Det. (Distance Encoded Tree): A Scalable Visualization Tool for Mapping Multiple Traits to Large Evolutionary Trees

Bongshin Lee, Lev Nachmanson, George Robertson, Jonathan Carlson, and David Heckerman

Abstract—Evolutionary biologists are often interested in finding correlations among biological traits (or attributes) across a number of species, as such correlations may lead to testable hypotheses about the underlying function. Because some species are more closely related than others, computing and visualizing these correlations must be done in the context of the evolutionary tree that relates the species. Although dozens of visualizations for correlated traits have been developed over the decades, the recent explosive growth in the number of traits and species has created a need for a visualization that can scale to dozens of traits mapped to thousands of species and their evolutionary tree to allow the interactive exploration of complex interactions. In this paper, we introduce Det., called detective, an evolutionary tree visualization that allows biologists to see multiple attributes of leaf nodes. We describe a new tree layout algorithm to represent different branch lengths and several visualization and intereaction techniques to address user requirements. We also report informal feedbacks we collected from evolutionary biologists.

Index Terms—Evolutionary tree, phylogenetic tree, tree layout, tree visualization, graphical user interfaces.

1 Introduction

Biomedical research often begins with a large scale search for correlated traits. Examples range from the "comparative method" in classical evolutionary biology, in which environmental and morphological traits are correlated to infer adaptation [4], to the thoroughly modern genome wide association studies, in which hundreds of thousands of genetic variations are screened for correlation with disease [10]. As the number of genetic sequences continues its exponential growth, there is an increasing interest in identifying underlying genetic causes of traits by comparing genetic sequences across a large number of species.

In a seminal paper in 1985, Felsenstein [5] pointed out a key flaw in the comparative method as traditionally practiced: namely, individual species cannot be considered statistically independent samples because of their shared ancestry. For example, it should not be surprising to see a large number of correlations between mice and rats when the other species in the study are reptiles. Rather, it is when a correlation persists across a diverse range of species that we should take notice.

The visualization and exploration of correlated traits is typically done by mapping the traits onto the evolutionary tree inferred for the species. The tree is typically inferred from genetic data and is annotated by branch lengths that indicate the genetic similarity between two species, with the leaves of the tree representing the species and internal nodes representing (unobserved) speciation events. Thus, the structure of the tree provides a natural visual representation of which species are generally similar to each other and which are different. If two traits are correlated with each other, but are also clustered in the same subtree, then the correlation may be explained simply by common ancestry.

In this paper, we present Det. (Distance Encoded Tree, called detective) (Figure 1), a scalable visualization tool for mapping multiple traits to large evolutionary trees. Det. employs a novel coloring scheme that allows multiple binary traits to be mapped to large trees consisting of thousands of species. By allowing multiple traits to be visualized on a large tree that preserves branch lengths, biologists can visualize complex interactions and how they relate to the evolutionary history of the species.

After providing domain-specific background with user requirements and related work that can be applied to the problem, we describe the design and features of Det. using an example from the emergence of HIV drug resistance. We then report informal feedback obtained from a number of HIV biologists. We conclude our paper with potential future work.

- *Bongshin Lee, Lev Nachmanson, George Robertson, Jonathan Carlson, David Heckerman are with Microsoft Research, E-Mail: {bongshin, levnach, ggr, carlson, heckerma}@microsoft.com.*

2 Background

In this section, we provide background on the terminology used in this paper and describe related work in evolutionary tree visualizations.

2.1 Terminology

A *phylogenetic tree*, or an *evolutionary tree*, refers to an inferred evolutionary history relating a number of species, strains, or individuals, which occupy the *tips*, or *leaves*, of the tree. The tree is parameterized by *branch (or edge) lengths*, which are the product of evolutionary rate of change and time and are additive. Thus, species that are near each other in the tree will be more similar than species that are separated in the tree, with the precise similarity given by the sum of the branch lengths along the shortest path between the two species.

The purpose of mapping genetic, morphologic, or environmental *traits*, or *attributes*, to the tree is to gain insight into the underlying mechanism. For example, if a genetic tree closely *follows the tree*, meaning species that exhibit the trait are clustered together on the tree, then one likely explanation is that the trait first arose in a *common ancestor* of the species in question; that is, the trait may have existed at the root of the subtree containing the species with the trait, and was then passed on throughout the course of evolutionary history. If two traits arose in the same common ancestor, then a high proportion of leaves in the corresponding subtree will have both traits, making the traits appear correlated if the tree is not considered. In contrast, if two traits are correlated across a diverse range of species, then that may be taken as evidence of a causal link between the species. For example, a correlation between the environmental trait “cold temperature” and the morphological trait “long fur,” which exists across a diverse range of species, may indicate that cold temperature *selects for* the *adaptation* long fur.

