FINAL PROJECT PROPOSAL - SCIE6062001

“A Research to Study the comparison of Phylogenetic Algorithms and help endangered animals from extinction”

Christopher Alexander Tjiandra (2502019230)

Christopher Owen (2502019180)

Arvin Yuwono (2502009721)

BINUS INTERNATIONAL

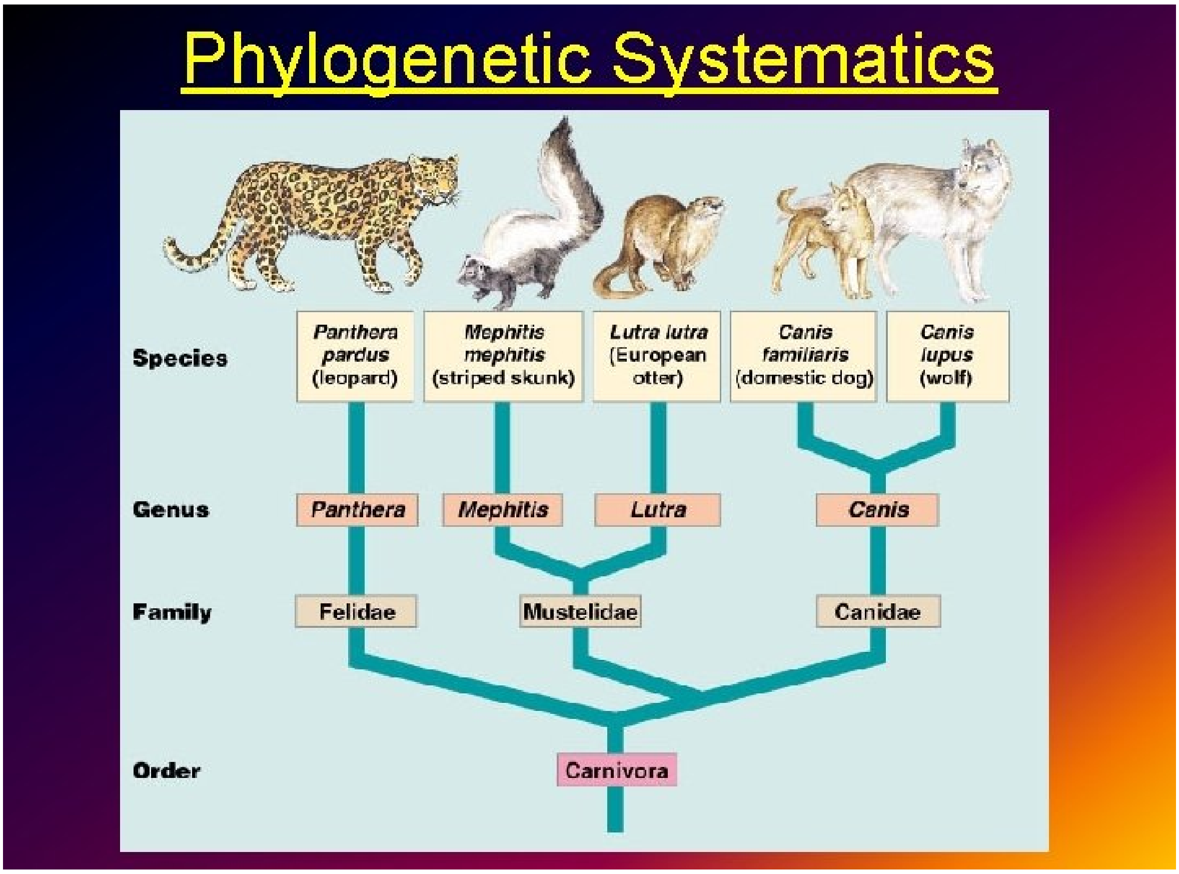
FACULTY OF COMPUTING AND MEDIA

COMPUTER SCIENCE

2023

TABLE OF CONTENTS

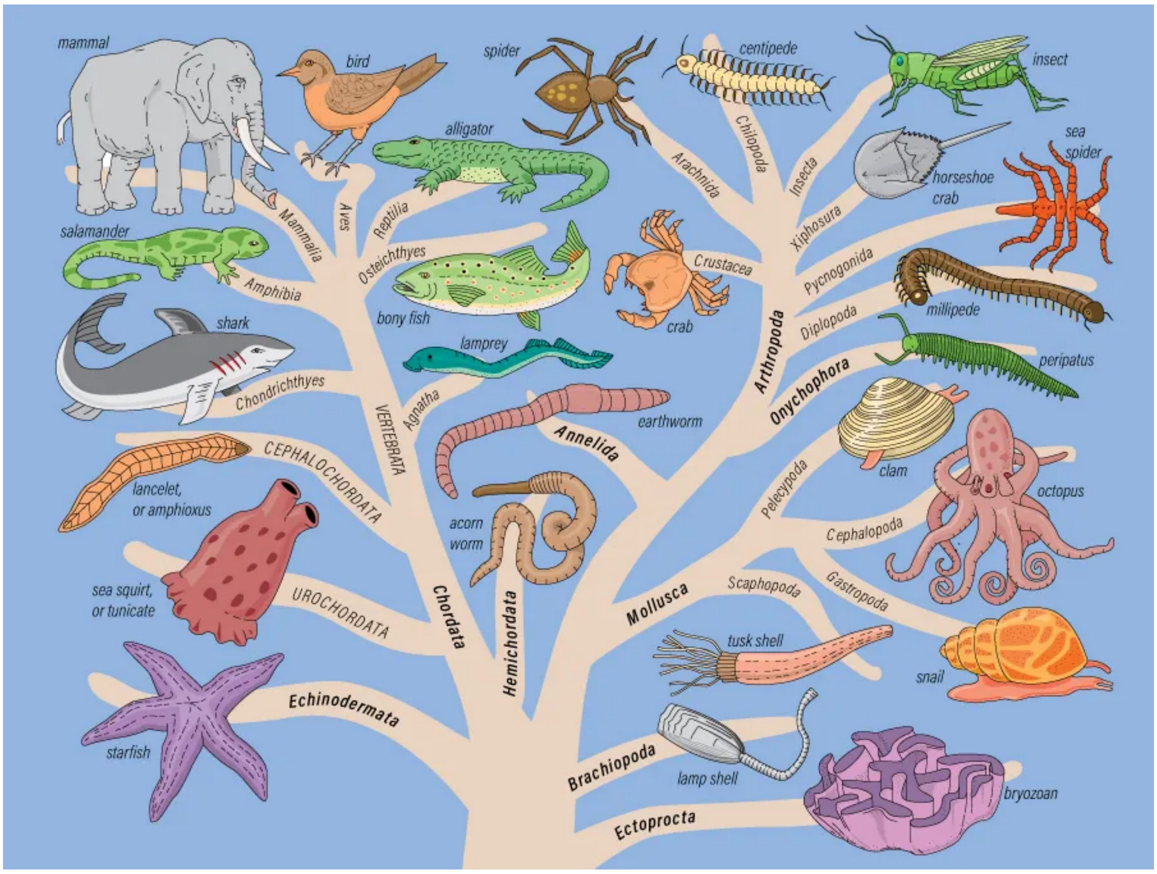
BACKGROUND



Humans continue to grow and develop rapidly every day. New inventions and advanced technology are part of human development. This is because humans are living creatures which will experience a phase of evolution where there are changes in new traits that have been inherited from previous generations. Humans are becoming smarter and more creative. However, this does not only happen to humans but also to other living things, including animals and plants. But, the different is that some of their species are endangered, while the human species is growing rapidly. We needs to help and slow down the process of extinction of these endangered species. One of the best solution is to find out the relationship between these species, how they evolve, who is their closest relatives, and etc. To find out the evolutionary relationship in a living thing, a study method is needed, which we can call as phylogenetics.

