IQ-NET: Fast and Accurate Quartet Phylogenetic Inference Using Deep Learning Trained on Empirical DNA Alignments

Chen Yang[1], Zixin Zhuang[1], Piyumal Demotte[1], Cuong Cao Dang[2], Le Sy Vinh[2], Bui Quang Minh[1], Nhan Ly-Trong[1,*]

[1]School of Computing, Australian National University, Canberra, 2600, ACT, Australia
[2]Faculty of Information Technology, University of Engineering and Technology, Vietnam National University, Hanoi, 144 Xuan Thuy, Cau Giay, 10000 Hanoi, Vietnam

*Corresponding author. E-mail: Trong.Ly@anu.edu.au

Abstract

Phylogenetic inference is fundamental to modern biology, with many applications including evolutionary biology, epidemiology, and comparative genomics. While maximum likelihood and Bayesian methods remain the gold standard for phylogenetic analysis, they rely on simplifying assumptions and are computationally intensive. Recent machine learning approaches for phylogenetics offer speed advantages, but have several limitations: exclusive reliance on simulated data for training, inadequate handling of gaps, and sensitivity to input sequence order. Here, we introduce IQ-NET (Intelligent Quartet NETwork), a deep learning framework that solves these limitations to infer four-taxon trees. IQ-NET estimates both tree topology and branch lengths directly from gapped alignments. IQ-NET outperforms existing machine learning methods in terms of accuracy, and obtained a 24-fold speedup compared with the widely used maximum likelihood software, IQ-TREE. We finally introduce a pipeline using IQ-NET and the ASTRAL software to reconstruct a larger species tree, i.e., with more than four taxa.

Keywords: Phylogenetic inference, Machine learning, Neural network, Quartet trees, Empirical data training

1 Introduction

Phylogenetic inference is the reconstruction of evolutionary relationships from molecular sequence data. It serves as a cornerstone of modern biology with

many applications ranging from evolutionary biology, epidemiology, ecology, to comparative genomics (Felsenstein, 2004; Grenfell et al., 2004; Delsuc et al., 2005). These methods have elucidated the evolutionary histories across many species and timescales from early life forms on Earth (Wehbi et al., 2024; Williamson et al., 2025; Dombrowski et al., 2020) to the recent emergence of the SARS-CoV-2 virus that caused the COVID-19 pandemic (Li et al., 2020; Hodcroft et al., 2021; Attwood et al., 2022; Turakhia et al., 2021; Gómez-Carballa et al., 2020).

Maximum likelihood (Felsenstein, 1981) and Bayesian inference (Huelsenbeck and Ronquist, 2001) are the gold standard for phylogenetic inference, as implemented in widely used software such as IQ-TREE (Minh et al., 2020), RAxML (Stamatakis, 2014), PHYML (Guindon and Gascuel, 2003), MrBayes (Ronquist and Huelsenbeck, 2003), and BEAST (Drummond and Rambaut, 2007). However, these methods rely on substitution models with simplifying assumptions (Felsenstein, 2004; Yang and Rannala, 2012; Jermiin et al., 2016) - that may fail to present complex evolutionary processes. Moreover, they are computationally intensive (Izquierdo-Carrasco and Stamatakis, 2011; Stamatakis, 2006), taking days or even months to analyse large datasets with millions of samples or large alignments containing hundreds of taxa.

Machine learning offers promising alternatives for maximum likelihood and Bayesian approaches with a potential for much faster inference. However, existing applications using machine learning (Suvorov et al., 2020; Zou et al., 2020; Wang et al., 2023; Smith and Hahn, 2023; Suvorov and Schrider, 2024; Kulikov et al., 2024; Nesterenko et al., 2025) face several limitations. For example, these methods rely exclusively on simulated training data, potentially limiting generalisation to empirical datasets as demonstrated by Zhu et al. (2025); Braichenko et al. (2025). Many methods (Kulikov et al., 2024; Zou et al., 2020) either ignore or fail to handle gaps, despite their prevalence in real alignments. Most studies (Suvorov et al., 2020; Zou et al., 2020; Wang et al., 2023) focus primarily on topology inference, with only one (Suvorov and Schrider, 2024) addressing branch length estimation. Moreover, most existing approaches rely on standard neural network architectures (e.g., CNNs or multilayer perceptrons) that are not permutation-invariant, making their predictions sensitive to the order of input sequences.

To address these limitations, we introduce IQ-NET (Intelligent Quartet Neural Network), an end-to-end permutation-invariant deep neural architecture for complete phylogenetic reconstruction of four-taxon trees directly from multiple sequence alignments. IQ-NET consists of three key innovations: (1) exclusive training, validating and testing on real datasets, (2) simultaneous inference of both topology and branch lengths from gapped alignments, (3) permutation invariant to sequence order. Compared with existing machine learning methods, IQ-NET achieves higher accuracy. Moreover, IQ-NET obtained a 24-fold speedup against

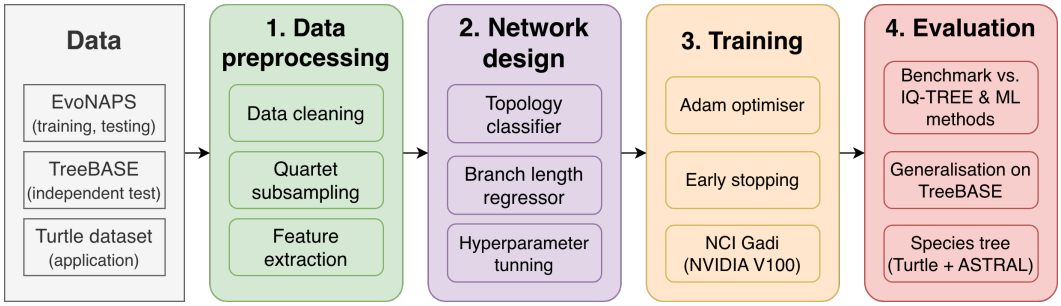

Figure 1: Overview of the IQ-NET development workflow, comprising four stages: data preprocessing, network design, training, and evaluation.

the state-of-the-art IQ-TREE, reducing the runtime from 2.67 hours to 6.7 minutes when reconstructing over 50,000 quartet trees. We further demonstrate the utility of IQ-NET to reconstruct tree with more than four species by using the quartet trees as input for the ASTRAL method (Mirarab et al., 2014), enabling the pipeline to reconstruct larger trees.

2 Materials and Methods

IQ-NET comprises two key components: (1) a quartet tree topology classifier and (2) a branch length regressor. Figure 1 provides an overview of the IQ-NET development workflow, which consists of four main stages: data preprocessing, network design, training, and evaluation. Each stage is described in detail in the following sections.

2.1 Data Preprocessing

To address the limited generalisation of neural networks trained on simulated data, as recently highlighted by Zhu et al. (2025), we trained and tested IQ-NET on real data from the EvoNAPS database (`https://github.com/Cibiv/EvoNAPS`) and conducted extensive tests on the independent TreeBASE (Piel et al., 2002) and Turtle (Chiari et al., 2012) datasets.

The EvoNAPS database comprises empirical alignments, their corresponding phylogenetic trees, and model parameters inferred by IQ-TREE v2.2.0.5. The alignments in EvoNAPS were collected from public resources: BenchmarkAlignments (`https://github.com/roblanf/BenchmarkAlignments/`), PANDIT (Whelan et al., 2006), OrthoMaM (Scornavacca et al., 2019), and TreeBASE (Piel et al., 2002). EvoNAPS includes both DNA and protein alignments and represents a diverse range of species, for example, microbes, fungi, plants, birds, turtles, and mammals.

EvoNAPS contains 48,706 DNA alignments with 4 to 2,957 taxa, and 12 to 127,813 sites per alignment. We first removed 9,998 alignments derived from TreeBASE. To avoid duplication between database versions, we further removed 14,509 alignments from OrthoMaM v10c, as EvoNAPS includes both v10c and v12a versions.

Data Cleaning

To ensure data quality, we applied two filtering criteria. First, we excluded trees containing branches longer than 9 substitutions per site, as these often indicate problematic sequence alignments (Mai and Mirarab, 2017). Second, we excluded alignments shorter than 200 sites, as these typically lack sufficient evolutionary signal for reliable phylogenetic inference. The remainder contains 29,836 phylogenetic trees, which were inferred from 24,199 alignments (5,637 alignments were used to reconstruct trees under different evolutionary models). The remaining phylogenetic trees were divided into training, validation, and testing sets with a ratio of 80:10:10.

Sampling quartet trees and alignments

From each tree with N taxa, we randomly sampled $\max(1, \min(30, \lfloor \frac{N}{4} \rfloor - 1))$ quartet subtrees along with their corresponding sub-alignments. The factor $\frac{N}{4}$ represents the maximum number of unique 4-taxon sets that can be extracted from N taxa, helping to reduce the chance of repetition. To avoid oversampling from large trees and enhance dataset diversity, we limited the number of subtrees extracted per original tree to 30. At the end, the final dataset comprised 415,303 training, 51,401 validation, and 51,241 testing samples.

Feature Extraction

To encode the input alignments, we extracted site pattern frequencies as features, following the approach proposed in Leuchtenberger et al. (2020). This encoding approach is widely adopted in maximum likelihood-based phylogenetic methods, such as IQ-TREE, RAxML, and PhyML. Specifically, DNA sequences are represented using five characters: the four nucleotides (A, C, G, T) and the gap ('-'). For an alignment of four taxa, this results in $5^4 = 625$ possible site patterns. We counted the occurrences of each site pattern in the alignment and normalised them by the alignment length, ensuring that the resulting frequency vector sums up to one. In other word, each alignment is represented by a 625-element vector.

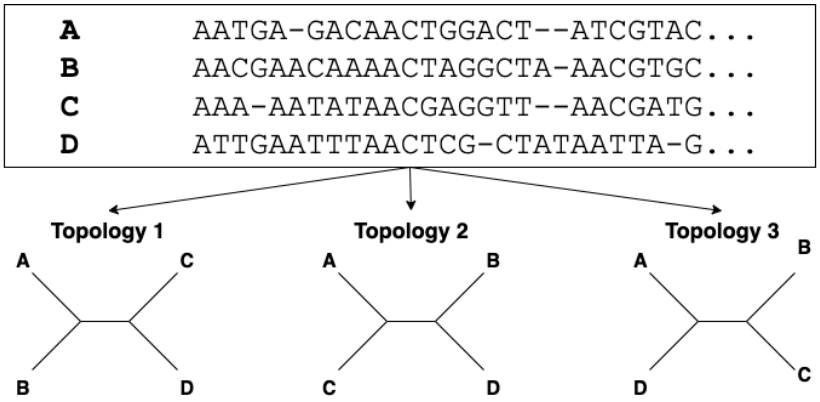

Figure 2: Three possible unrooted topologies for four taxa A, B, C, and D. Topology 1: $AB|CD$; Topology 2: $AC|BD$; and Topology 3: $AD|BC$.