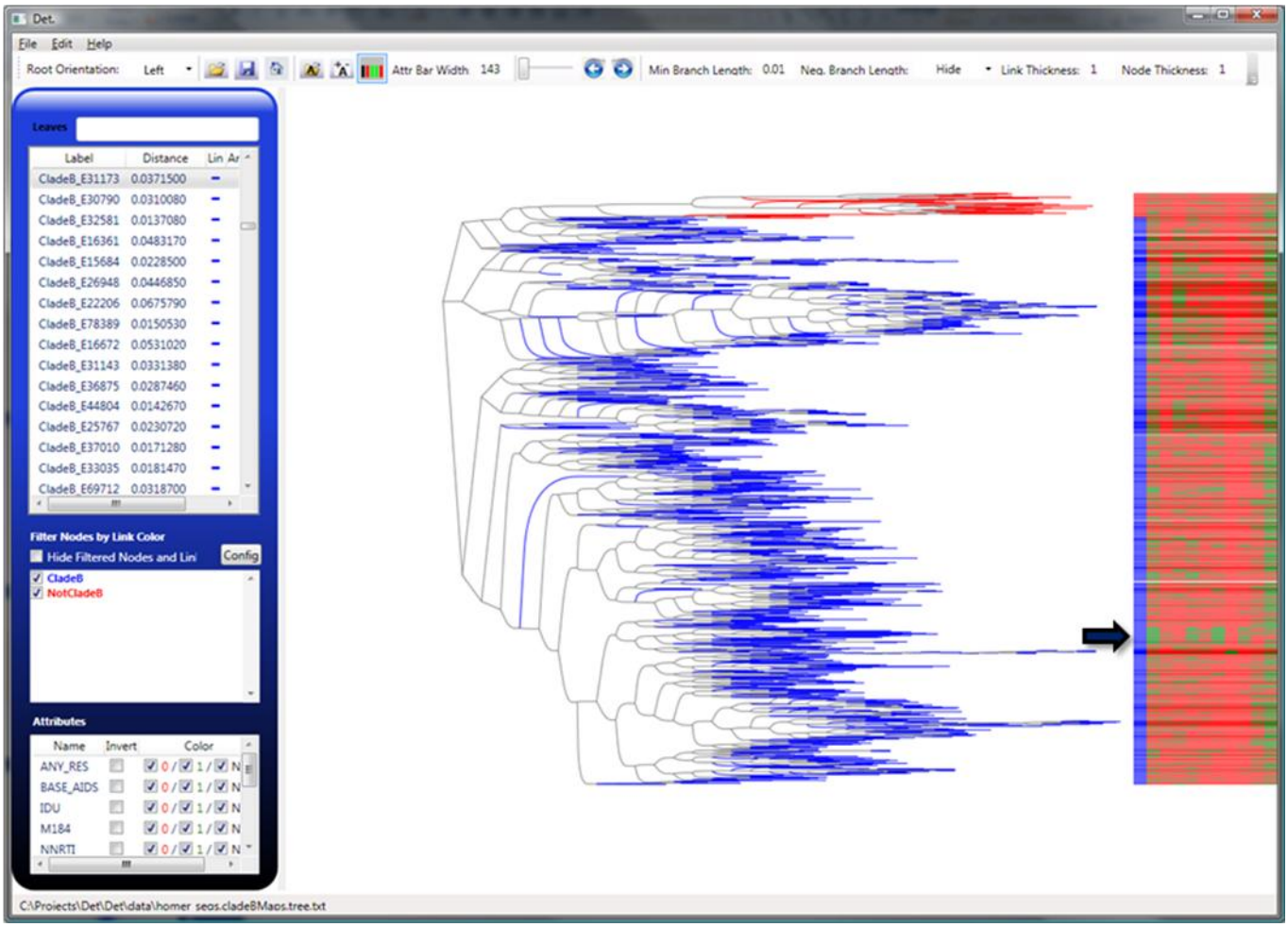

Figure 1. Det. shows ten traits mapped to 1134 species in their evolutionary tree. The black arrow, added for illustration, points to a strong clustering of HIV sequences exhibiting MDR to NRTI and 3TC.

2.2 HIV and Drug Resistance

Since its identification as the underlying cause of Acquired Immune Deficiency Syndrome (AIDS), Human Immunodeficiency Virus Type I (HIV-1) has developed into a major global pandemic, with an estimated 33 million infected cases worldwide at the end of 2007 [28]. Although there is no cure or vaccine for HIV, the development of a several classes of antiretroviral drugs lead to a precipitous drop in death rates in the developed world, after a peak from 1992-1995 when AIDS was the leading cause of death among men and women aged 25-44 in the United States [30]. Nevertheless, the extraordinary mutational capacity of HIV means mutations often arise that result in a virus that is resistant to one or more classes of *anti-retrovirals drugs* (ARVs). When a drug resistance mutation randomly occurs in a single virus, that mutation confirms an evolutionary advantage by virtue of the fact that virus replication is less affected by the drug. Thus, that mutation quickly takes over the virus population infecting that patient (called a *quasi-species*), leading to *drug failure* and a spike in viral load. If the drug regimen is not changed, the benefits of ARV therapy in reducing morbidity and mortality are lost [9].

It is thus of vital importance to understand the underlying risk factors that may contribute to drug failure. Commonly assumed risks include poor adherence to the drug regimens, *plasma viral load* levels at time of initiation of drug therapy (i.e., concentration of virus in the blood), previous exposure to similar drugs, and inherent genetic differences of circulating virus strains [9]. Recently, Harrigan and colleagues investigated the correlations between several risk factors and drug resistance to Highly Active Anti-Retroviral Therapy (HAART), which is a cocktail of several anti-retroviral drugs, using a large cohort of 1191 patients who were initiating HAART for the first time [9]. For each patient, the authors sequenced the infecting virus at several time points throughout the course of therapy. Throughout the trial, several attributes were measured, including viral load, resistance to one or more classes of drugs in HAART, and whether or not the patient was an injection drug user (IDU). Correlations to HAART failure were measured using Cox proportional hazards; the underlying phylogeny that relates the HIV quasi-species was not considered.

To demonstrate Det., we selected at least one HIV sequence per patient (total N=1134) and constructed a phylogenetic tree relating the HIV quasi-species. We then mapped each of the traits provided by Harrigan *et al*. (personal correspondence) to visualize interactions among the traits and between the traits and the evolutionary history. The specific traits included resistance to the different drug classes, defined as *M184*, indicating resistance to the 3TC ARV; *NRTI*, indicating resistance to nucleoside reverse transcriptase inhibitors; *NNRTI*, indicating resistance to non-nucleoside reverse transcriptase inhibitors; *PI*, indicating resistance to protease inhibitors; and *Any_Res*, indicating resistance to any class of ARVs (Figure 1).

2.3 User Requirements

The user (a biomedical researcher) requirements can be summarized as follows:

- In phylogenetic trees, branch lengths indicate the genetic similarity between two species, with the leaves of the tree

representing the species and internal nodes representing (unobserved) speciation events. Encoding branch lengths allows biologists to immediately identify species that are surprisingly divergent from the other species, or groups that are quite similar.

- Biologists often merge several datasets representing different cohorts. By comparing how the cohorts mix with respect to the tree, biologists can determine if the cohorts are directly comparable or if they represent substantially different populations.
- To gain insight into the underlying mechanism, multiple binary traits should be mapped to phylogenetic trees consisting of thousands of species.
- There has been recent explosive growth in the number of traits and species. So, a visualization system should able to handle dozens of traits mapped to thousands of species.

2.4 Phylogenetic Tree Visualizations

Tree visualization has been studied extensively over the last few decades. The algorithm by Reingold and Tilford [23], revisited later by Walker [29] is one of the well-known node link tree layout techniques. It produces a classical tree drawing in the sense that the drawing clearly represents the inherent hierarchy of the data. There have been several other efforts to improve node link tree layout techniques, which include a variety of 2-dimensional [2][11][22] as well as 3-dimensional approaches [14][24]. These tools are originally developed to visualize general trees, which represent only a hierarchical structure (or parent child relationships). So, the edge lengths between a parent node and a child node are determined by the layout algorithms.

There have been a handful of tools to generate and visualize the phylogenetic trees, which demonstrate evolutionary progression [21]. For example, TaxonTree [19], an extension of SpaceTree [22], has applied to the general tree visualization tool to a biodiversity domain. While TaxonTree can scale to very large trees because it accesses the data from a database, it does not encode edge lengths. PHYLIP (PHYLogeny Inference Package) [20] is one of the most popular phylogeny programs. It consists of a set of programs for estimating phylogenies and making inferences about them. However, PHYLIP uses very rudimentary tree visualization, drawing trees using '-' and '+' signs with indentation. Another well-known tool called TreeView [18] currently reads trees with up to 1000 taxa. Mesquite [13], a refinement of Macclade [12], is also commonly used by evolutionary biologists. While these tools can preserve branch lengths, they are not scalable. FigTree [6] has recently become very popular because it preserves branch lengths as well as lays out large trees. However, FigTree, as with the tree visualization tools described above, draws the links with straight lines. Precisely speaking, FigTree provides an option to set the curvature of the links, but it just changes the shape of the line rather than lay out the nodes differently. So, it does not compress the trees vertically (or horizontally depending on their root orientation). Even if it does, it cannot compress as much as the curved line does. This often results in a very cluttered drawing for the large trees, meaning that it is not as scalable as needed. For example, Figure 2 generated by FigTree and Figure 3 generated by Det. show the same tree of 1134 leaf nodes. Figure 3 is more readable and uses less vertical space.

Some of the tools have focused on comparisons of multiple trees. For example, TreeJuxtaposer [15] enables biologists to compare two large phylogenic trees by using paired tree views side-by-side and highlighting where the differences are in two trees. Mesquite [13] also provides a mirror tree view so that users can compare two different analyses for the same tree. While they show different branch lengths, all the leaf nodes are vertically aligned. This makes it impossible to compress the trees vertically.

Most of the above tree visualizations either do not show any attributes of the leaf nodes or show only one attribute at a time. This makes it very difficult to identify any correlations between attributes.

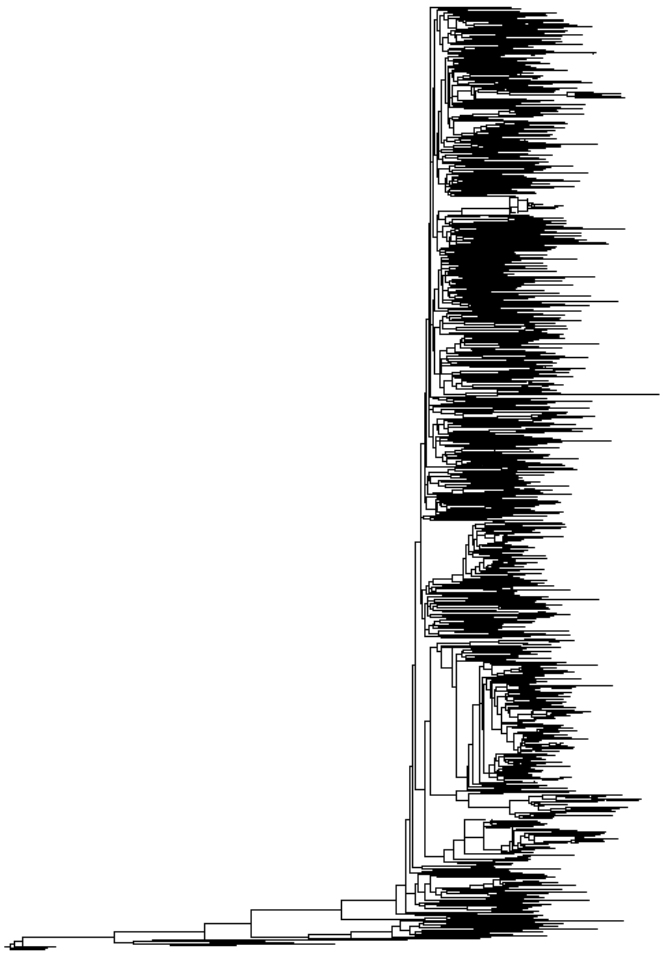

Figure 2. Screen shot of the evolutionary tree with 1134 species using FigTree [6]. The same tree drawn with Det. is shown in Figure 3.

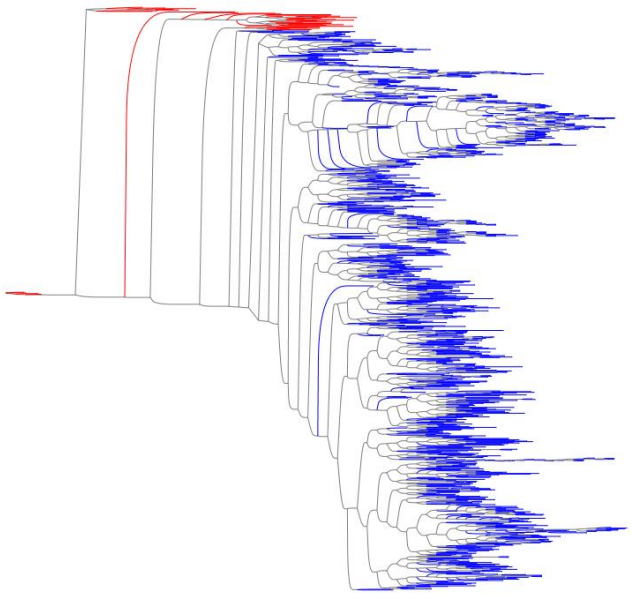

Figure 3. The tree as originally rooted by the tree building program (note that many tree building programs using mathematical models that result in "unrooted" trees, meaning any node can be chosen as the root without changing any mathematical properties.)

In contrast, in the field of genomic data analysis, the results of hierarchical clustering are typically displayed using dendrograms with a color mosaic by the leaf nodes to show the visual pattern of underlying gene expression profile data [3][8][25]. As with TreeJuxtaposer, all the leaf nodes are vertically (or horizontally depending on their root orientation) aligned, making it impossible to compress the dendrograms.

To address these problems, Det. modified the MSAGL (Microsoft Automatic Graph Layout) algorithm [16] to encode the branch lengths, utilizing layers in a tree layout. Det. also introduces a movable attribute color bar to visualize multiple attributes of the leaf nodes. Furthermore, it enables the user to filter out nodes by attributes values.

3 DET. (DISTANCE ENCODED TREE)

We developed Det. (Distance Encoded Tree, called detective) in order to help biologists better analyze phylogenetic trees. In the following sections, we describe a new tree layout algorithm that encodes edge lengths and our interface designs to address the user requirements described above.

3.1 Tree Layout to Encode Edge Lengths

Our approach to encode edge lengths in the tree layout is to utilize the powerful techniques [7] developed for the Sugiayma scheme [26], which works on a directed acyclic graph. The scheme starts by organizing the nodes of the graph into horizontal layers in such a way that each edge of the graph goes down at least one layer. This way the source nodes are positioned at the top layers and the sinks at the bottom layers. When the graph is a tree, the root is positioned at the top layer.

The original scheme does not handle the graphs where edges have prescribed lengths. Therefore, we added a step to find the layers based on the edge lengths. The following procedure allows us to assign nodes to horizontal layers. Let m be the minimum of the edge lengths. For each node v of the tree, let $d(v)$ be the length of the path from the tree root to the node, which is the sum of edge lengths on this path. The integer $l(v)=floor\ (d(v)/m)$ gives us the layer of the vertex v, where $floor(p)$ is the maximal integer that is at least as small as p, so $floor(0.9)=0$. This way two nodes u and v have the same layer if and only if $l(u) = l(v)$ (Figure 4). Since m is the minimum edge length, $d(v)/m-d(u)/m=(d(u)+n)/m-d(u)/m=n/m\geq 1$, where n is the length of edge (u, v). Then $floor(d(v)/m)-floor(d(u)/m)\geq 1$. So, $l(v)-l(u) \geq 1$ for every edge (u, v) of the tree..This ensures that every edge goes at least one layer down. We now meet to the assumptions of the scheme. The standard Sugiyama scheme reduces edge crossings by ordering nodes inside of the layers. We avoid this step because we have a tree. We just order nodes naturally preserving the order of children in the tree data file, such that for every two siblings belonging to the same layer, every node in between is also their sibling. After this layer calculation, we know Y-coordinates of the nodes; they are given by the lengths of the paths from the root. We also know the order of the nodes within the layers. To find X-coordinates we apply the biased alignment method of Brandes and Kopf [1]. We then draw splines following Nachmanson *et al*. [16].

Let us discuss the performance of the algorithm. For the X-coordinate calculation and the edge routing every tree edge spanning more than one layer is replaced by a sequence of edges going only one layer down. The new graph is called a proper layered graph. Let n be the number of nodes in the tree. In the worst case, each node of the tree creates its own layer, and this layer can intersect $O(n)$ tree edges. That means that in our situation the proper layered graph can have $O(n^2)$ nodes. The X-coordinate calculation algorithm is linear in the size of the proper layered graph. Let t be the number of nodes in the proper layered graph. We build a spatial hierarchy to route an edge and spend $O(t \log(t))$ steps on it. Since we have n - 1 edges in the tree, the total running time of our approach is $O(n^3 \log(n))$. In practice, the algorithm runs satisfactory fast. The spatial hierarchy of the edge routing step usually has the size linearly proportional to the edge span measured in the layers, and our estimation is overcautious. The algorithm lays out a tree with a thousand nodes in approximately 10 seconds on a PC with.an Intel Xeon 1560, 3 GHz processor and 2GB of memory.

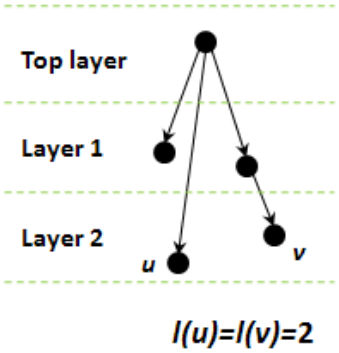

Figure 4. Illustration of the layer assignment process; node u and v are in the same layer when $l(u)=l(v)$.

3.2 Visualization Techniques

3.2.1 Color and Shape Coding for Leaf Nodes

Large cohorts often consist of heterogeneous sources and it can be useful to identify how those sources are distributed across the tree. For example, the extreme diversity of HIV-1 is often simplified by identifying individual sequences as belonging to one of several clades, which roughly map to continental regions. Because of the sequence diversity among clades, a first step in any HIV sequence analysis is to identify clades. In our drug resistance example, Harrigan and colleagues labelled each sequence as belonging to clade B (the predominant clade in North America) or not clade B. By coloring the branches and leaf nodes based on whether the strain is clade B (blue) or not clade B (red), the researchers can immediately identify the subtree that corresponds to non-clade B patients. In this case, the researchers may disregard results from this subtree, as they represent a substantially different population from the rest of the sequences.

Another common application for leaf coloring that we have observed arises when multiple centers combine data from their respective cohorts. In this context, it is important to identify whether the data from the cohorts, which are often collected from patients of different demographics, are directly comparable, which can be determined by the extent of cohort mixing in the tree.

Biologists can configure the leaf coloring and shaping information using the Color and Shape Configuration dialog (Figure 5). For example, for the nodes whose label contains the string "CladeB," biologists can color the links (and nodes) with a color “Blue” and draw them in a “Square” shape. These color/shape coding can also be used to filter out nodes and links. The names of color/shape coding (e.g., CladeB) will be shown with the mapped color ("Blue") in the Filter Nodes by Link Color list (Figure 1). When biologists uncheck the check box by their name, Det. filters out all the leaf nodes (and their incoming links) whose link color was the color mapped to that name. Biologists can either de-emphasize the links and nodes (50% transparent) or hide them.

These color and shape codings are automatically generated and saved in a configuration file. Because this configuration file is in a simple xml format, technically savvy users can directly edit it. In addition to the colors for the nodes, colors for the attribute values can be configured. While attribute color configuration is not currently incorporated in the UI, it can be easily added.