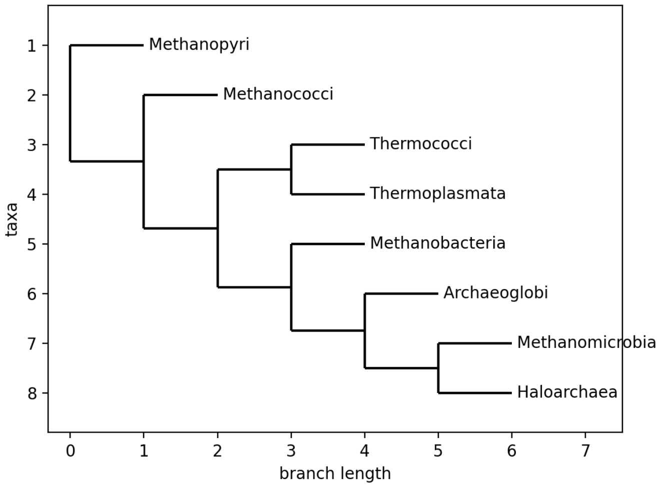
Phylogenetics is a part of biology study which focuses on determining the evolutionary relationships among various biological entities, such as species or groups of creatures. The purpose of phylogenetics is to further analyze the relationships, similarities, and differences among species after sequence alignment is done. Moreover, scientists can trace and reconstruct the evolutionary history of the experimented species. Phylogenetics relies on information extracted from the genetic materials of the experimented creatures, such as deoxyribonucleic acid (DNA), ribonucleic acid (RNA) or protein sequences. This sequence alignment method is an earlier step before implementing phylogenetics to narrow down the wider range of irrelevant genes. In addition, phylogenetics involves developing phylogenetic trees, which are diagrams that illustrate the evolutionary history and relationship between genes, usually called phylogenies. The study of biodiversity, tracing the history of species, and speculating about their shared ancestry all rely heavily on these trees. Phylogenies are helpful for classifying organisms, organizing our understanding of biological variety, and shedding light on specific evolutionary processes. Additionally, in order to completely comprehend the huge body of data that supports the theory of evolution, one needs to have a thorough understanding of phylogenies, as these trees demonstrate descendants from a common ancestor and provide much of the strongest evidence for evolution.

PROBLEM



The intricacy of displaying and analyzing phylogenetic trees is one of the fundamental issues in the study of phylogenetics. It is difficult for researchers and students to understand the subtleties of these evolutionary relationships using traditional phylogenetic analysis techniques since they require complicated algorithms and data sets. When working with enormous datasets or complex branching patterns, traditional tree diagrams can be complicated, difficult to understand, and difficult to alter. In addition, computer science as a study has not been well utilized in this field. The complexity of the phylogenetic trees makes it difficult for scientists to manage everything manually. The development of research and education in this area may be restricted by its complexity, which can make it difficult to understand and communicate the relationships between species that have evolved over time.

PROPOSED SOLUTION



To solve this issue, we suggest using Python programming to visualize phylogenetic trees in order to make them easier to understand and more approachable. Python is a flexible programming language that offers a large selection of libraries and tools that can be used for this. We will use Python as a comprehensive method to solve the difficulties of displaying phylogenetic trees in a user-friendly way. The method includes data acquisition and preparation, acquiring phylogenetic data, often DNA sequences or other evolutionary markers, from credible sources or research activities, such as Kaggle, and structuring it for analysis. Then we choose several algorithms to help us determine which one is the best for visualizing the tree by looking at some aspects. Above is the image that pretty much visualizes how the phylogenetics tree works with Python.