2.2 Neural Network Design

Let $[S_A, S_B, S_C, S_D]$ denote the input sequences representing the four taxa A, B, C, and D.

2.2.1 Tree Topology Classifier

For any four-taxon alignment $[S_A, S_B, S_C, S_D]$, there are three possible unrooted quartet tree topologies (Figure 2): Topology 1: $AB|CD$; Topology 2: $AC|BD$; and Topology 3: $AD|BC$. Our classifier extracts site-pattern frequencies from the input alignment and predicts a score for each topology, selecting the topology with the highest score.

Sequences within an alignment can appear in any order, thus, the same alignment can be represented by $4! = 24$ permutations. These permutations may produce different site pattern frequencies and consequently different predicted scores for the three topologies. To ensure that the classifier consistently predicts the same topology regardless of sequence order, we adopted the symmetry-preserving architecture proposed by Tang et al. (2024).

The Adapted Symmetry-preserving Design

A permutation is said to 'preserve' a topology if it does not change the grouping implied by that topology. For example, assuming that the true topology for an alignment $[S_A, S_B, S_C, S_D]$ is Topology 1: $AB|CD$, which groups the first (S_A) and the second (S_B) sequences in the alignment together. Then swapping the first two sequences to $[S_B, S_A, S_C, S_D]$ results in the tree $BA|CD$, which still groups the first two sequences together (thus, remaining Topology 1). Therefore, the permutation $[S_B, S_A, S_C, S_D]$ preserves Topology 1. In contrast, the permutation $[S_A, S_D, S_B, S_C]$ result in a tree that groups the first and the third sequences together, thus, 'does not preserve' Topology 1.

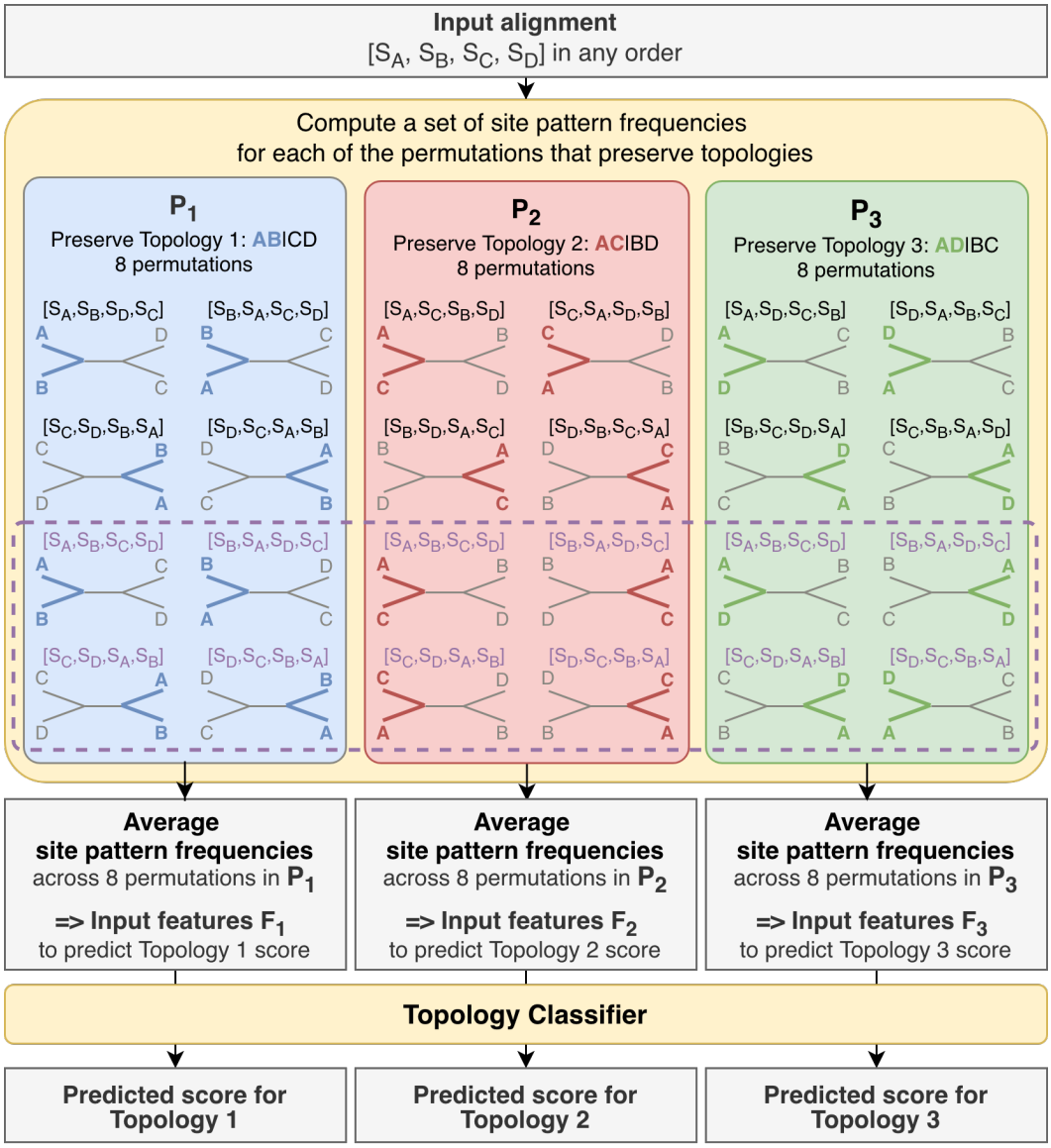

Figure 3: Symmetry-preserving design for the topology classifier. The input is an alignment $[S_A, S_B, S_C, S_D]$ (in any order). Among the 24 permutations of the input alignment, 16 preserve at least one topology and are partitioned into three sets: P_1, P_2, and P_3, containing permutations that preserve Topologies 1, 2, and 3, respectively. Four permutations (purple dashed box) appear in all three sets, as they preserve all topologies. For each topology i ($i \in \{1, 2, 3\}$), the site-pattern frequencies from the eight permutations in P_i are averaged to produce a single feature vector F_i. The three resulting feature vectors are then passed sequentially to the topology classifier to predict scores for each of the three topologies.

Of the 24 possible permutations, 16 preserve at least one topology. These are partitioned into three sets P_1, P_2, and P_3, each containing 8 permutations that preserve Topologies 1, 2, and 3, respectively (Figure 3). Four permutations appear in all three sets (purple dashed box in Figure 3), as they preserve all three topologies. The remaining 8 permutations alter the underlying topology (Suppl. Table S1) and

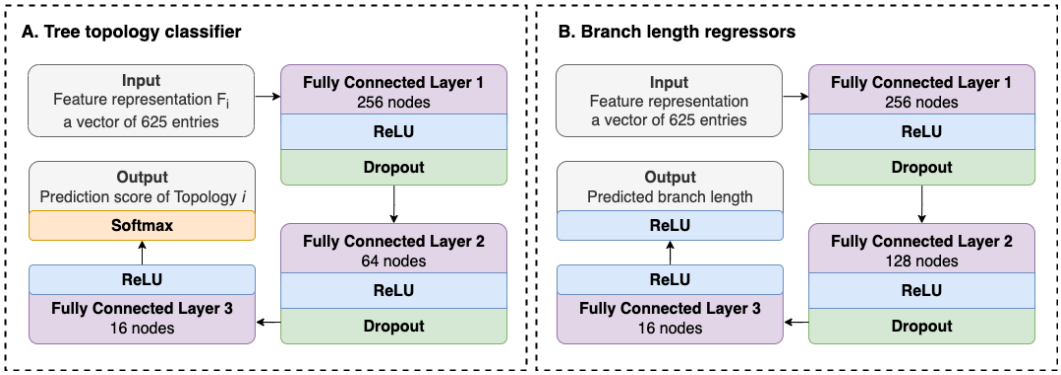

Figure 4: The network architecture of (A) the tree topology classifier; and (B) the branch length regressors.

are excluded from our symmetry-preserving design.

The full pipeline is illustrated in Figure 3. Given an input alignment with sequences in any order, site-pattern frequencies are computed for all permutations in P_1, P_2, and P_3. For each topology i ($i \in \{1, 2, 3\}$), a feature vector F_i is constructed by averaging the site-pattern frequencies across eight permutations in P_i. The three resulting feature vectors are then fed into the topology classifier sequentially to predict scores for each of the three possible topologies.

The Network Architecture

Figure 4A illustrates the architecture of the tree topology classifier, a five-layer fully-connected neural network. The network takes a feature vector F_i, which contains 625 averaged site pattern frequencies, as input. That input is processed through three hidden layers with decreasing dimensionality of 256, 64, 16, before outputting a prediction score for the Topology i. For each input alignment, the network is applied three times to predict the scores for all three possible topologies. Dropout regularization is applied at the first two hidden layers to reduce overfitting. The hidden layers use rectified linear unit (ReLU) activation functions (Glorot et al., 2011), while the output layer employs softmax activation to convert raw network outputs into class probabilities (Goodfellow et al., 2016).

Hyperparameter Tunning

We employed Optuna (Akiba et al., 2019) to fine tune the hyperparameters of our classifier through 200 trials. Each trial samples one candidate configuration from predefined parameter ranges. The configuration yielding the lowest validation loss was selected for final network training. Supplementary Table S2 summarises the search ranges and the best values.

2.2.2 Branch Length Regressor

The branch length regressor estimates five non-negative values: one internal and four external branch lengths. The internal branch length remains unchanged for any input sequence order, while the external lengths should permute accordingly. To enforce these constraints, we implemented separate regressors for the internal and external branches.

