Format for Efficient Storage of Homology Relations

Week 3 Report: Understanding Queries and Inference on Gene Trees

Kevin Gao

University of Toronto

June 16, 2022

Outlines

1. Orthologs and Paralogs
2. Impact on Choice for Formats
3. References

Orthologs and Paralogs

Orthologs: Homologs separated by **speciation** events

- 1-to-1 orthologs: one to one pair
- 1-to-many orthologs: one gene is orthologous to many
- many-to-many

Paralogs: Homologs separated by **duplication** events

- same-species paralog: paralogs within the same species
- between-species paralog: paralogs in different species due to duplication in common ancestor
- fragments of the same gene: stored with separate labels on leaves?

Naive Algorithm for Inference

Boils down to ancestry query. Given two leaves, if they share a **common ancestor** in the gene tree, then they are homologous.

If the **lowest common ancestor** is a duplication node, then the two genes are paralogs of each other. If the lowest common ancestor is a speciation node, then the two genes are orthologs of each other.

Within/Between can be determined via a simple query of the species to which the genes belong. One-to-many/Many-to-many can be determined by counting the number of duplication nodes on the path from the LCA to the two leaves.

A Speed-up

The worst case for the naive algorithm occurs when two nodes do not have a common ancestor (are not homologous), in which case we won't find that out until we reach the root.

A string-based binary index like the one discussed last week can allow us to determine whether two nodes share a common ancestor without actually traversing the tree.

Batch Queries

We should also consider batch queries and queries asking for “all orthologs/paralogs/etc.” of a given gene.

For the “list all” type of queries, we look at event nodes instead of individual leaves. For a given gene represented as a leaf, to list all paralogs, we find all duplication events on the root-to-leaf paths. After we find the duplication event nodes, we list all leaves of the subtree rooted at these duplication nodes.

Outlines

1. Orthologs and Paralogs

2. Impact on Choice for Formats

3. References

If we use Newick

- We would need some special label to indicate speciation and duplication node
- Convert an LCA query to an equivalent RMQ query
- Compact but less information
- Would still have to parse the text file into a tree in memory for more complex queries

If we use PhyloXML

- VTD would be most ideal since it indexes parent-child-sibling relationship
- Implement along with a string-based index for ancestry relationship
- Streaming models can be used for extremely large data
- Well-established specification for including speciation and duplication events (e.g. recPhyloXML)

Outlines

1. Orthologs and Paralogs

2. Impact on Choice for Formats

3. References

T. C. Lam, J. J. Ding and J. Liu, "XML Document Parsing: Operational and Performance Characteristics," in Computer, vol. 41, no. 9, pp. 30-37, Sept. 2008, doi: 10.1109/MC.2008.403.

Haim Kaplan, Tova Milo, and Ronen Shabo. 2002. A comparison of labeling schemes for ancestor queries. In Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms (SODA '02). Society for Industrial and Applied Mathematics, USA, 954-963.

L. Nakhleh, D. Miranker and F. Barbancon, "Requirements of phylogenetic databases," Third IEEE Symposium on Bioinformatics and Bioengineering, 2003. Proceedings., 2003, pp. 141-148, doi: 10.1109/BIBE.2003.1188940.

Cardona, G., Rossello, F. and Valiente, G. Extended Newick: it is time for a standard representation of phylogenetic networks. BMC Bioinformatics 9, 532 (2008). https://doi.org/10.1186/1471-2105-9-532

Kmettlca, E.A. O(log n) persistent online lowest common ancestor search without preprocessing. https://github.com/ekmett/lca/

Wansong Zhang, Daxin Liu and Jian Li, "An encoding scheme for indexing XML data," IEEE International Conference on e-Technology, e-Commerce and e-Service, 2004. EEE '04. 2004, 2004, pp. 525-528, doi: 10.1109/EEE.2004.1287357.