METHOD

Algoritms/Techniques and Libraries Used

In order to build phylogenetic trees, we'll use Python libraries like Biopython. We'll try to implement algorithms to build trees based on the supplied data while utilizing a variety of tree-building techniques and phylogenetic analysis tools. The phylogenetic tree is constructed using several algorithms; one example is the Neighbor-Joining technique. The NJ algorithm is a popular phylogenetic method for constructing a tree that best portrays the evolutionary relationships between species. It is notable for its speed and capacity to handle big datasets because it is based on the distances in the generated distance matrix. Other than that, we plan to use Maximum Likelihood method and Bayesian Inference. Maximum likelihood is another method that assesses an evolutionary history hypothesis in terms of the likelihood that the suggested model and predicted history will result in the observed data set, while Bayesian Inference produces a posterior distribution for a parameter based on the likelihood of the data produced by a multiple alignment and the prior for that parameter. The parameter is composed of a phylogenetic tree and an evolution model. After these algorithm do their job to construct the tree, we will implement machine learning to find the best performers in terms of some measurement.

To generate good visualization and instructive phylogenetic tree diagrams, we plan to use Python's broad array of visualization packages, which may include Matplotlib, Seaborn, or Plotly, which allow modification of appearances, highlighting crucial elements, and adding comments. Furthermore, interactive elements may be included, allowing users to zoom, pan, and modify tree branches for a more in-depth examination of the relationships. Along with this, we will create extensive documentation and teaching tools to help researchers, students, and educators use the tool successfully and analyze the graphical trees, promoting a better understanding of evolutionary relationships. Remember that this is our first plan, which may be modified soon along with the process of developing the project.

Flow

1. Data Collection: Obtain phylogenetic data, typically DNA sequences or other evolutionary markers, from reputable sources or research activities, such as Kaggle.
2. Data Preprocessing: Structure the acquired data for analysis, ensuring it is in the appropriate format.
3. Phylogenetic Analysis: Implement algorithms to build phylogenetic trees based on the acquired data. Utilize various tree-building techniques and phylogenetic analysis tools to ensure accuracy.
4. Visualization: Utilize Python's visualization packages, such as Matplotlib, Seaborn, or Plotly, to create visually appealing and informative phylogenetic tree diagrams. These visualizations will allow for customization, highlighting of critical elements, and the addition of explanatory annotations.
5. Comparison : Compare algorithms to find the best performers.
6. Interactivity: Incorporate interactive features, enabling users to zoom, pan, and modify tree branches for a more in-depth examination of evolutionary relationships.

Measurement

1. Measuring Robinson-Fould distance: The topological dissimilarity between two phylogenetic trees is measured using the RF distance. A lower RF distance suggests that the trees are more closely related.
2. Examine how different algorithms compare in terms of computing effectiveness and resource needs, particularly for huge datasets (time and space complexity).
3. Accuracy and error rate produced by those algorithm.

4. Etc. (might be added soon).

PROGRESS

Step on progress : Data collection

Next Step Planning : Data pre-processing

REFERENCES

https://www.ebi.ac.uk/training/online/courses/introduction-to-phylogenetics/what-is-phylogenetics/#:~:text=Phylogenetics%20is%20the%20study%20of,be%20referred%20to%20as%20taxa).

https://biopython.org/wiki/Phylo

https://medium.com/geekculture/phylogenetic-trees-implement-in-python-3f9df96c0c32

https://www.frontiersin.org/articles/10.3389/fgene.2020.584785/full

https://www.nature.com/scitable/topicpage/reading-a-phylogenetic-tree-the-meaning-of-41956/#:~:text=Phylogenies%20are%20useful%20for%20organizing,events%20that%20occurred%20during%20evolution.

https://www.deduveinstitute.be/~opperd/private/max_likeli.html#:~:text=Maximum%20Likelihood%20is%20a%20method,to%20the%20observed%20data%20set.

https://www.bionity.com/en/encyclopedia/Bayesian_inference_in_phylogeny.html

https://www.kaggle.com/code/singhakash/dna-sequencing-with-machine-learning/input

CANVA LINK :

https://www.canva.com/design/DAFyEhtmwqg/j6jYqXFz3kvpb1hiC7OWKQ/edit?utm_content=DAFyEhtmwqg&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton