Rapid Development and Optimization of SARS-CoV-2 Mutation-Annotated Phylogenetic Trees

Mahdi Yousuf

March 2024

1 Introduction

The aim of this project was to use rapid development and pruning of SARS-CoV-2 phylogenetic trees to create more optimized trees where the mutations most present in the virus are highlighted. The spread of the SARS-CoV-2 virus during the height of the pandemic led to the increased need for real-time genomic contact tracing. Genomic contact tracing refers to the phylogenetic placement of new samples of the disease or virus in question as it evolves and spreads, as well as tracing its evolutionary history. However, with such a large database of genetic information, it becomes more important to distinguish which mutations are more recurring as they give researchers a higher chance for gaining a better understanding of the the SARS-CoV-2 virus. The purpose of this project was to create trimmed phylogenetic trees where only the more recurring mutations were present.

2 Methods

2.1 Developing the Pruning Algorithm

To create trees with only the more recurring mutations present, I utilized a Python script that I coded to prune phylogenetic trees represented in the Protocol Buffer (pb) format. The program utilizes the Big Tree Explorer (BTE) module, which was actually developed by a team that includes one of our own, Professor Russ Corbett-Detig. The BTE module is used to traverse and manipulate mutation-annotated trees and is called upon to access the pb file. The program retrieves a list of all nodes in the tree using breadth-first expansion. It counts the occurrences of mutations across all nodes and stores them in a dictionary. Then, it filters out mutations that occur fewer than k times (k-value) and updates each node's mutations based on the filtered list. Finally, it removes leaf nodes with no mutations and no children and saves the pruned tree to a new file.

2.2 Utilizing UShER to Process the Data

The data used for this project was taken from the UCSC Genome Browser. To keep the sample size manageable, the SARS-CoV-2 data used were from datasets updated in January 2022. I downloaded the necessary FASTA files from the Genome Browser and then ran them through UShER to generate pb files. Once I generated the pb files, I then ran them through my Python program to generate the pruned tree files. Finally, the pb files were then converted to Newick (nwk) files, which can then be used to visualize the trees. In addition to generating the necessary files, the parsimony scores for each of the pb files used, both for the original and pruned versions of each tree, were calculated through UShER. A parsimony score is the sum of the smallest number of substitutions needed for each site. In other words, the shortest possible tree that explains the data is considered best.

3 Results

The Newick files generated through UShER were then uploaded to Taxonium.org to generate working data visualizations of the phylogenetic trees.

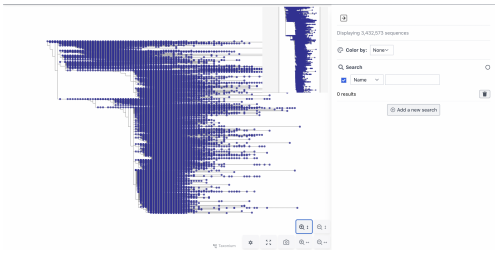

Figure 1: Phylogenetic tree before pruning

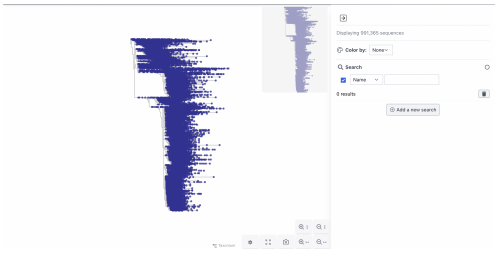

Figure 2: Phylogenetic tree after pruning

The two phylogenetic trees shown is an example of one of the tree files used for the project. As you can see in the generated trees, the phylogenetic tree of the sample dataset before it was pruned is more dense whereas the phylogenetic tree of the sample dataset after it was pruned is skinnier. The total number of sequences shown in the original tree was 3,432,573 and the total number of sequences shown in the pruned tree was 991,365, thus showing that the algorithm worked in pruning the tree.

	Original Scores	Pruned Scores
pruned_01.pb	2860754	117043
pruned_03.pb	2871499	117551
pruned_04.pb	2871357	117554
pruned_05.pb	2871483	117558
pruned_06.pb	2871539	117558
pruned_07.pb	2903461	127622
pruned_08.pb	2903589	127602
pruned_09.pb	2933711	129012

Figure 3: Parsimony scores of different trees before and after pruning

The table shown above depicts the parsimony scores of a few of the trees used for the project before and after they were optimized. The much smaller scores for the pruned trees also indicate success in optimization.

4 Discussion

Overall, the Python algorithm was successful in traversing and manipulating phylogenetic trees to create more optimized versions of them. This tool can potentially be beneficial because finding more recurring mutations in SARS-CoV-2 allows for researchers to better understand the evolutionary direction of the virus. In addition, this algorithm can be used to extract and generate information for any other virus or disease. Developing such technologies was useful during the height of the COVID-19 pandemic and they can be used again to discover more regarding other pandemics that may be happening now or in the future.

The main improvement regarding this project would be to improve the pruning algorithm to generate faster results and be more efficient overall. It did take a while for me to process all the data so improving the algorithm would certainly go a long way.

Acknowledgements

I would like to acknowledge my PI, Omar Cornejo, and his lab. The project was based on some I've been doing in the lab recently. I also wanted to acknowledge Professor Russ Corbett-Detig. A significant portion of the project was influenced and inspired by the work he has done so far in phylogenetics.

References

[1] McBroome, Jakob, et al. "BTE: A python module for pandemic-scale mutation-annotated phylogenetic trees." Journal of Open Source Software, vol. 7, no. 77, 5 Sept. 2022, p. 4433, https://doi.org/10.21105/joss.04433.