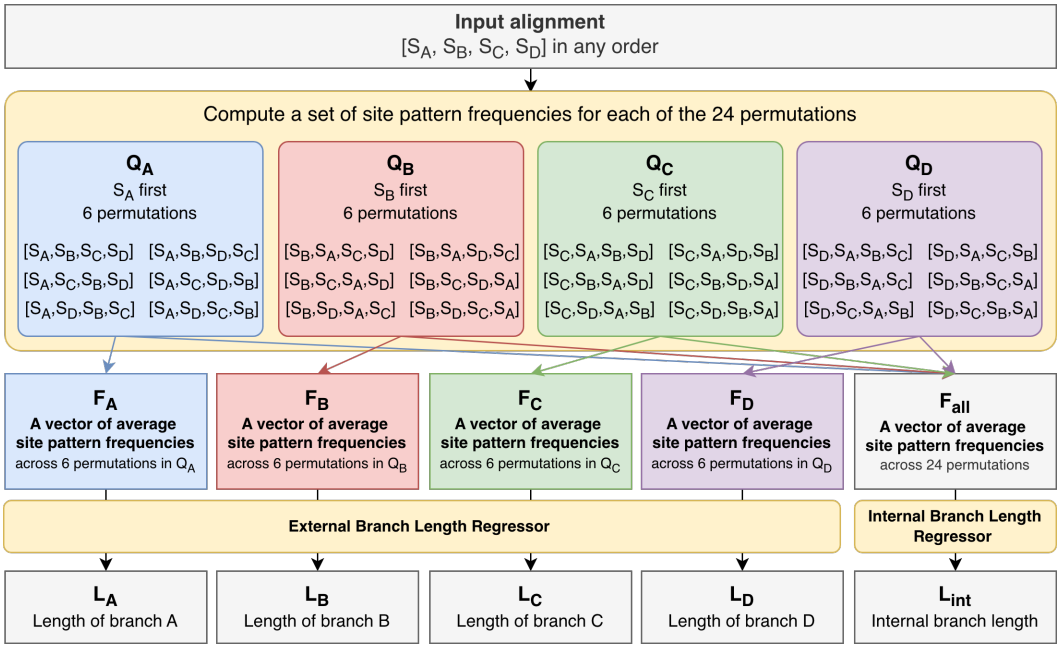

Figure 5: Symmetry-preserving design for the branch length regressor. All 24 permutations of the input alignment are partitioned into four sets based on the first taxon: Q_A (S_A first), Q_B (S_B first), Q_C (S_C first), and Q_D (S_D first), each containing six permutations. Site-pattern frequencies within each set are averaged to form four feature vectors (F_A, F_B, F_C, F_D). A fifth feature vector, F_{all}, is obtained by averaging across all 24 permutations. The vectors F_i ($i \in \{A, B, C, D\}$) are used to predict the corresponding external branch lengths L_i, while F_{all} is used to predict the internal branch length L_{int}.

Figure 5 illustrates the full pipeline for branch length prediction. For the external branches, the 24 permutations of the input alignment are partitioned into four groups according to the first taxon. For example, Q_A contains the six permutations in which S_A appears first; Q_B, Q_C, and Q_D are defined similarly. The site-pattern frequencies within each group are averaged to produce feature vectors F_A, F_B, F_C, and F_D. Each vector, associated with a taxon $i \in \{A, B, C, D\}$, is fed into the external branch length regressor sequentially to estimate the corresponding external branch length L_i. For the internal branch, site-pattern frequencies are averaged across all 24 permutations to form F_{all}, which is then used to estimate the internal branch length L_{int}. This design ensures the predicted branch lengths remain consistent regardless of the input sequence order.

The Network Architecture

Figure 4B illustrates the architecture of the branch-length regressors, each implemented as a five-layer fully connected neural network that outputs a single branch length. The same architecture is used for both internal and external branch length regressors. This architecture resembles the topology classifier but with two key differences. The three hidden layers have sizes of 256, 128 (instead of 64), and 16, respectively. ReLU activation functions are applied to both hidden and output layers. For the four external branches, four separate feature vectors are fed sequentially into the external branch length regressor to predict each external branch length independently (Figure 5).

Hyperparameter Tunning

Similar to the tree topology classifier, we employed Optuna to fine-tune the hyperparameters of the branch length regressors. The search ranges and the best hyperparameter settings are summarised in Supplementary Table S3.

2.3 Neural Network Training

All networks were trained using the Adam optimiser (Kingma and Ba, 2014) with exponentially decaying learning rates. Cross-entropy and mean squared error (MSE) were used as the loss functions for the tree topology classifier and the branch length regressor, respectively. To mitigate overfitting, we implemented early stopping, terminating training if the validation accuracy failed to improve for 5 consecutive epochs.

Training was conducted on the Gadi supercomputing system at the National Computational Infrastructure (NCI), Australia (`https://nci.org.au/`), using an NVIDIA Tesla V100-SXM2-32GB GPU.

2.4 Evaluation

We conducted an extensive evaluation of IQ-NET through three key assessments. First, we benchmarked it against IQ-TREE and existing machine learning methods using the testing set from the EvoNAPS database. Second, we evaluated IQ-NET on an independent TreeBASE dataset to assess its generalisation. Finally, we demonstrated a practical application of IQ-NET by providing quartet trees to ASTRAL for species tree reconstruction using the Turtle dataset (Chiari et al., 2012).

2.4.1 Benchmark IQ-NET against IQ-TREE and Existing Machine Learning Methods

We benchmarked IQ-NET against IQ-TREE and several state-of-the-art machine learning methods. For topology prediction, comparing methods included Fusang (Wang et al., 2023), DeepNNPhylogeny (Kulikov et al., 2024), and the neural network by Suvorov et al. (2020) (hereafter referred to as 'Suvorov-topology'). Since DeepNNPhylogeny does not support gapped alignments, all gapped sites were removed prior to its evaluation. For branch length estimation, we compared IQ-NET with IQ-TREE. We did not benchmark the published machine learning method by Suvorov and Schrider (2024), because that study provides a collection of many pretrained neural networks - each trained on data simulated under specific conditions - rather than a universal general-purpose neural network. Consequently, no single pretrained network could be appropriately selected for real datasets.

Benchmarks were performed on the testing dataset extracted from the EvoNAPS database (see Section 2.1). EvoNAPS contains maximum likelihood trees inferred from empirical alignments. We randomly subsampled quartet trees from these original maximum likelihood trees and used them as ground truth. This approach leverages additional phylogenetic information from the original many-taxon alignment to produce more reliable phylogenetic trees compared to direct inference from four-taxon alignments. In contrast, when IQ-TREE served as a benchmark method, it directly inferred four-taxon trees from four-taxon alignments; we refer to this as 'IQ-TREE-quartet' to avoid confusion. All benchmarks were conducted on Gadi using a single core of an Intel Xeon Platinum 8274 (Cascade Lake, 3.2 GHz) CPU with 4 GB of allocated RAM.

2.4.2 Evaluate IQ-NET's Generalisation on the Independent TreeBASE Dataset

To further assess the generalisation of IQ-NET, we evaluated it on an independently TreeBASE dataset (Piel et al., 2002). Noting that all TreeBASE-derived data were removed from EvoNAPS during the development of IQ-NET, ensuring complete independence between our training and testing data. We then downloaded all TreeBASE studies, but filtered out those containing multiple trees, non-DNA data, less than four sequences, or more than 40 sequences. The resulting set consists of 921 trees and their corresponding alignments. For computational feasibility, we generated three testing subsets by randomly sampling 0.1%, 1%, and 10% of all possible quartets from each original TreeBASE alignment, yielding 62,494, 234,151, and 2,336,286 quartets, respectively. Since many TreeBASE trees do not include branch lengths, we assessed IQ-NET's performance only on topology prediction.

2.4.3 Reconstruct Species Trees from Quartet Trees using the Turtle Dataset

We demonstrate the application of IQ-NET for species tree construction using a phylogenomic data set comprising 16 vertebrate taxa and 248 genes (Chiari et al., 2012). This data set contains approximately 187,000 base pairs and focuses on resolving the phylogenetic placement of turtles relative to birds and crocodiles. The original study demonstrated that the inferred relationships among crocodiles, birds, and turtles are sensitive to the choice of phylogenetic model: under single-site homogeneous DNA substitution models, crocodiles and turtles form a clade, whereas partitioned models group birds and crocodiles as a sister clade to turtles. Here, we assess IQ-NET's ability to recover the relationships among crocodiles, birds, and turtles.

For the analysis, we subsample 4-taxon alignments from each gene. Given a gene contains N species, we consider all possible combinations of 4-taxon alignments $\binom{N}{4}$ and randomly subsample a proportion of these combinations without replacement ranging from 10% to 100%. Sampling x% of all possible 4-taxon alignments from a gene results in $\frac{x}{100} \times \binom{N}{4}$ alignments. For each sub-sampled alignments, IQ-NET is used to infer a corresponding quartet tree. The resulting quartet trees are subsequently provided as input to ASTRAL version 5.7.8 (Mirarab et al., 2014) for species tree construction. To examine the effect of third codon positions, we repeated the analysis after removing the third codon position from each gene, reconstructed quartet trees, and inferred species trees using ASTRAL with these modified quartet trees.

3 Results

The tree topology classifier converged at epoch 13 and completed training after 18 epochs (Suppl. Figure S1A), while the branch length regressor converged at epoch 2 and finished training after 7 epochs (Suppl. Figure S1B). The rapid convergence of both networks can be attributed to the use of compact, information-rich site-pattern frequency vectors as input features, together with relatively small network architectures.

3.1 Benchmark Results

We first compared the accuracy and runtime of IQ-NET with IQ-TREE-quartet (IQ-TREE inferring trees directly from quartet alignments) and existing machine learning methods. This benchmark used the testing set extracted from the

EvoNAPS database, where the ground-truth quartet trees were subsampled from larger maximum likelihood trees inferred by IQ-TREE from the original multi-taxon alignments (see Section 2.1).

3.1.1 IQ-NET Outperforms Existing Methods in Tree Topology Prediction

IQ-NET achieved the highest accuracy in predicting tree topologies, with an average accuracy of 82.3%, followed by IQ-TREE-quartet (79.9%), DeepNNPhylogeny (79.4%), Fusang (76.6%), and Suvorov-topology (57.7%). Moreover, IQ-NET demonstrated balanced performance across all three topologies (82-83% accuracy) without bias toward any particular topology (Suppl. Figure S2).

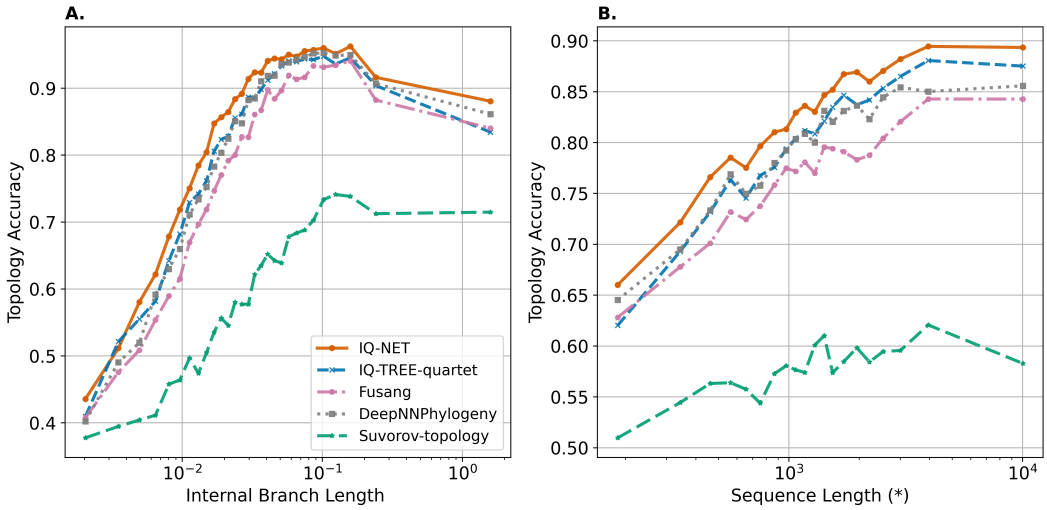

Figure 6: Topology prediction accuracy vs. (A) internal branch length; and (B) sequence length. (*) Sequence length excludes fully gapped sites, which contain no evolutionary signal.

We also evaluated the impact of internal branch length and sequence length on prediction accuracy as these factors are known to influence the performance of maximum-likelihood methods. As expected, increasing either internal branch length or sequence length tends to improve the accuracy of all methods (Figure 6), as longer branches and sequences provide more evolutionary signals (i.e., more mutations). However, accuracy declined when the internal branch length exceeded 0.24 or when the sequence length reached 3,950 sites. A possible explanation for this phenomenon lies in the characteristics of our testing dataset: samples with long internal branches often derive from short sequences (Suppl. Figure S3), making topology reconstruction more difficult. Conversely, long sequences in our dataset tend to be highly similar, resulting in short internal branch lengths (Suppl.

Figure S3) that also challenge the topology inference. Another explanation is that long sequences may contain multiple genes that have evolved independently under different evolutionary processes, potentially supporting multiple tree topologies rather than a single one.

3.1.2 IQ-NET Excels in Branch Length Prediction

Figure 7 shows scatter plots of reference versus predicted branch lengths for IQ-NET and IQ-TREE-quartet (scatter plots for each of the four external branches are further provided in Suppl. Figure S4). Note that reference branch lengths were those from quartet trees sampled from larger maximum-likelihood trees inferred by IQ-TREE on the original multi-taxon alignments. IQ-NET closely match the reference values, with a Pearson correlation of 0.90 and a slope of 0.81, compared with 0.68 and 0.69 for IQ-TREE-quartet.

The limited accuracy of IQ-TREE-quartet could be explained by the challenge of estimating branch lengths from four-taxon alignments. In these subsampled quartets, each branch represents a composite path spanning multiple internal nodes in the original larger tree, resulting in longer branch lengths that are more prone to substitution saturation. Under saturation, the likelihood surface flattens, making it difficult to distinguish, for example, a branch length of 1.0 from one of 2.0 or greater. In contrast, IQ-NET appears less affected by this limitation.

Supplementary Table S4 presents the estimation errors of IQ-NET and IQ-TREE-quartet. IQ-NET outperformed IQ-TREE-quartet on most metrics, achieving lower Mean Squared Error (MSE), Mean Absolute Error (MAE), and Branch Score Distance (BSD) (Kuhner and Felsenstein, 1994). However, IQ-TREE-quartet obtained a lower Mean Relative Error (MRE). Since MRE is computed by dividing the absolute error by the reference value, shorter branches have a greater influence on this metric. Thus, the lower MRE suggests that IQ-TREE-quartet is more accurate at predicting shorter branches.

We also examined the influence of branch length and sequence length on branch length estimation. Increasing branch length tends to increase prediction error for both methods (Suppl. Figures S5A and C). IQ-NET performs comparably to IQ-TREE-quartet on short branches but clearly surpasses IQ-TREE-quartet on longer ones (> 0.02).

In contrast, increasing sequence length tends to enhance the accuracy of both methods (Suppl. Figures S5B and D). For short alignments, IQ-NET is more accurate than IQ-TREE-quartet, while for long alignments ($> 1,000$ sites), the performance gap becomes negligible.

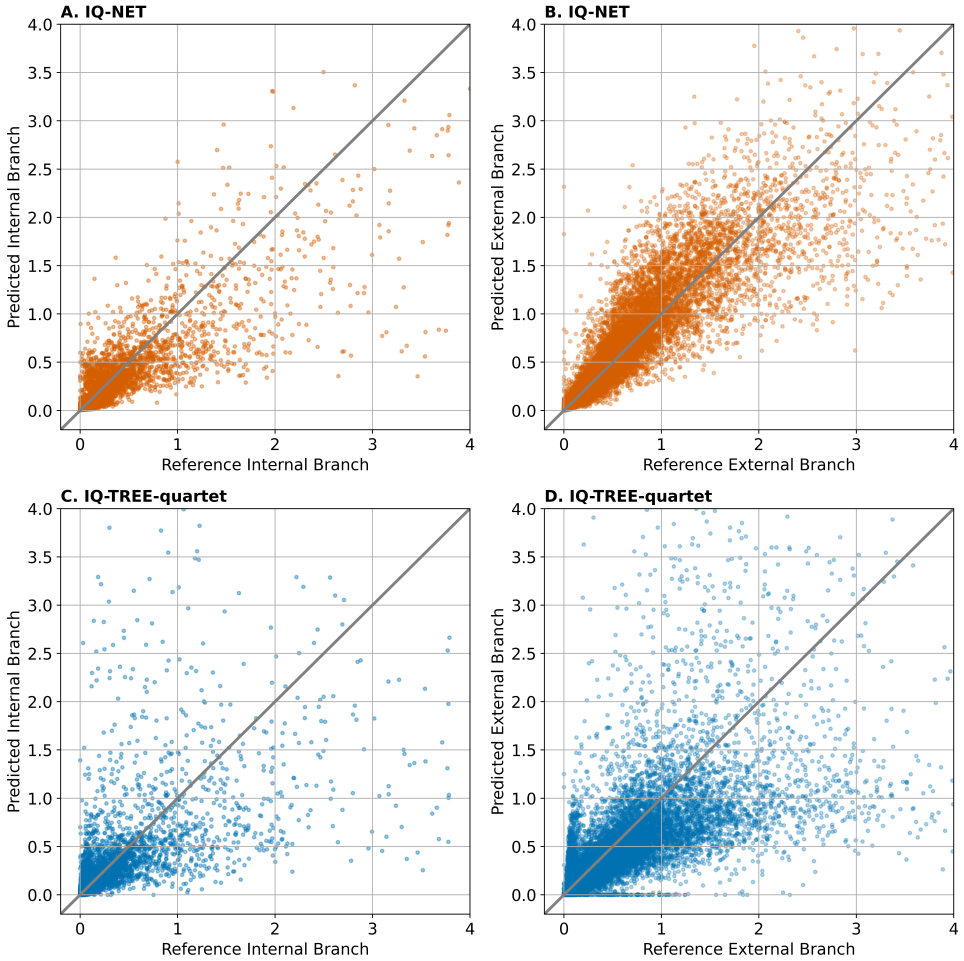

Figure 7: Scatter plots of reference vs predicted branch lengths estimated by IQ-NET (Panels A and B) and IQ-TREE-quartet (Panels C and D), where reference branch lengths are those of the quartet trees sampled from larger maximum likelihood trees inferred by IQ-TREE on the original multi-taxon alignments. Panels A and C correspond to internal branches, whereas B and D correspond to external branches.

3.1.3 IQ-NET Achieves 24-Fold Speedup in Quartet Tree Inference

We compared the runtime of IQ-NET and IQ-TREE-quartet since they are the only two methods among our benchmarks that can perform complete quartet tree inference, including branch length estimation. Using a single CPU core, IQ-NET required 6.7 minutes to reconstruct 51,241 quartet trees compared with 2.67 hours for IQ-TREE, a 24-fold speedup.

For fairness, it should be noted that IQ-NET, as a machine learning method, required initial training, which took 13.75 and 15.37 GPU minutes for the tree topology classifier and branch length regressors, respectively.

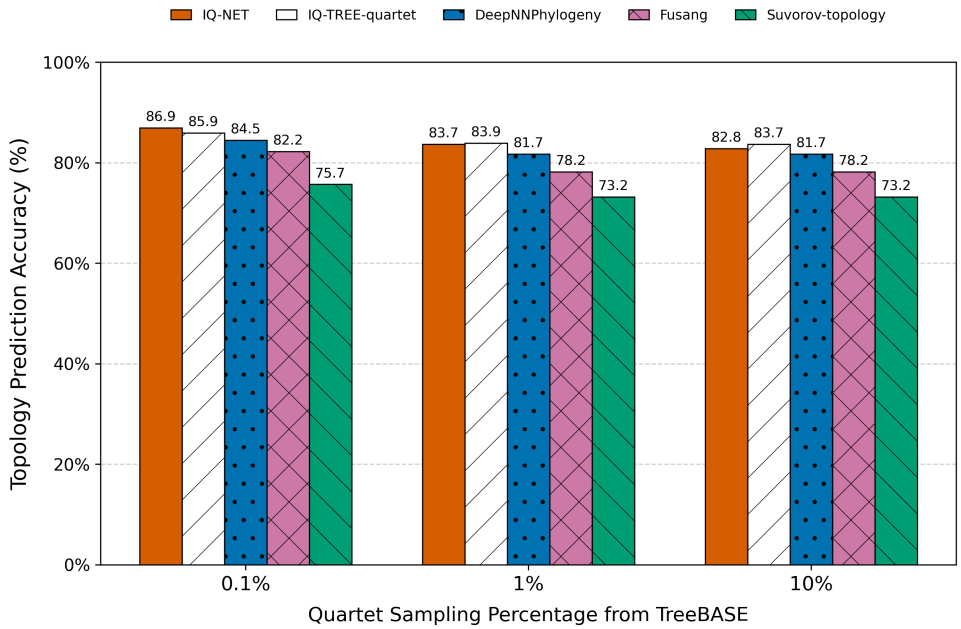

Figure 8: The topology prediction accuracies of IQ-NET, IQ-TREE-quartet, Fusang, DeepNNPhylogeny, and Suvorov-topology on the independent TreeBASE dataset. .

3.2 IQ-NET Demonstrates Strong Generalisation on the Independent TreeBASE Dataset

IQ-NET achieves topology prediction accuracy on the TreeBASE dataset comparable to that of IQ-TREE-quartet. On 1% TreeBASE dataset, IQ-NET obtains an accuracy of 83.7%, compared to 83.9% for IQ-TREE-quartet, while DeepNNPhylogeny, Fusang, and Suvorov-topology yield lower accuracies of 81.7%, 78.2%, and 73.2%, respectively. Detailed results for the 0.1%, 1%, and 10% subsets are provided in Figure 8. These results highlight the ability of IQ-NET in generalising to an empirical dataset independent of its training data.

3.3 IQ-NET + ASTRAL Recovers Turtle Phylogeny Consistent with Existing Studies

We compared species trees inferred using several approaches: (i) **IQ-NET + ASTRAL**, (ii) **concatenation-based analysis with IQ-TREE**, (iii) **ASTRAL** using gene trees inferred with IQ-TREE and ModelFinder, and (iv) **ASTRAL** using gene trees inferred with IQ-TREE and MixtureFinder. Here, ModelFinder selects the best-fit single substitution model for each gene alignment (Kalyaanamoorthy et al., 2017), whereas MixtureFinder estimates a mixture of multiple substitution components, which better accounts for evolutionary heterogeneity across sites (Ren et al., 2025).

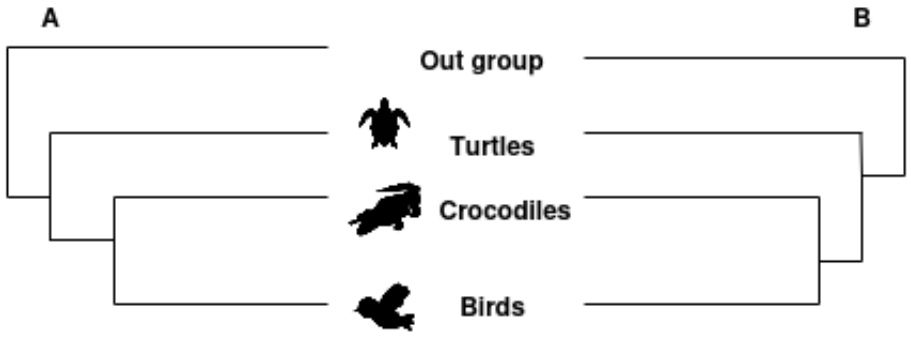

Figure 9: (A) Accepted tree for placement of turtles as sister clade to birds and crocodiles(Chiari et al., 2012). (B) Tree constructed under methods of IQ-NET+ASTRAL after removing third codon positions, ASTRAL using gene trees estimated with IQ-TREE and MixtureFinder.

When all three codon positions were included, the IQ-NET + ASTRAL approach recovered turtles as the sister group to crocodiles, with this clade forming a sister group to birds (Suppl. Figure S6B). This topology was also recovered by the concatenation-based method and by ASTRAL when using gene trees inferred under site-homogeneous models with ModelFinder.

To further examine the effect of codon positions, we repeated the analysis after removing the third codon position from every gene as recommended by Chiari et al. (2012). Quartet trees were estimated with IQ-NET from the reduced alignments, and a species tree was subsequently inferred with ASTRAL. Under this setting, turtles were placed as the sister clade to the combined group of birds and crocodiles (Figure 9B). This topology is congruent with the results obtained from ASTRAL using gene trees estimated with MixtureFinder, as well as with the conclusions of the original study (Chiari et al., 2012). Notably, subsampling only 10% of the possible quartet trees was sufficient to recover this relationship, highlighting the efficiency of the IQ-NET + ASTRAL approach.

4 Discussion

We present IQ-NET, an end-to-end machine learning framework for quartet tree reconstruction from DNA alignments. Unlike traditional likelihood-based approaches that rely on explicit substitution models, IQ-NET learns evolutionary relationships directly from empirical data, enabling phylogenetic inference from gapped alignments without substitution model assumptions. Existing machine learning methods are typically trained on simulated data, which can limit their ability to generalise to real-world datasets - a limitation recently highlighted by Zhu et al. (2025) and Braichenko et al. (2025). IQ-NET addresses this by training

and testing on empirical alignments from the EvoNAPS database, covering diverse taxa including microbes, fungi, plants, birds, and mammals.

This design is reflected in IQ-NET's strong performance. IQ-NET achieves superior topology prediction accuracy (82.3%) compared with existing machine learning methods including DeepNNPhylogeny (79.4%), Fusang (76.6%), and Suvorov-topology (57.7%), and generalises well to the independent TreeBASE dataset. Beyond topology inference, IQ-NET jointly estimates branch lengths - a capability absent in most existing machine learning approaches (Suvorov et al., 2020; Zou et al., 2020; Wang et al., 2023). The only comparable method (Suvorov and Schrider, 2024) relies on multiple networks each trained under specific simulated conditions, limiting its applicability to empirical data. In contrast, IQ-NET provides a single, general-purpose model applicable to real data. Furthermore, by adopting a symmetry-preserving architecture (Tang et al., 2024), IQ-NET produces consistent predictions regardless of input sequence order. Excluding training time, IQ-NET reconstructed over 50,000 trees in just 6.7 minutes, compared with 2.67 hours for IQ-TREE - a 24-fold speedup.

Despite these advantages, IQ-NET has several limitations that should be acknowledged. First, IQ-NET is designed for quartet trees and cannot directly infer larger phylogenies. Although we demonstrate its integration with ASTRAL for species tree reconstruction, this requires an intermediate quartet sampling and aggregation step. Second, IQ-NET currently supports only DNA alignments. Many phylogenomic studies rely on protein or other molecular data types, and extending IQ-NET to these would substantially broaden its applicability. Third, as with all machine learning methods, IQ-NET's performance depends on the quality and diversity of its training data. Although EvoNAPS is extensive, certain evolutionary scenarios - such as extreme rate heterogeneity, very short internal branches, or high substitution saturation - may be underrepresented, potentially affecting accuracy. Additionally, complex evolutionary processes such as incomplete lineage sorting and introgression cannot be fully presented by a single phylogenetic tree, and IQ-NET currently provides no mechanism for detecting or quantifying such signals.

These limitations suggest several directions for future work. First, while we demonstrate integration with ASTRAL, IQ-NET could also be combined with other quartet puzzling approaches (Strimmer and von Haeseler, 1996; Schmidt et al., 2002) to scale reconstruction to larger trees. More broadly, alternative machine learning architectures offer promising directions for extending IQ-NET beyond quartet trees. Transformer-based models, for instance, are not constrained by fixed input sizes and have demonstrated strong performance in biological sequence analysis; adapting such architectures may enable direct inference of larger phylogenies. Second, extending IQ-NET to protein alignments represents an important step toward broader applicability in phylogenomic studies. Third, the accuracy

and generalisation of IQ-NET may be further improved by expanding training to larger and more diverse empirical datasets, particularly those representing underrepresented evolutionary scenarios. Additionally, employing ensemble learning - combining predictions from multiple independently trained models - could further enhance robustness and stability. Fourth, since IQ-NET outputs scores for all three quartet topologies, these values could potentially serve as support measures, offering a foundation for detecting signals of incomplete lineage sorting or introgression, or for constructing phylogenetic networks (Huson et al., 2011) that explicitly represent conflicting evolutionary signals. However, systematic validation is required to confirm this utility. Fifth, while this study focuses on methodological development, with validation on two empirical datasets (TreeBASE and Turtle), broader empirical evaluation across diverse biological datasets will be important to further establish the practical utility of IQ-NET. Finally, integrating IQ-NET into IQ-TREE would make it accessible to the broader phylogenetics community, facilitating its adoption in routine phylogenetic analyses.

Data Availability

All source code and scripts used in the development of IQ-NET, including those for data generation, training, testing, and the pre-trained neural networks, are publicly available at `https://github.com/Clipper1331757/IQ_Net/`.

The data underlying this study are available in the Supplementary Material and the Zenodo Repository, at `https://zenodo.org/records/19582620`.

CRediT authorship contribution statement

Chen Yang: Methodology, Data curation, Software, Formal analysis, Validation, Visualization, Writing - original draft. **Zixin Zhuang**: Methodology, Software, Validation, Writing - review and editing. **Piyumal Demotte**: Formal analysis, Visualization, Writing - original draft. **Cuong Cao Dang**: Formal analysis, Writing - original draft. **Le Sy Vinh**: Conceptualization, Supervision, Writing - review and editing. **Bui Quang Minh**: Conceptualization, Supervision, Writing - review and editing. **Nhan Ly-Trong**: Methodology, Supervision, Visualization, Writing - review and editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in

this paper.

Acknowledgements

This research was undertaken with the assistance of resources and services from the National Computational Infrastructure (NCI), which is supported by the Australian Government. We thank Professor Robert Lanfear, Professor Nick Goldman, Dr. Thomas Wong, Dr. Nicola De Maio, and Hashara Kumarasinghe for their valuable suggestions and discussions. We used Claude and ChatGPT to assist in spelling checks and grammar corrections during the preparation of this manuscript.

Funding

This work was supported by a Chan-Zuckerberg Initiative grant for open-source software for science (EOSS4-0000000312 to B.Q.M.) and the Vietnam National Foundation for Science and Technology Development (102.05-2025.65 to C.C.D, L.S.V).

References

Akiba T, Sano S, Yanase T, et al (2019) Optuna: A next-generation hyperparameter optimization framework. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Association for Computing Machinery, New York, pp 2623–2631, https://doi.org/10.1145/3292500.3330701

Attwood SW, Hill SC, Aanensen DM, et al (2022) Phylogenetic and phylodynamic approaches to understanding and combating the early sars-cov-2 pandemic. Nature Reviews Genetics 23(9):547–562. https://doi.org/10.1038/s41576-022-00483-8

Braichenko S, Borges R, Kosiol C (2025) Phylogenetic methods meet deep learning. Genome Biology and Evolution 17(10):evaf177

Chiari Y, Cahais V, Galtier N, et al (2012) Phylogenomic analyses support the position of turtles as the sister group of birds and crocodiles (archosauria). BMC Biology 10:65. https://doi.org/10.1186/1741-7007-10-65

Delsuc F, Brinkmann H, Philippe H (2005) Phylogenomics and the reconstruction of the tree of life. Nature Reviews Genetics 6(5):361–375. https://doi.org/10.1038/nrg1603

Dombrowski N, Williams TA, Sun J, et al (2020) Undinarchaeota illuminate dpann phylogeny and the impact of gene transfer on archaeal evolution. Nature Communications 11(1):3939

Drummond AJ, Rambaut A (2007) Beast: Bayesian evolutionary analysis by sampling trees. BMC Evolutionary Biology 7:214. https://doi.org/10.1186/1471-2148-7-214

Felsenstein J (1981) Evolutionary trees from dna sequences: a maximum likelihood approach. Journal of molecular evolution 17(6):368–376

Felsenstein J (2004) Inferring Phylogenies. Sinauer Associates, Inc., Sunderland, Massachusetts, USA

Glorot X, Bordes A, Bengio Y (2011) Deep sparse rectifier neural networks. In: Gordon G, Dunson D, Dudk M (eds) Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, Proceedings of Machine Learning Research, vol 15. PMLR, Fort Lauderdale, Florida, USA, pp 315–323

Gómez-Carballa A, Bello X, Pardo-Seco J, et al (2020) Mapping genome variation of sars-cov-2 worldwide highlights the impact of covid-19 super-spreaders. Genome Research 30(10):1434–1448. https://doi.org/10.1101/gr.266221.120

Goodfellow I, Bengio Y, Courville A (2016) Softmax units for multinoulli output distributions. In: Deep Learning. MIT Press, Cambridge, Massachusetts, USA, chap 6.2.2.3, p 180–183

Grenfell BT, Pybus OG, Gog JR, et al (2004) Unifying the epidemiological and evolutionary dynamics of pathogens. Science 303(5656):327–332. https://doi.org/10.1126/science.1090727

Guindon S, Gascuel O (2003) A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood. Systematic Biology 52(5):696–704. https://doi.org/10.1080/10635150390235520

Hodcroft EB, Zuber M, Nadeau S, et al (2021) Spread of a sars-cov-2 variant through europe in the summer of 2020. Nature 595:707–712. https://doi.org/10.1038/s41586-021-03677-y

Huelsenbeck JP, Ronquist F (2001) Mrbayes: Bayesian inference of phylogenetic trees. Bioinformatics 17(8):754–755

Huson DH, Rupp R, Scornavacca C (2011) Phylogenetic Networks: Concepts, Algorithms and Applications. Cambridge University Press

Izquierdo-Carrasco F, Stamatakis A (2011) Computing the phylogenetic likelihood function out-of-core. In: 2011 IEEE International Symposium on Parallel and Distributed Processing Workshops and PhD Forum. IEEE, Anchorage, Alaska, USA, pp 444–451, https://doi.org/10.1109/IPDPS.2011.185

Jermiin LS, Jayaswal V, Ababneh FM, et al (2016) Identifying optimal models of evolution. In: Bioinformatics: Volume I: Data, Sequence Analysis, and Evolution. Springer, New York, USA, p 379–420

Kalyaanamoorthy S, Minh BQ, Wong TK, et al (2017) ModelFinder: Fast model selection for accurate phylogenetic estimates. Nat Methods 14(6):587–589. https://doi.org/10.1038/nmeth.4285

Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. https://doi.org/10.48550/arXiv.1412.6980, preprint at https://arxiv.org/abs/1412.6980

Kuhner MK, Felsenstein J (1994) A simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. Molecular Biology and Evolution 11(3):459–468. https://doi.org/10.1093/oxfordjournals.molbev.a040126

Kulikov N, Derakhshandeh F, Mayer C (2024) Machine learning can be as good as maximum likelihood when reconstructing phylogenetic trees and determining the best evolutionary model on four taxon alignments. Molecular Phylogenetics and Evolution 200:108181. https://doi.org/10.1016/j.ympev.2024.108181

Leuchtenberger AF, Crotty SM, Drucks T, et al (2020) Distinguishing felsenstein zone from farris zone using neural networks. Molecular Biology and Evolution 37(12):3632–3641. https://doi.org/10.1093/molbev/msaa164

Li T, Liu D, Yang Y, et al (2020) Phylogenetic supertree reveals detailed evolution of sars-cov-2. Scientific Reports 10:22366. https://doi.org/10.1038/s41598-020-79484-8

Mai U, Mirarab S (2017) Treeshrink: efficient detection of outlier tree leaves. In: RECOMB International Workshop on Comparative Genomics, Springer, pp 116–140

Minh BQ, Schmidt HA, Chernomor O, et al (2020) Iq-tree 2: new models and efficient methods for phylogenetic inference in the genomic era. Molecular biology and evolution 37(5):1530–1534

Mirarab S, Reaz R, Bayzid MS, et al (2014) Astral: genome-scale coalescent-based species tree estimation. Bioinformatics 30(17):i541–i548. https://doi.org/10.1093/bioinformatics/btu462

Nesterenko L, Blassel L, Veber P, et al (2025) Phyloformer: fast, accurate, and versatile phylogenetic reconstruction with deep neural networks. Molecular Biology and Evolution 42(4):msaf051. https://doi.org/10.1093/molbev/msaf051

Piel WH, Donoghue MJ, Sanderson MJ (2002) Treebase: a database of phylogenetic information. In: Proceedings of the 2nd International Workshop of Species 2000. National Institute for Environmental Studies, Tsukuba, Japan, pp 41–47

Ren H, Wong TK, Minh BQ, et al (2025) MixtureFinder: Estimating DNA Mixture Models for Phylogenetic Analyses. Mol Biol Evol 42(1):1–13. https://doi.org/10.1093/molbev/msae264, URL https://doi.org/10.1093/molbev/msae264

Ronquist F, Huelsenbeck JP (2003) Mrbayes 3: Bayesian phylogenetic inference under mixed models. Bioinformatics 19(12):1572–1574. https://doi.org/10.1093/bioinformatics/btg180

Schmidt HA, Strimmer K, Vingron M, et al (2002) Tree-puzzle: maximum likelihood phylogenetic analysis using quartets and parallel computing. Bioinformatics 18(3):502–504. https://doi.org/10.1093/bioinformatics/18.3.502

Scornavacca C, Belkhir K, Lopez J, et al (2019) Orthomam v10: scaling-up orthologous coding sequence and exon alignments with more than one hundred mammalian genomes. Molecular Biology and Evolution 36(4):861–862. https://doi.org/10.1093/molbev/msz015

Smith ML, Hahn MW (2023) Phylogenetic inference using generative adversarial networks. Bioinformatics 39(9). https://doi.org/10.1093/bioinformatics/btad543

Stamatakis A (2006) Phylogenetic models of rate heterogeneity: a high performance computing perspective. In: Proceedings of the 20th IEEE International Parallel and Distributed Processing Symposium. IEEE, Rhodes, Greece, p 8, https://doi.org/10.1109/IPDPS.2006.1639535

Stamatakis A (2014) Raxml version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. Bioinformatics 30(9):1312–1313. https://doi.org/10.1093/bioinformatics/btu033

Strimmer K, von Haeseler A (1996) Quartet puzzling: a quartet maximum-likelihood method for reconstructing tree topologies. Molecular Biology and Evolution 13(7):964. https://doi.org/10.1093/oxfordjournals.molbev.a025664

Suvorov A, Schrider DR (2024) Reliable estimation of tree branch lengths using deep neural networks. PLoS Computational Biology 20(8):e1012337. https://doi.org/10.1371/journal.pcbi.1012337

Suvorov A, Hochuli J, Schrider DR (2020) Accurate inference of tree topologies from multiple sequence alignments using deep learning. Systematic Biology 69(2):221–233. https://doi.org/10.1093/sysbio/syz060

Tang X, Zepeda-Nuñez L, Yang S, et al (2024) Novel symmetry-preserving neural network model for phylogenetic inference. Bioinformatics Advances 4(1):vbae022. https://doi.org/10.1093/bioadv/vbae022

Turakhia Y, Thornlow B, Hinrichs AS, et al (2021) Ultrafast sample placement on existing trees (usher) enables real-time phylogenetics for the sars-cov-2 pandemic. Nature Genetics 53(6):809–816. https://doi.org/10.1038/s41588-021-00862-7

Wang Z, Sun J, Gao Y, et al (2023) Fusang: a framework for phylogenetic tree inference via deep learning. Nucleic Acids Research 51(20):10909–10923. https://doi.org/10.1093/nar/gkad805

Wehbi S, Wheeler A, Morel B, et al (2024) Order of amino acid recruitment into the genetic code resolved by last universal common ancestors protein domains. Proceedings of the National Academy of Sciences 121(52):e2410311121

Whelan S, de Bakker PI, Quevillon E, et al (2006) Pandit: an evolution-centric database of protein and associated nucleotide domains with inferred trees. Nucleic Acids Research 34:D327–D331. https://doi.org/10.1093/nar/gkj087

Williamson K, Eme L, Baños H, et al (2025) A robustly rooted tree of eukaryotes reveals their excavate ancestry. Nature 640(8060):974–981

Yang Z, Rannala B (2012) Molecular phylogenetics: principles and practice. Nature Reviews Genetics 13:303–314. https://doi.org/10.1038/nrg3186

Zhu Y, Li Y, Li C, et al (2025) A critical evaluation of deep-learning based phylogenetic inference programs using simulated datasets. Journal of Genetics and Genomics 52(5):714–717. https://doi.org/10.1016/j.jgg.2025.01.006

Zou Z, Zhang H, Guan Y, et al (2020) Deep residual neural networks resolve quartet molecular phylogenies. Molecular Biology and Evolution 37(5):1495–1507. https://doi.org/10.1093/molbev/msz307

Supplementary

No	Permutation
1	$[S_A, S_C, S_D, S_B]$
2	$[S_A, S_D, S_B, S_C]$
3	$[S_B, S_C, S_A, S_D]$
4	$[S_B, S_D, S_C, S_A]$
5	$[S_C, S_A, S_B, S_D]$
6	$[S_C, S_B, S_D, S_A]$
7	$[S_D, S_A, S_C, S_B]$
8	$[S_D, S_B, S_A, S_C]$

Table S1: List of the eight permutations of a quartet alignment that change the underly topology.