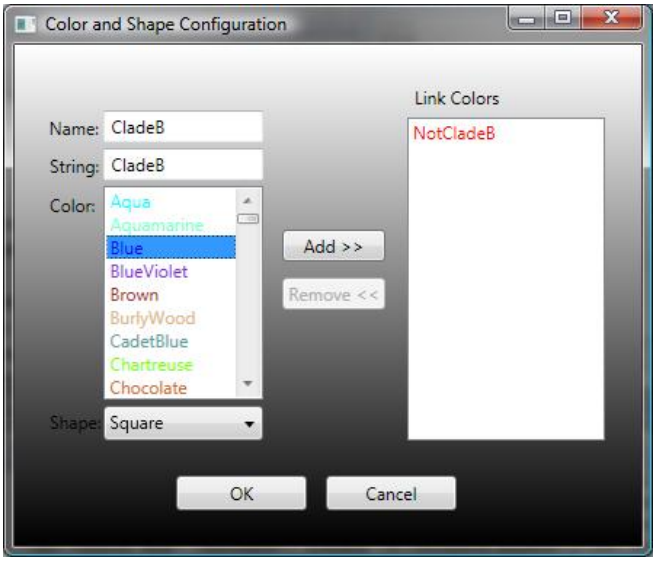

Figure 5. Biologists can configure the color and shape of the nodes based on the node labels.

3.2.2 Visualization of Multiple Binary Traits

To gain insight into the underlying mechanism, biologists need to see the multiple binary traits (attributes) mapped to phylogenetic trees. Det. enables biologists to selectively show binary traits of the leaf nodes. For example, when biologists open the attribute file for the currently opened phylogenetic tree, Det. provides the list of available attributes so that biologists can select which attributes to show using the Attributes dialog (Figure 6). When biologists open several attributes files by repeating this process, Det. merges attributes from all of the opened attributes files. Since there may be hundreds of attributes, Det. provides a way to search for specific attributes – a simple substring match with attribute name. While biologists are typing, the Attributes list is updated with search results. Biologists can change the attributes to show by using the same dialog. For example, they can add and remove attributes to and from the Selected Attributes list.

The attribute color bar, by default, located at the right (or bottom) edge of the window (depending on the root orientation). So, some of the leaf nodes could be far away from the color bar. To help biologists map the attribute value with the leaf node, the first column (or row) of the color bar is the repetition of the link color of leaf nodes (Figure 1). The second column represents the first selected attribute, the third the second selected attribute, and so on. Furthermore, Det. enables biologists to drag the entire color bar so that they can put the bar close to the particular leaf node.

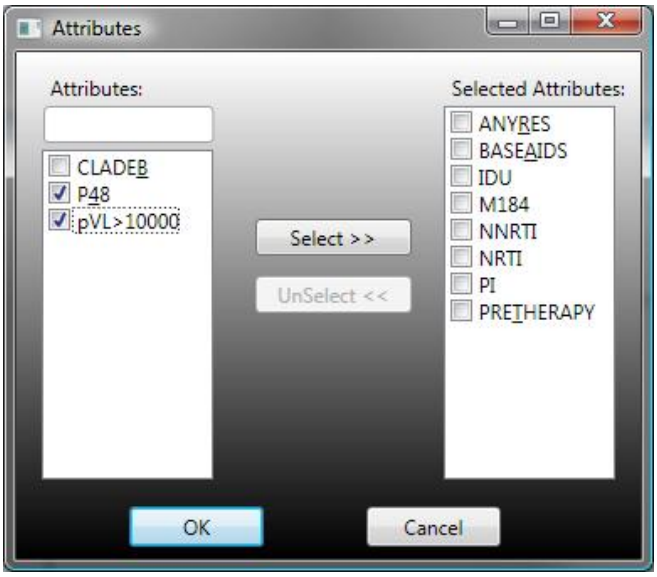

Figure 6. Biologists can select the attributes to show by adding and removing attributes to and from the Selected Attributes list.

A single column (or row) of the attribute color bar consists of rectangles each of which maps to one leaf node. Its height and vertical location are determined by the height and Y position of the matching leaf node. To make the best use of available screen space, Det. tries to compress trees vertically as much as possible. For the large trees, some of these rectangles may overlap with other rectangles, meaning that one rectangle could occlude the other rectangle. To alleviate this problem, Det. draws each rectangle 50% transparent.

The currently selected attributes are listed in the Attributes list at the bottom of the left panel (Figure 1). Biologists can filter out nodes by their attributes values. When biologists uncheck the check box by the attribute value, Det. hides all the leaf nodes (and their incoming links) with that attribute value. By default, color red and green represent value 0 and 1 respectively, and color black represents a missing value. By checking the "Invert" check box, biologists can swap color red and green so that red and green can represent value 1 and 0 respectively.

In our HIV drug resistance example, Harrigan and colleagues tracked several attributes related to drug resistance or else relevant to the demographics of the cohort. For example, they recorded attributes for baseline AIDS status at the start of therapy, plasma viral load status, whether the patient was an injection drug user (IDU), and whether the infecting HIV quasi-species has already selected for one or more resistance mutations (M184, NRTI, NNRTI, PI, AnyRes). By displaying these attributes simultaneously (Figure 1) on the tree, we can visualize several relationships reported by Harrigan *et al.*, as well as observe some new effects. For example, we see that HIV sequences sampled from IDUs tend to cluster together on the tree, an observation that is likely indicative of the fact that infection in the IDU population is circulated (in large part) separately from the rest of the cohort. Nevertheless, as later reported by the authors [32], IDU status does not appear to be strongly correlated with resistance. Rather, individual drug resistance is largely distributed throughout the tree.

By simultaneously mapping multiple traits, we can investigate the distribution of multiple drug resistance (MDR). For example, it is evident that there is a strong clustering of HIV sequences exhibiting MDR to NRTI and 3TC (indicated by an arrow in Figure 1). There are several possible explanations, for this observation, including a circulating MDR strain among a small group of patients, intrinsic sequence characteristics that make NRTI and 3TC MDR more likely, or the fact that drug resistance results in specific mutations, the accumulation of which will make resistant strains more similar. To make it easier to view specific subsets of the data, we allow biologists to hide leaves that have a given subset of the traits. For example, we can make the clustering of NRTI and 3TC MDR more evident by displaying only those sequences that exhibit both these traits (Figure 7).

