Cluster and Visualization Applications for Phylogenetic Tree Analysis

Cadran Cowansage and Tiffani L. Williams
Department of Computer Science
Texas A&M University

August 12, 2007

Abstract

Phylogenetic studies often produce thousands of trees that each represents a possible evolutionary history for a set of taxa. Large numbers of trees are difficult to analyze, particularly because there are few existing methods to do so. One common technique is to compute a representative strict consensus tree, but this approach can cause information about individual trees to be lost when their topologies are notably different. This paper investigates clustering as a tool to evaluate trees collected during the heuristic search process and employs a *cluster grid* to graphically interpret them. We identify some topological features of the trees within search spaces and present the cluster grid as a useful means for visualizing relationships among them.

1 Introduction

The process of inferring evolutionary histories among organisms with the ultimate goal of determining the relationships between all living things on earth is known as phylogenetic reconstruction [**?**]. Evolutionary history is often displayed in trees, with descendent species, or taxa, branching from ancestral ones. Determining the phylogenetic relationships that structure a tree can be difficult because many common ancestors are extinct and the fossil record is imprecise and does not include every species. Consequently, scientists infer relationships by examining common, inherited characteristics among species. It is thought that organisms that share more inherited characteristics are more likely to have descended from a common ancestor [**?**].

Often numerous hypotheses about evolutionary events are discovered and it can be difficult to compare the information contained in the resulting trees, particularly when the collection of trees is large. This paper explores clustering to compare these trees and graphical plots to represent the relationships among them.

Phylogenetic reconstruction is a computationally challenging problem that can be approached using heuristics. Maximum Parsimony (MP) is an NP-hard optimization problem that is used to search within tree-space for a phylogenetically true tree. Trees that minimize the number of branching (speciation) events within the tree, or minimize the tree length, have lower parsimony scores, and it is thought that trees with lower parsimony scores more accurately characterize evolution history among organisms [**?**].

The MP search method often produces a number of equally parsimonious solution trees for a given set of taxa. A common approach for interpreting solution, or *candidate* trees is to compute

one representative consensus tree. However, compacting the information from multiple trees into one summary tree can cause potentially valuable data to be lost [**?**].

In addition, during the heuristic search process, *search history* trees examined along the path to candidate trees are generally discarded without being analyzed. These trees can present a data-mining opportunity because they provide insight into search behavior in tree space [**?**]. A better understanding of search behavior can drive the design of better heuristics, which will ultimately lead to more accurate evolutionary trees.

Though interpreting large sets of trees is elemental in improving and understanding search heuristics, limited work has been done in this area. We use clustering to explore the topological differences within search history and candidate tree collections discovered during an MP search. We also present the cluster grid to visualize relationships among trees and enhance individual tree interpretation. We observed unique relationships within search history and candidate trees collected using different search parameters and we found that the cluster grid is an effective means to illustrate those patterns.

2 Acknowledgements

The authors wish to thank Bill Murphy and Matt Yoder for providing us with the biological dataset used in this study. We would also like to thank Hyun Jung Park for providing SLS code and Seungjin Sul for providing the Hash RF algorithm to compute the RF distance matrix.