Hyperparameter	Description	Search Range	Sampling Strategy	Best Setting
Learning Rate	Step size for weight updates during training	[e−4, 2e−3]	Log-uniform	7.844e−4
Learning Rate Decay	Factor to reduce the learning rate over epochs	[0.85, 1.0]	Uniform	0.9044
Dropout	Fraction of neurons randomly deactivated to prevent overfitting	[0.0, 0.3]	Uniform	0.08314
β_1(Adam)	Exponential decay rate for first moment estimates	[0.85, 0.95]	Uniform	0.8687
β_2(Adam)	Exponential decay rate for second moment estimates	[0.9, 0.999]	Uniform	0.9976
Batch Size	Number of samples processed per training iteration	{8, 16, 32, 64, 128, 256, 512}	Categorical	128

Table S2: Hyperparameter search ranges and the best setting for the tree topology classifier.

Hyperparameter	Description	Search Range	Sampling Strategy	Best Setting
Learning Rate	Step size for weight updates during training	[e−5, e−2]	Log-uniform	6.297e−4
Learning Rate Decay	Factor to reduce the learning rate over epochs	[0.85, 1.0]	Uniform	0.9667
Dropout	Fraction of neurons randomly deactivated to prevent overfitting	[0.0, 0.3]	Uniform	0.05618
Weight Decay	L_2 regularization to prevent large weights	[e−6, e−3]	Log-uniform	6.297e−4
β_1(Adam)	Exponential decay rate for first moment estimates	[0.85, 0.95]	Uniform	0.9281
β_2(Adam)	Exponential decay rate for second moment estimates	[0.9, 0.999]	Uniform	0.9717
Batch Size	Number of samples processed per training iteration	{8, 16, 32, 64, 128, 256, 512}	Categorical	32

Table S3: Hyperparameter search ranges and the best setting for the branch length regressor.

Metric	Branch Type	IQ-NET	IQ-TREE-quartet
MAE	Internal branch	**0.0322**	0.0388
	External branch	**0.0273**	0.042
MSE	Internal branch	**0.0202**	0.0538
	External branch	**0.0122**	0.0442
MRE	Internal branch	0.7824	**0.6564**
	External branch	0.2519	**0.2286**
BSD	All branches	**0.0946**	0.1485

Table S4: Branch length estimation errors of IQ-NET and IQ-TREE-quartet. For IQ-NET and IQ-TREE-quartet, only trees with correct topology prediction were counted. Bold numbers denote the best results.

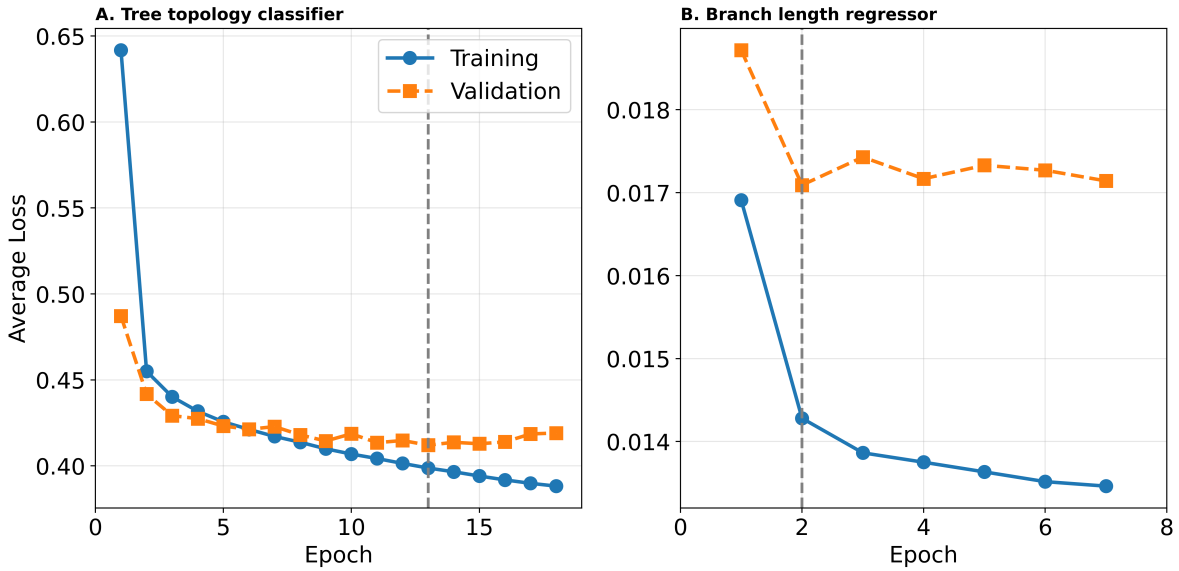

Figure S1: The training and validation loss of (A) the tree topology classifier; and (B) the branch length regressor. The regressors for internal and external branches were jointly trained, resulting in a single loss curve. The black dashed vertical line indicates the epoch at which the best parameters were obtained (epoch 13 for the topology classifier and epoch 2 for the branch length regressor) before training was terminated by early stopping.

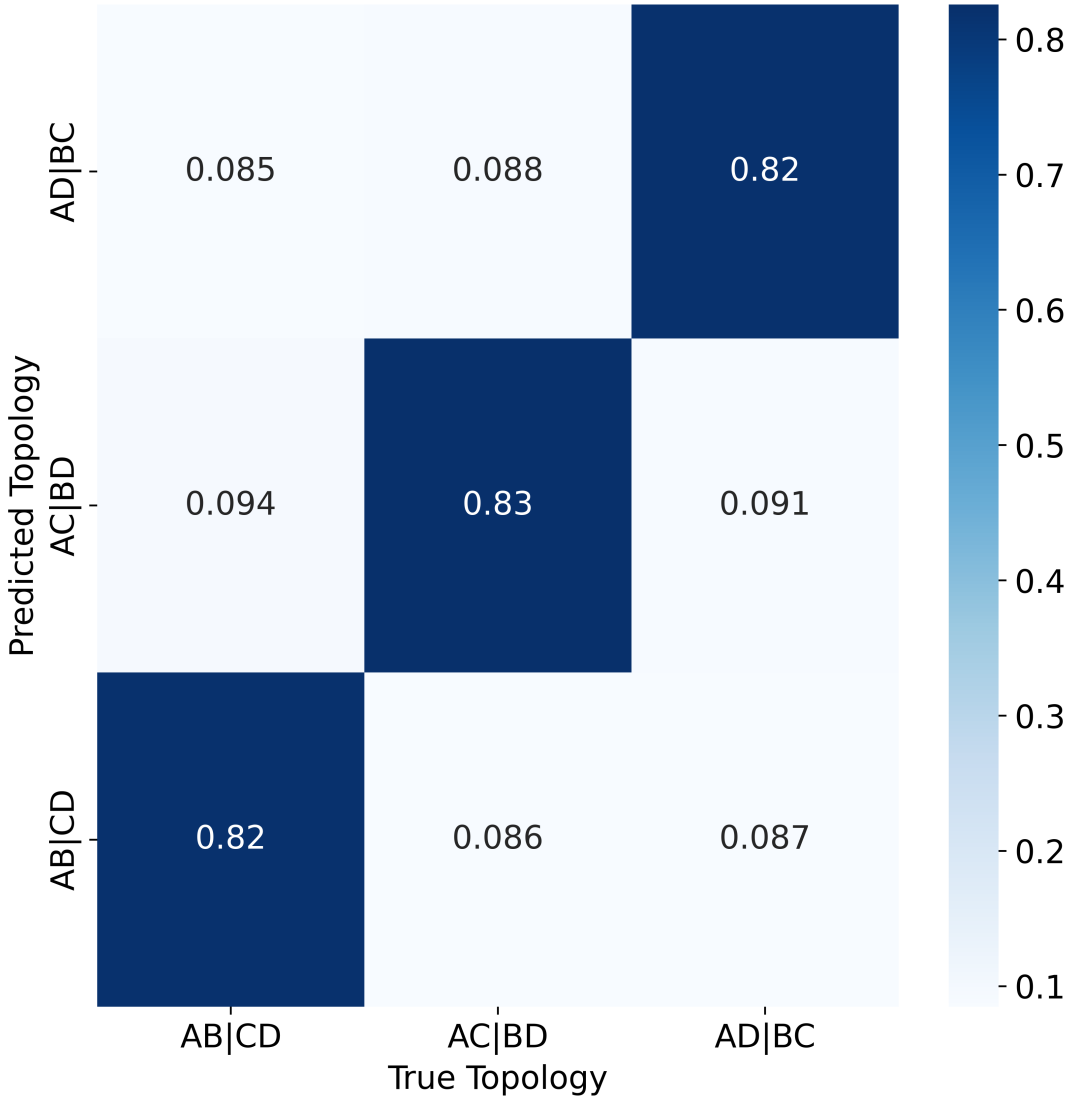

Figure S2: Confusion Matrix of True vs Predicted tree topologies.

