Deep Neighbor Information Learning From Evolution Trees for Phylogenetic Likelihood Estimates

CHENG LING1, WENHAO CHENG1, HAOU ZHANG2, HANHAO ZHU3, AND HUA ZHANG4

1Department of Computer Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China
2School of Information Engineering, Zhejiang Ocean University, Zhoushan 316022, China
3Institute of Marine Science and Technology, Zhejiang Ocean University, Zhoushan 316022, China
4School of Naval Architecture and Maritime, Zhejiang Ocean University, Zhoushan 316022, China

Corresponding author: Haoyu Zhang (haoyu19871202@163.com)

This work was supported in part by the Natural Science Foundation of China under Grant 61602026, and in part by the Stable Supporting Fund of Acoustic Science and Technology Laboratory under Grant JCKYS2020604SSS011.

ABSTRACT Likelihood probability based phylogenetic analysis approaches have contributed to impressive advances in minimizing the variance of estimating the evolutionary parameters. However, their actual applications are greatly limited due to the very time-consuming calculations of Conditional Likelihood Probabilities (CLPs). Accurately and quickly obtaining the likelihoods of massive tree samples can facilitate phylogenetic analysis process. Inspired by recent advance of machine learning techniques that greatly improve the performance of many related prediction tasks, this study proposes a Random Forest (RF) based learning and prediction approach, called NeoPLE. The approach initially learns the deep neighbor information between nodes from the topology representations of evolution trees, integrates likelihood information from these trees, and trains a non-linear prediction model. Instead of having to depend on the recursive calculations of the CLPs of tree nodes, NeoPLE transfers the process to a prediction by the trained model, thus the likelihood estimates become irrelevant with the calculations of CLPs. In terms of performance improvement, speedup factors ranging from 2.1 to 3.5X can be achieved on the analysis of realistic data sets. Moreover, NeoPLE is very suitable to handle the data sets having relatively large number of alignment sites, the factor of up to 27.5X can be achieved on the analysis of simulated data sets. In addition, NeoPLE is robust against a wide range of choices of evolutionary models and is ready to integrate in more phylogenetic inference tools. This study fills in the gaps of phylogenetic analysis using a machine learning approach in feature representation and likelihood prediction of evolution trees, which has not been reported in literatures.

INDEX TERMS Likelihood probability, phylogenetic analysis, machine learning, prediction model, likelihood prediction, evolution trees.

I. INTRODUCTION

Evolutionary research at the molecular level addresses two major issues: rebuilding the evolutionary relationships between species and understanding the dynamics and mechanisms of evolutionary processes [1]–[4]. The former belongs to the field of systematics, and traditionally uses morphological traits and fossils to carry out research [5]–[7]. Because of the wide application and easy access of molecular data, it has become the most commonly used data type in reconstructing phylogenetic relationships of most species groups [2], [8]. The problem of the latter molecular evolution mechanism is divided into two aspects. One is to estimate the replacement rate of nucleotides and amino acids. On the other hand, sequence data is used to detect mutation and select models [9], [10].

At present, the genetic sequence data are accumulating explosively. However, due to the great improvement of computer software and hardware, the application of complex statistical methods in solving biological problems has also made great progress, so the two main problems in molecular level evolution research have made amazing progress [11]–[16]. There are indications that this progress, especially in terms of data generation, will certainly continue. In other words, phylogenetic analysis has entered the genomic era, and it has become common practice to analyze...
large data sets containing hundreds or even thousands of species or sequences. However, phylogenetic analysis needs to construct phylogenetic trees [5], [17]–[19]. At the moment, the commonly used methods for constructing evolutionary trees are distance method [20], maximum parsimony method [21], maximum likelihood method [22], [23] and Bayesian method [24], [28]. The latter two methods harness empirical evolutionary models and have better statistical basis, as the minimum variance of parameter statistics can be obtained when sampling size becomes larger. As long as a reasonable and correct alternative model is adopted, a considerable evolution result can be deduced [29], [30]. Nevertheless, the disadvantages are that both methods need to calculate the Conditional Likelihood Probabilities (CLPs) of tree nodes one by one, while the computational cost is very expensive, resulting in undesirably long execution time.

Machine learning methods are suitable for areas with large data, noise patterns, and lack of unified theory [31]–[33]. The basic idea of machine learning is to learn theory from massive data automatically by reasoning, pattern matching or sample learning [32], [34]. Inspired by recent advance of machine learning techniques that greatly improve the performance of many related prediction tasks, this study proposes a Random Forest (RF) based learning and prediction approach, called NeoPLE. The approach initially learns the deep neighbor information between nodes from the topology representations of evolutionary trees, integrates likelihood information from these trees, and trains a non-linear prediction model. Instead of having to reply on the recursive calculations of CLPs of sampling trees, NeoPLE transfers the process to a prediction by a well-trained model, thus the likelihood estimates become irrelevant with the calculations of CLPs. Besides generating the same consensus trees, the speedup factors ranging from 2.1 to 3.5X can be achieved by NeoPLE on the analysis of realistic data sets comparing with MrBayes - a widespread phylogenetic inference tool. Moreover, NeoPLE is very suitable to handle the data sets having relatively large number of alignment sites, as it makes the time complexity of estimating CLPs be O(1) if only increasing the number of alignment sites of data sets. The speedup factor of up to 27.5X can be achieved on the analysis of simulated data sets containing 54 taxa and 8,000 sites. In addition, NeoPLE is robust against a wide range of choices of evolutionary models, as JC69, K80, HKY and GTR models are successfully established in the approach. This study fills in the gaps of phylogenetic analysis using a machine learning approach in feature representation and likelihood prediction of evolution trees, which has not been reported in literatures.

II. METHOD

A. OVERVIEW OF OUR METHODOLOGY

The Bayesian inference of phylogeny employs Markov Chain Monte Carlo (MCMC) process to approximate the posterior probabilities of phylogenetic trees, with the Bayes theorem, prior probabilities, stochastic starting-states, calculations of acceptance probability and others. Tree re-arrangements are harnessed in the process recursively for the purpose of generating new combinations of branches and nodes. First, we analyze the essence of the process through introducing the key procedures in likelihood estimates. Next, we explore the associations between tree nodes, branches and the likelihood by decomposing a sampling tree. We then illustrate how to obtain the evolution time representing the features between arbitrary two leaf nodes. Based on the feature matrix representation of the combinations of nodes and branches, a novel adaptive regressor based on Random Forest (RF) algorithm is then established to predict the likelihood of new combined trees.

B. PHYLOGENETIC LIKELIHOOD ESTIMATE DECOMPOSITION

The MCMC process tries to improve the likelihood probability by constantly modifying the branch length or some other evolution parameters. Fig. 1A illustrates two consecutive trees in a sampling chain. Mi+1 is generated from Mi through rearranging the branch length between node 7 and 9 of Mi. In essence, once a branch length representing the evolution time between nodes is updated, the evolutionary models, such as JC69 [34], [35], [35]–[37], K80 [36], [38], HKY [39] and GTR [40]–[42], generate new transition probabilities to modify the evolution process between the nodes. The likelihood probabilities of the ancestral node, i.e. node 9, are then re-calculated by a likelihood function using these transition probabilities in conjunction with the likelihood probabilities of the observable or non-observable states of the descendant nodes. More precisely, a non-observable state means the state is not certain because multiple possible states existing in nodes of extinction (non-leaf nodes), whereas an observable state means the state is certain which can be observed from input sequences (leaf nodes). Fig. 1B depicts the associations between the components, i.e., branch, node and likelihood, of a tree. The node-branch association denotes the evolution time. The node-node association denotes the evolutionary origin and direction. The root-likelihood association reveals the matching degree of the tree and the given model. As depicted in Fig. 1C, for the node that evolves into two non-leaf nodes, the likelihood probability of a possible state of the node, e.g. a single nucleotide residue ‘A’, is the product of the probability that ‘A’ evolving into the left and right descendant nodes. For each evolution branch, the probability of the state is the summation of the probabilities that the state evolves into all possible states of the descendant node. Since the likelihood probability is 100% for the observed state and 0% for the rest, the probability of the state only depends on the transition probability between the state to the observed state if the descendant is a leaf node.

Let X denote the input data, x denote the i-th alignment site of X, τ denote the tree topology of Mi+1, Sk represent a residue state of the node under estimation, Sr(i(k)) and Sσj(k) represent the left and right descendant nodes, P represent the transition probabilities between nucleotide states, θ denote the parameters of the evolutionary model calculating P, a comprehensive pruning algorithm [43]–[45] is applied to
calculate the likelihood probability of Sk for each arbitrary internal node:

\[
L^i_k (x|\tau, v, \theta) = \left( \sum_{S_{\sigma(k)}} p_{S_k S_{\sigma(k)}} \left( V_{\sigma(k)}(k, \theta) \right) L^i_{S_{\sigma(k)}} \right) \prod_{S_{\delta(j)}} p_{S_j S_{\delta(j)}} \left( V_{\delta(j)}(k, \theta) \right) \tag{1}
\]

where \( p(v, \theta) \) represents the transition probabilities between ancestral and descendant nodes over a branch length of v under the specified evolutionary parameters \( \theta \). Let \( \pi \) denote the prior base probabilities of residues, the overall likelihood of the tree for site \( x \) is then obtained by (2).

\[
L^i (x|\tau, v, \theta) = \sum_{S_k} \pi_{S_k} L^i_k
\tag{2}
\]

Fig. 1D depicts the likelihood estimation flow. The likelihood probability of the entire data \( X \) is the product of the probability of each site, hence the log-likelihood ratio is equivalent to the summation of the likelihood probabilities of all sites in \( X \).

\[
\log L = \sum_{i=1}^n \log L^i (x|\tau, v, \theta)
\tag{3}
\]

C. FEATURE MATRIX PRESENTATION

As above, the calculation of the likelihood probabilities depends on the tree topology and the transition probabilities of the related branches. The former determines the relationships between nodes, while the latter depends on the setting of hypothesis parameters in different evolutionary models, in which, the evolution time are most closely related. Thus, the relationships and branches can be selected as the features representing an evolutionary tree which can be used by machine learning algorithms, e.g. training. Fig. 1E then indicates that the evolution time between two nodes can be obtained by accumulating the branch length of all the related tree nodes. For instance, the evolution time between node 1 and 5 is the summation of t1, t9, t13, t14, t11 and t5. Based on this consideration, the feature matrix of each sampled tree can be constructed as shown in Fig. 1F. The value of...
elements in the matrix denotes the distance between leaf nodes. To verify whether the likelihood of new tree samples can be predicted by the distance matrix (Fig. 1F), we extract the upper triangular elements of the matrices that are created from 50 sampling trees, and harness Principal Component Analysis (PCA) method to carry out the dimension reduction on the features so that the trend of the reduced features can be compared with the log likelihood of these samples in the Cartesian coordinate system. As illustrated in Fig. 2A, the very similar trend between the curves of the features and the log likelihoods indicates that there are implied connections between the extracted features and the likelihoods of these sampling trees. The scatter plot in Fig. 2B further demonstrates the strong linear relationships between the feature of decreased dimension and likelihood.

D. CART AND BAGGING

The Classification and Regression Tree (CART) is a tree-based classifier. Based on minimizing Gini coefficient, the construction of CART starts by searching all attributes in the training samples and selecting the features which can split the whole data set into two portions with the lowest impurity among the samples within each subset. Then, the process is carried out recursively for the rest subsets. The first selected feature is deemed as the root node and the division process continues until the subset only contains a single sample. In order to improve the generalization ability, CART needs to be pruned before it is transformed into final decision rules. The bagging algorithm [43], [44] is proposed to overcome over fitting, which harnesses bootstrap strategy to sample data set for multiple weak classifiers. Bootstrap collects a fixed number of samples from training set and put them all back after the sample. In other words, the collected samples may continue to be collected after being put back. A joint vote is performed by all classifiers, and the category voted by the most classifier voting is selected as the final prediction result. Random Forest (RF) algorithm [45]–[49] implements a combination classifier based on CART and Bagging. RF uses CART as the meta classifiers and uses Bagging method to create different training sample sets. Moreover, RF randomly selects features to split the attributes of internal nodes when building a single tree. The attribute splitting is actually to randomly group input features to construct CART based on these features. The combination of Bagging and CART as well as the random feature selection from attribute splitting, makes RF have better noise tolerance and...
C. Ling et al.: Deep Neighbor Information Learning From Evolution Trees for Phylogenetic Likelihood Estimates

FIGURE 3. The modeling process on the data set that is transferred from \( N \) trees; A-B Using bagging method to divide the samples; C Each group of data set is trained using CART algorithm to get a weak classifier.

classification performance. Let \((x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\) denote the samples in training data set \( D \), \( R \) denote the category of these samples, the impurity function of CART for all samples in \( R \) can be calculated by (4). For each split, the optimization target is to search splitting feature and value, i.e. \( j \) and \( s \), that can minimize the impurity function \( L \).