3.2.3 Auto Adjustment to Support Large Trees

When biologists open a tree, Det., by default, shows an overview of the tree. To fit the whole tree in the available screen space, it automatically zooms out after laying out the tree on a canvas. When the tree is large, the links are too thin to read because they had to be significantly zoomed out. Det. automatically adjusts the widths of the links depending on the tree size and screen space. It also supports zooming so that biologists could see the details of the part (often a cluster) of the trees. Whenever the zoom factor changes, Det. automatically readjusts the widths of the links depending on the visible tree size and screen space.

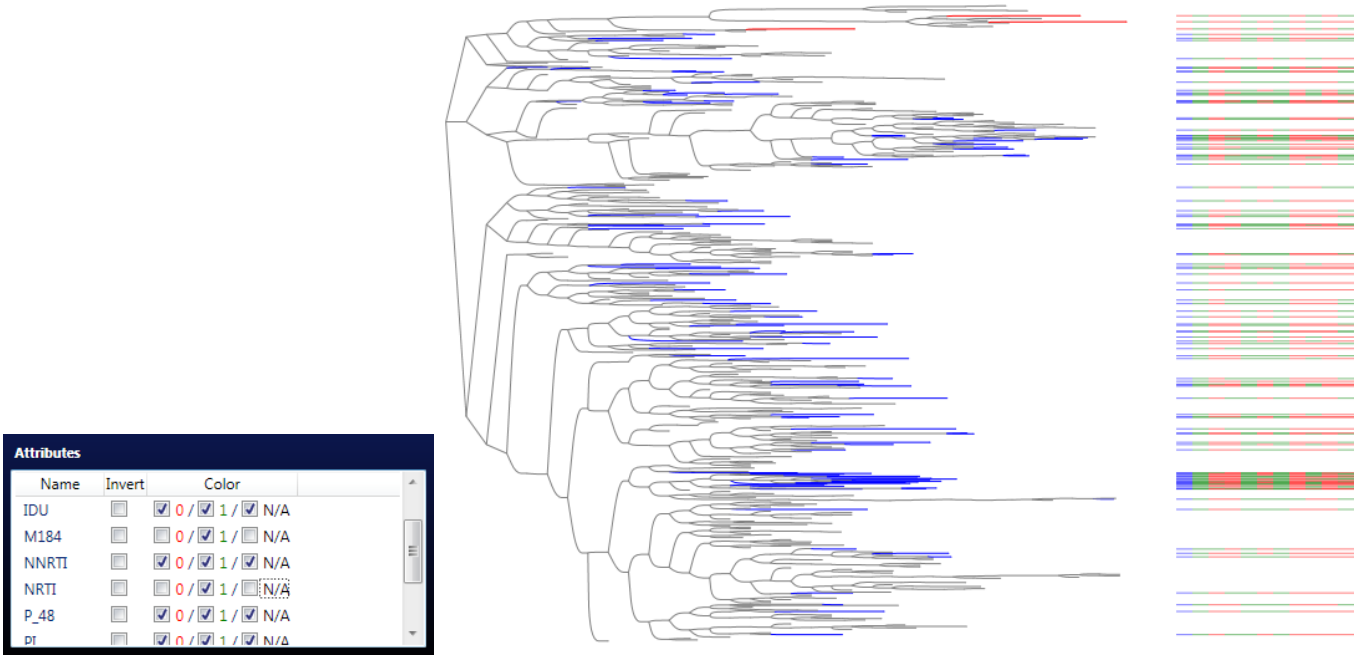

Figure 7. Biologists can filter out leaf nodes by the attribute value. This shows a tree after filtering out patients who do not have M184 or NRTI.

3.3 Additional Features

3.3.1 Reroot

The phylogenetic trees Det. handles are un-rooted trees, meaning that there is no dedicated root in the data file. Since Det. needs a specific node to be the root for the layout purposes, it chooses the first node in the data file as a root. Then, Det. allows biologists to pick any internal node and set it as the root of the tree to get a better layout. They can do it by right-clicking on any internal nodes and selecting the "Set this as root" menu from the popup menu. Figure 3 shows a tree before biologists reroot the tree shown in Figure 1. If biologists find a good layout then they can save the tree, so that they can have a same layout when they open the tree later.

While Det. is not an interactive browser that supports opening and closing of each node, it allows biologists to show/hide any branches. Det. does not re-layout the tree. However, biologists can save the tree as a new tree without hidden branches.

3.3.2 Leaves List and Annotation

The tree data file, which is in a Newick tree format [17], is not easy to read. This makes it difficult to manually edit the tree file to add any annotation. In fact, the original Newick format contains only labels and edge length. So, to enable biologists to add annotation for each leaf node, Det. embedded the annotation information into the name part of the leaf node using a delimiter ('#'). When biologists open a tree, Det. processes the result of the tree parser to compile the name part of the node into a label and annotation. Det. shows all the leaf nodes in the Leaves list -- the list shown at the left panel (Figure 1). Since there may be thousands of leaf nodes, Det. provides a way to search for specific leaf nodes – a simple substring match either with node label or with annotation. While biologists are typing, the Leaves list is updated with search results.

The Leaves list is tightly coupled with the main tree view on the right. When biologists select a leaf from the list, a leaf node mapped to the selected leaf is highlighted in the tree view. If the leaf node is off-screen, Det. animates the view to show the matching leaf node. When biologists mouse over a leaf node in the tree, Det. scrolls the list to give a focus to the matching list item.

3.4 Implementation Details

Det. was implemented in C# using WPF (Windows Presentation Foundation) [31], a unified programming model for building rich Windows client user experiences.

Det. reads one tree file and multiple attribute files. The tree files are in the Newick tree format [17]. As described before, we embedded the annotation information into the name part of the leaf node using a delimiter ('#'). For example, the annotation for the leaf node D is 'Longest' in the following sample tree:

(A:0.1, B:0.2, (C:0.3, D#Longest:0.4):0.5);

Det. supports three attribute file formats; 1) Sequence based, 2) HLA based, and 3) Generic. All the attribute files are tab-delimited text files. For more information about these attribute file formats, visit http://research.microsoft.com/vibevis/det/manual.html.

4 User Feedback

We conducted an informal interview with 7 biologists with two main goals. First, we wanted to show Det. to them and get some feedback on how they like our tool, and any obvious usability problems. Second, we wanted to better understand what kind of tools they are currently using and what their pain points are.

4.1 Meet the Biologists

We first showed Det. to one biologist who was attending the CROI HIV conference held in Boston this year. She mainly uses three different tools depending on her tasks; 1) TreeEdit [27] to edit the tree and TreeView and FigTree to see the trees. She said that “I’ve never used tree tools for analysis, but I think I can use [Det.] more as an analytical tool.”

We also showed Det. to several HIV biologists working at the Partners AIDS Research Center in Boston. We showed a demo of Det. to all of them at once in the morning. During this demo session, when we showed their data using Det. one biologist immediately noticed that “There must be something wrong. This (node) isn’t supposed to stick out.” This invoked a discussion on why it may look like that. Then they concluded with a statement “But, other than that it makes sense.” When we showed the attribute color bar, several people mentioned that “Cool,” “Right, that’s very nice.” Another positive comment was “It’s surprising that no one has developed this

tool before." After the demo session was over, we had 5 individual sessions, each of which with one or two biologists. Several biologists said that Det. was usable and had desirable content.

As explained before, our tree layout uses curved lines to compress trees. This is different than regular phylogenetic tree drawing, which uses straight lines, although some of the existing tree visualization tools allows biologists to draw the links with curved lines. When we specifically asked if the curved line caused any problems for them to analyze the trees, most of the biologists answered that it is ok. Two biologists who said they are so used to the straight lines also said that if might be because it is the first time they saw the curved lines.

There have been two very promising results since that meeting. First, one biologist really liked the color-coding of nodes and links as well as the filtering. He captured several tree images using Det. to prepare a presentation for an upcoming workshop. A few weeks later, he sent us an email with the following contents.

'To follow up on the test of the tree viewer program that you demonstrated, I have tried a couple more programs now and yours is really unique in its ability to display large trees, so that will really help for future publications. I presented some of our sequence data in NY last week, and also included screenshots that I took from trees built with your program. The response was very good. Comments and suggestions that came up to improve would be if you were able to allow changes to the thickness of branches so that they would be better visible e.g. depending of the size of the figure in Power Point Presentations. The other point was that people found it irritating that leaves are taken out when sub-populations are selected (and the remaining leaves were hard to see as lines are very thin in the big tree). As a suggestion one possible way to go would be to change the shape of leaves of a selected group e.g. to squares, circles or triangles, and keep the other leaves in the tree, just in their original shape. This would also give you the option to select several groups at a time and distinguish them by shape, which in big trees would likely be easier than only by colour. For publications or presentations ideally the user would have the option to choose the thickness, symbol sizes and colours for leaves / squares, circles, triangles etc.'

He was preparing a paper on drug resistance mutations in his sequence data in the following week or two and asked if we could help him generate a figure using Det. He also mentioned that Det. should generate high quality figures because all the biology journals use them for publication. Since his comments on changing the thickness of the nodes and links and the shape of the leaf nodes are valid and help us improve Det. we made those changes and now he has submitted a paper with images generated by Det.

Second, while two biologists were playing with Det. using their own data, they found a very interesting cluster showing that several patients from the same region, who have drug usage history, are clustered together. They immediately hypothesized that this could be caused by the fact that many patients were infected by sharing a syringe in early 90s. They captured a screenshot showing the cluster to show it to their colleagues. We were recently informed that they are planning to write a paper with that hypothesis. In fact, this is the example we described in section 3.2.2.

4.2 Make It Better

In this section, we summarize additional user requirements we collected through the meetings with biologists.

Biologists want to have a more powerful color coding mechanism. For example, they wanted to color code nodes (and their incoming links) based on the attributes of the nodes. They also wanted to interactively assign the color to the leaf nodes in a branch. If all descendents down a node have the same characteristics, then the entire nodes in that branch should be coloured the same. They also wanted to visualize non-binary attributes.

We were very surprised to learn that, for biologists, preparing images to represent their findings in data is as important as analyzing the trees. While they already have nice packages to generate and see the trees, such as PHYLIP and many phylogenetic tree visualizations, they do not have good tools for producing images for their publications. They often manually overlay icons (with different colors and shapes) to the tree images captured from the visualization tools. Several improvements we made, described in the previous section, addressed these issues. Additionally, biologists want to show the labels for the leaf nodes. We intentionally did not show labels because it is not readable at all if all the labels for a large tree are displayed. However, allowing biologists to selectively display node labels would be beneficial.

One other feature requested by several biologists is to swap two children of an internal node. This could help biologists see the subtree boundary of the node.

5 Future Work

While we iteratively refined Det. based on the user requests, there is still room for improvement. First, we need to support the additional user requirements we collected from meetings with biologists. Second, we would like to examine ways to improve the tree layout performance. The current algorithm is based a general graph layout algorithm. We might be able to get better performance if we take into account the fact that our data is always a tree. Once we accomplish these improvements, it will be important to evaluate Det. to reach more biologists. We are planning to make Det. available on the web and write an application notes paper for the Bioinformatics journal. When we have enough users, we plan to conduct a survey to collect their feedback. We also want to conduct a longitudinal case study with several biologists.

While Det. was originally developed to visualize phylogenetic trees, it is applicable to other trees when preserving edge lengths is meaningful. For example, if we visualize a decision tree where edge lengths represent a probability, the node that is farthest from the root is the most probable case. If the leaf nodes have meaningful attributes (once Det. supports visualization of non-binary attributes), Det. can help biologists find important correlations between the attribute and the highly probable cases.

Finally, next generation sequencing technologies, which are only now beginning to see widespread use, will bring new challenges to tree visualization. For example, so called pyrosequencing can yield tens of thousands of HIV sequences per patient, allowing researchers to explore the intricacies of intra- and inter-patient HIV evolution. Although Det. is highly scalable, the visualization of tens of thousands of sequences over hundreds or thousands of patients represents an entirely new challenge.

6 Conclusion

In this paper, we have presented Det. (called detective), a novel visualization tool to help biologists better understand phylogenetic trees. To address basic user requirements, Det. introduces a new tree layout that encodes edge lengths and shows mapping multiple traits to large phylogenetic trees. It also incorporates several visualization and interaction techniques. We described an application in which Det. is being used to explore novel relationships among HIV drug resistance traits. We described feedback from several biologists. We summarized the additional user requirements we learned from meetings with biologists and described how we improved Det. so that biologists can better manipulate their data for exploration and presentation. Finally we have discussed future work.

Acknowledgements

The authors wish to thank Philippa Mathews, Thomas Kuntzen, Toshi Miura, Yaoyu Wang, Jenna Rychert, Zabrina Brumme, and Chanson Brumme for their time and constructive feedback. Special thanks to Thomas Kuntzen for using Det. to generate images for his

presentation and paper. We also would like to thank Richard Harrigan for providing his valuable data and comments for the HIV drug resistance example, and Meredith Skeels, who initiated the collaboration that made this project possible.

REFERENCES

[1] U. Brandes and B. Kopf, "Fast and simple horizontal coordinate assignment," *Proc. GD '01*, Springer-Verlag, LNCS, vol. 2265, pp. 31–44, 2002.

[2] S. Card and D. Nation, "Degree-of-Interest Trees: A Component of an Attention-Reactive User Interface," *Proc. AVI '02*, 2002.

[3] M.B. Eisen, P.T. Spellman, P.O. Brown, and D. Botstein, "Cluster Analysis and Display of Genome-Wide Expression Patterns," *Proc Natl Acad Sci USA*, vol. 95, no. 25, pp. 14863-14868, 1998.

[4] J. Felsenstein. *Inferring Phylogenies*. Sunderland, MA: Sinauer Associates, Inc, 2005.

[5] J. Felsenstein, "Phylogenies and the Comparative Method," *The American Naturalist*, vol. 125, no. 1, pp. 1-15, 1985.

[6] FigTree, http://tree.bio.ed.ac.uk/software/figtree

[7] E.R. Gansner, E. Koutsofios, S.C. North, and K.-P. Vo, "A Technique for Drawing Directed Graphs", *Software Engineering*, vol. 19, no. 3, pp. 214-230, 1993.

[8] GeneMaths, http://www.applied-maths.com/ge/ge.htm

[9] P.R. Harrigan, R.S. Hogg, W. Dong, B. Yip, B. Wynhoven, J. Woodward, C. Brumme, Z.L. Brumme, T. Mo, C.S. Alexander, and J.S.G. Montaner, "Predictors of HIV Drug-Resistance Mutations in a Large Antiretroviral-Naive Cohort Initiating Triple Antiretroviral Therapy," *Journal of Infection Diseases*, vol. 191, no. 3, pp. 339-347, 2005.

[10] J. Hirschhorn and M. Daly. "Genome-wide association studies for common diseases and complex traits," *Nature Reviews Genetics*, vol. 6, no. 2, pp. 850-108, 2005.

[11] J. Lamping and R. Rao, "Laying Out and Visualizing Large Trees using a Hyperbolic Space," *Proc. UIST '94*, pp. 13-14, 1994.

[12] D.R. Maddison and W.P. Maddison, Macclade 4: Analysis of Phylogeny and Character Evolution (CD-ROM), Sinauer Associates.

[13] Mesquite, http://mesquiteproject.org/mesquite/mesquite.html

[14] T. Munzner, "H3: Laying out Large Directed Graphs in 3D Hyperbolic Space," *Proc. InfoVis '97*, pp. 2-10, 1997.

[15] T. Munzner, F. Guimbretière, S. Tasiran, L. Zhang, and Y. Zhou, "TreeJuxtaposer: Scalable Tree Comparison using Focus+Context with Guaranteed Visibility," *ACM Trans. Graphics*, vol. 22, no. 3, pp. 453-462, 2003.

[16] L. Nachmanson, G. Robertson, and B. Lee, "Drawing Graphs with GLEE", *Proc. GD '07*, Springer-Verlag, LNCS, vol. 4875, pp. 389–394, 2007.

[17] Newick format, http://en.wikipedia.org/wiki/Newick_format

[18] R.D.M. Page, "Tree View: An application to display phylogenetic trees on personal computers," *Bioinformatics*, vol.12, pp. 357-358, 1996.

[19] C.S. Parr, B. Lee, D. Campbell, and B.B. Bederson, "Visualizatios for Taxonomic and Phylogenetic Trees," *Bioinformatics*, vol. 20, no. 17, pp. 2997-3004, 2004.

[20] PHYLIP (PHYLogeny Inference Package), http://evolution.genetics.washington.edu/phylip/software.html

[21] Phylogeny Programs, http://evolution.genetics.washington.edu/phylip/software.html

[22] C. Plaisant, J. Grosjean, and B. Bederson, "SpaceTree: Supporting Exploration in Large Node Link Tree, Design Evolution and Empirical Evaluation," *Proc. InfoVis '02*, pp. 57-64, 2002.

[23] M. Reingold and J. S. Tilford, Tidier Drawing of Trees, *IEEE Trans. Software Engineering*, vol. 7, no. 2, pp. 223-228, 1981.

[24] G. Robertson, J. Mackinlay, and S. Card, "Cone Trees: Animated 3D Visualizations of Hierarchical Information," *Proc. CHI '91*, pp. 189-194, 1991.

[25] J. Seo and B. Shneiderman, "Interactively Exploring Hierarchical Clustering Results," *IEEE Computer*, vol. 35, no. 7, pp. 80-86, 2002.

[26] K. Sugiyama, S. Tagawa, and M. Toda, ''Methods for Visual Understanding of Hierarchical System Structures,'' *IEEE Trans. Systems, Man, and Cybernetics*, vol. 11, no. 2, pp. 109-125, 1981.

[27] TreeEdit, Phylogenetic Tree Editor, http://evolve.zoo.ox.ac.uk/software/TreeEdit/main.html

[28] UNAIDS/WHO. AIDS Epidemic Update 2007.

[29] J.Q. Walker II, A Node-Positioning Algorithm for General Trees, *Software – Practice and Experience*, vol. 20, no. 7, pp. 685-705, 1990.

[30] R. Weiss, "HIV and AIDS: Looking Ahead," *Nature Medicine*, vol. 9, no. 7, pp. 887-891, 2003.

[31] Windows Presentation Foundation, http://msdn2.microsoft.com/en-us/library/ms754130.aspx

[32] E. Wood, R.S. Hogg, B. Yip, W.W. Dong, B. Wynhoven, T. Mo, C.J. Brumme, J.S.G. Montaner, P.R. Harrigan. "Rates of antiretroviral resistance among HIV-infected patients with and without a history of injection drug use" *AIDS*, vol. 19, no. 11, pp. 1189-1195, 2005.