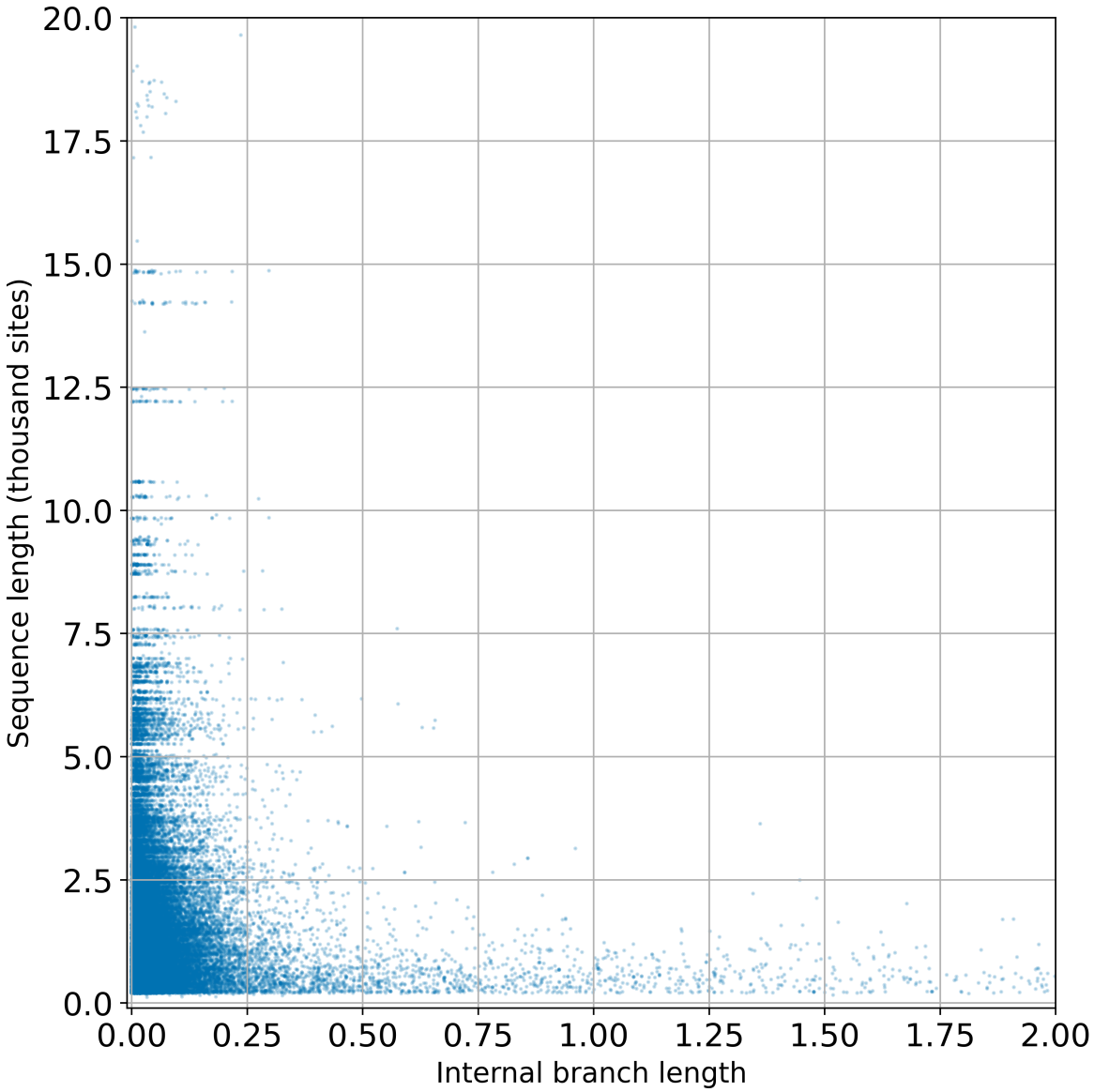

Figure S3: Distribution of Internal branch lengths vs Sequence length across testing samples.

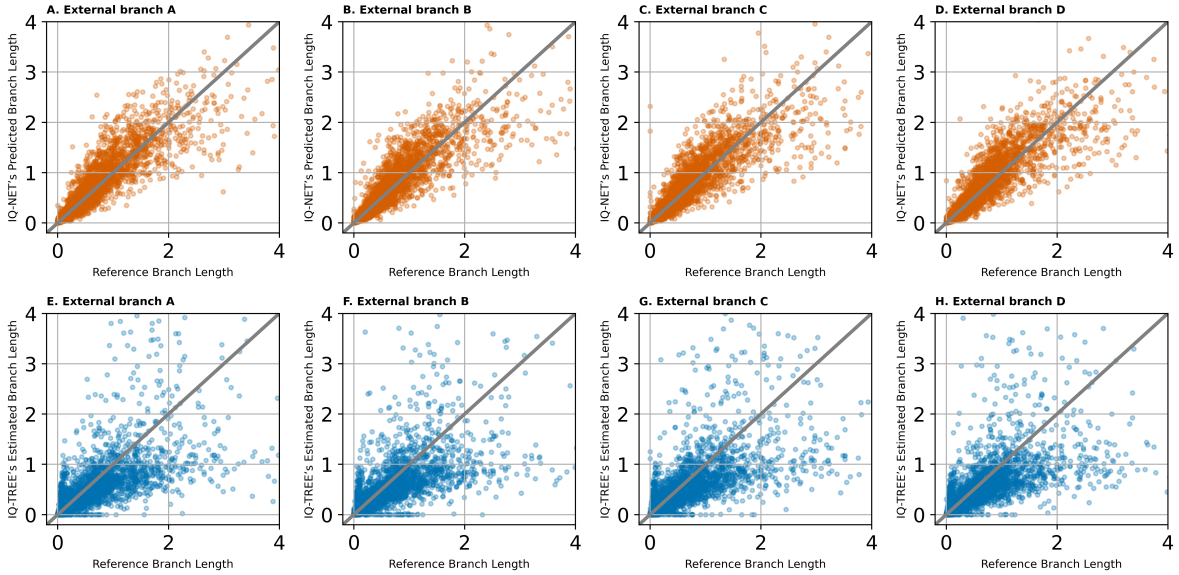

Figure S4: Scatter plots of reference vs predicted lengths of the four external branches estimated by IQ-NET (Panels A, B, C, and D) and IQ-TREE-quartet (Panels E, F, G, and H), where reference branch lengths are those of the quartet trees sampled from larger maximum likelihood trees inferred by IQ-TREE on the original multi-taxon alignments.

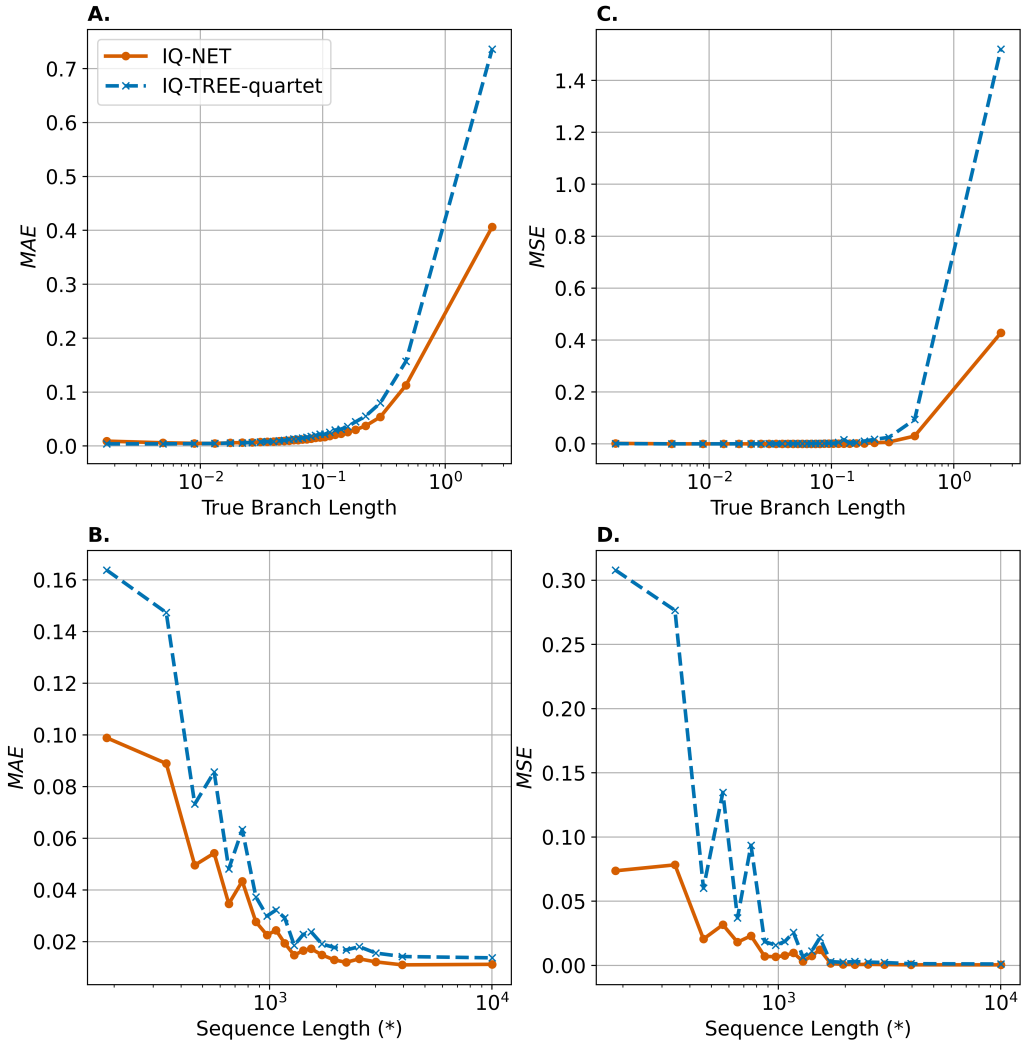

Figure S5: Mean absolute error (MAE) and Mean square error (MSE) of branch length estimated by IQ-NET and IQ-TREE-quartet. (A) MAE vs branch length; (B) MAE vs sequence length; (C) MSE vs branch length; (D) MSE vs sequence length. (*) Sequence length excludes fully gapped sites, which contain no evolutionary signal.

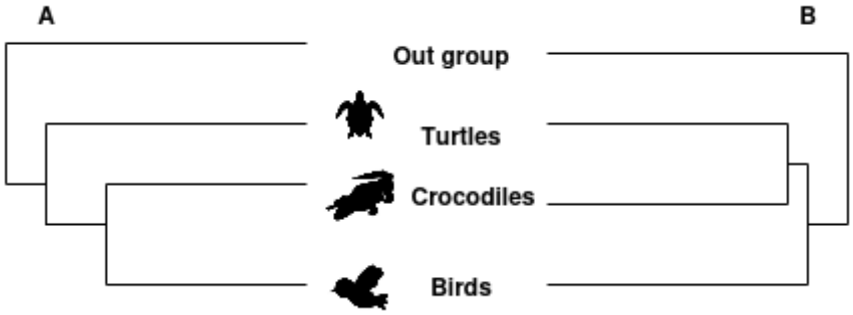

Figure S6: (A) Accepted tree for placement of turtles as sister clade to birds and crocodiles (Chiari et al., 2012). (B) Tree constructed under methods of IQ-NET+ASTRAL, IQ-TREE with concatenated MSA, ASTRAL using gene trees estimated with IQ-TREE and ModelFinder.