool version 1.0, conference paper (objective CHI2012). **Duration:** Tree months.

- Evaluate the effectiveness of TreeVersity 1.0 in one or more domains. I want to use the tool to analyze and find insights on at least one specific domain, the first candidate I have for this is the USA Federal Budget, comparing it between years. I might also include some other domains, like the transportation data I have worked on before. **Expected Outcome:** Evaluation report. **Duration:** Four months.

- Building on top of TreeVersity 1.0, I plan to include an extension that helps the user navigate through the differences on the trees. For this I want to design, implement and evaluate an step by step reporting tool that guides users through the different changes, ranking them by different criteria that might be dependent on the specific application field. Examples of this might be navigating the changes in the budget by identifying first nodes that have a significant increase, where all of its siblings have cuts. **Expected Outcome:** TreeVersity version 2.0 with step by step changes reporting tool, conference paper. **Duration:** Nine months.

- Finally I want to expand the application of TreeVersity from comparing just two trees to comparing many trees at once. I want to implement some of the other proposed use cases described in Section 3.2, where I analyze the evolution of a tree in more than just two steps. **Expected Outcome:** TreeVersity version 3.0, with one type of comparison of many trees at once, conference paper. **Duration**: Nine months.

5 Acknowledgements

I thank the Fulbright International Science and Technology Scholarship for supporting my dream of making doctoral studies. Ben Shneiderman, Catherine Plaisant and Audra Buck-Coleman for their thoughtful advise. The Center for Integrated Transportation Systems Management (a Tier 1 Transportation Center at the University of Maryland) for partial support of this research. I also thank the Center for Advanced Transportation Technology Laboratory (CATT LAB), Michael Pack, Michael VanDaniker and Tom Jacobs for their suggestions and feedback.

References

[1] J. A. Guerra Gómez, K. Wongsuphasawat, T. D. Wang, M. L. Pack, and C. Plaisant, "Analyzing incident management event sequences with interactive visualization," in *Transportation Research Board 90th Annual Meeting Compendium of Papers*, 2011.

[2] K. Wongsuphasawat, J. A. Gomez, C. Plaisant, T. D. Wang, B. Shneiderman, and M. Taieb-Maimon, "LifeFlow: visualizing an overview of event sequences," in *Proceeding of the twenty-ninth annual SIGCHI conference on Human factors in computing systems. ACM*, 2011.

[3] B. Johnson and B. Shneiderman, "Tree-maps: A space-filling approach to the visualization of hierarchical information structures," in *Proceedings of the IEEE Conference on Visualization (Vis)*, pp. 284 – 291, IEEE, 1991.

[4] D. Fisher, R. Dhamija, and M. Hearst, "Animated exploration of dynamic graphs with radial layout," in *Proceedings of the IEEE Symposium on Information Visualization*, vol. 2001, pp. 43 – 50, IEEE, 2001.

[5] J. B. Kruskal and J. M. Landwehr, "Icicle plots: Better displays for hierarchical clustering," *The American Statistician*, vol. 37, no. 2, pp. 162 – 168, 1983.

[6] T. Munzner, F. Guimbretière, S. Tasiran, L. Zhang, and Y. Zhou, "TreeJuxtaposer: scalable tree comparison using Focus+Context with guaranteed visibility," in *ACM SIGGRAPH 2003 Papers*, (San Diego, California), pp. 453–462, ACM, 2003.

[7] M. Graham and J. Kennedy, "Combining linking and focusing techniques for a multiple hierarchy visualisation," in *Information Visualisation, 2001. Proceedings. Fifth International Conference on*, p. 425–432, 2001.

[8] M. Graham and J. Kennedy, "Exploring multiple trees through DAG representations," *IEEE Transactions on Visualization and Computer Graphics*, p. 1294–1301, 2007.

[9] G. W. Furnas and J. Zacks, "Multitrees: enriching and reusing hierarchical structure," in *Proceedings of the SIGCHI conference on Human factors in computing systems: celebrating interdependence*, CHI '94, (New York, NY, USA), p. 330–336, ACM, 1994. ACM ID: 191778.

[10] B. Lee, G. G. Robertson, M. Czerwinski, and C. S. Parr, "CandidTree: visualizing structural uncertainty in similar hierarchies," *Information Visualization*, vol. 6, no. 3, p. 233–246, 2007.

[11] N. Amenta and J. Klingner, "Case study: Visualizing sets of evolutionary trees," in *Information Visualization, 2002. INFOVIS 2002. IEEE Symposium on*, p. 71–74, 2002.

[12] "Infovis benchmark - PairWise comparison of trees." http://www.cs.umd.edu/hcil/InfovisRepository/contest-2003/.

[13] J. Y. Hong, J. D'Andries, M. Richman, and M. Westfall, "Zoomology: comparing two large hierarchical trees," *Poster Compendium of IEEE Information Visualization*, 2003.

[14] M. Spenke, "Visualization and interactive analysis of blood parameters with InfoZoom," *Artificial Intelligence in Medicine*, vol. 22, pp. 159–172, May 2001.

[15] D. Auber, M. Delest, J. P. Domenger, P. Ferraro, and R. Strandh, "EVAT: environment for vizualisation and analysis of trees," *IEEE InfoVis Poster Compendium*, p. 124–125, 2003.

[16] M. J. Mohammadi-Aragh and T. J. Jankun-Kelly, "MoireTrees: visualization and interaction for multi-hierarchical data," 2005.

[17] C. S. Parr, B. Lee, D. Campbell, and B. B. Bederson, "Visualizations for taxonomic and phylogenetic trees," *Bioinformatics*, vol. 20, no. 17, p. 2997, 2004.

[18] C. Plaisant, J. Grosjean, and B. B. Bederson, "SpaceTree: supporting exploration in large node link tree, design evolution and empirical evaluation," in *Proceedings of the IEEE Symposium on Information Visualization*, p. 57–64, IEEE, 1998.

[19] M. Ghoniem and J. D. Fekete, "Animating treemaps," in *Proc. of 18th HCIL Symposium-Workshop on Treemap Implementations and Applications*, 2001.

[20] M. Wattenberg, "Visualizing the stock market," in *CHI'99 extended abstracts on Human factors in computing systems*, p. 188–189, 1999.

[21] Y. Tu and H. Shen, "Visualizing changes of hierarchical data using treemaps," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 13, no. 6, pp. 1286–1293, 2007.

[22] P. Bille, "A survey on tree edit distance and related problems," *Theoretical Computer Science*, vol. 337, pp. 217–239, June 2005.
[23] J. L. Thorley and R. D. Page, "RadCon: phylogenetic tree comparison and consensus," *Bioinformatics*, vol. 16, no. 5, p. 486, 2000.
[24] T. Margush and F. R. McMorris, "Consensusn-trees," *Bulletin of Mathematical Biology*, vol. 43, pp. 239–244, Mar. 1981.
[25] C. Stockham, L. Wang, and T. Warnow, "Statistically based postprocessing of phylogenetic analysis by clustering," *Bioinformatics*, vol. 18, pp. S285–S293, July 2002.