Fig. 3 illustrates the modeling process on the data set that is transferred from \( N \) trees. At first, Bagging is performed on the group of \( N \) samples. The bootstrap sampling results in \( S \) new groups. These new groups contain equal-size samples, but different features. The number of features in new groups is much less the original group. Next, each group of data set is trained using CART algorithm to get a weak classifier. For instance, the process on a group of 5 samples \( \{x_1, x_2, x_3, x_4, x_5\} \) and 3 features \( \{f_1, f_3, f_5\} \) is shown in Fig. 9C. For feature \( f_1 \), the splitting into the group of \( \{x_1, x_2, x_4\} \) and the group of \( \{x_3, x_5\} \) using \( a_{21} \) can minimize the impurity. Similarly, the splitting value for \( f_3 \) and \( f_5 \) are \( a_{33} \) and \( a_{45} \), respectively. By comparing the three impurities, \( f_5 \) and \( a_{45} \) are selected to be the splitting feature and value. As a consequence, the process makes the left derived node contain the samples of \( \{x_1, x_2, x_4\} \) and the right derived node contain the samples of \( \{x_3, x_5\} \).

\[
L = \sum_{X_j \in R} \left( y_j - \frac{1}{n} \sum_{X_j \in R} y_j \right)^2
\]

\[
\arg_{j,s} \min \left[ \sum_{X_j \in R_{1}(j,s)} (y_j - c_1)^2 + \sum_{X_j \in R_{2}(j,s)} (y_j - c_2)^2 \right]
\]

E. TRAINING AND PREDICTION

Currently, there are many kinds of integration libraries implementing the basic machine learning algorithms. The proposed method focuses on neither presenting the detailed principles of these algorithms nor giving the skills of adjusting the parameters when using them. Instead, off-the-shelf python library implementing RF, i.e., RandomForest-Regressor in sklearn, is used. Fig. 4 depicts the implementation flow of prediction. At first, let \((X, Y)\) denote the data set, where \( X \) is the data set composed of the upper triangular elements of \( \sqrt{N} \) square matrices, \( Y \) is the data set composed of the likelihood corresponding to \( \sqrt{N} \) square matrices. The data set \((X, Y)\) is then divided into training set \((X_{\text{train}}, Y_{\text{train}})\) and testing set \((X_{\text{test}}, Y_{\text{test}})\). The training set \((X_{\text{train}}, Y_{\text{train}})\) is input into the Random Forest Regressor for training. Once the training process is completed, the data set in \( X_{\text{test}} \) is predicted and recorded as \( Y_{\text{predict}} \).

F. GOODNESS OF FIT

The goodness of fit is a benchmark to measure the fitting degree of the regression line and the observation value. The statistic for measuring the goodness of fit is the determinable coefficient (also known as the deterministic coefficient), and the maximum value of is 1, and the closer the value of is to 1, the more the regression line fits the observed value. On the contrary, the smaller the value of, the worse the regression line fits the observed value. Let denote the value to be fitted, denote the mean value, denote the fitting value, (6) is used to calculate the Regression Square Sum (RSS), (7) is used to calculate the Error Square Sum (ESS). Since TSS is the
C. Ling et al.: Deep Neighbor Information Learning From Evolution Trees for Phylogenetic Likelihood Estimates

FIGURE 4. The implementation process of model prediction.

FIGURE 5. The fitting curve between the likelihood probabilities that are calculated by likelihood function and the one predicted by NeoPLE. X-axis represents the number of sampling trees, while Y-axis is the likelihood probabilities corresponding to each sample. The red curve stands for real value and the blue one is the predicted value. It can be seen that the blue curve fits the red curve well.

The summation of RSS and ESS, is then calculated by (8). Fig. 5 depicts a fitting curve, where the goodness of fit is 0.9786.

\[
\sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2 \quad (6)
\]

\[
\sum_{i=1}^{n} (y_i - \bar{y})^2 \quad (7)
\]

\[
R^2 = \frac{RSS}{TSS} = \frac{\sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} \quad (8)
\]

III. RESULT

The evaluation of NeoPLE is composed of two aspects. The first one is to evaluate the time cost of NeoPLE comparing MrBayes when analyzing different lengths of data sets. The elapsed time of producing the likelihoods of the same number of tree generations by NeoPLE and MrBayes is defined as \( tp \) and \( tc \), respectively. The efficiency of NeoPLE is then obtained through comparing \( tp \) and \( tc \). The other one is to evaluate the accuracy (\( acc \)) of NeoPLE through employing different evolutionary models (\( model \)), alignment sites (\( nsite \)), and sampling tree generations (\( ngen \)).

A. EXPERIMENTAL DATA SETS

The experimental data sets are composed of realistic nucleotide sequence data sets and simulated nucleotide sequence data sets. The former is obtained directly from the package of MrBayes v3.1.2, where data sets of 12, 17, 29, 37, 54 taxa are included, respectively. The latter is generated by Seq-Gen [50] for the purpose of obtaining the data sets with a fixed number of taxa, e.g., the same number of taxa as the realistic data sets, but different number of alignment sites. For analyzing each data set, \( ngen \) is determined by the average standard deviation of split frequencies representing the degree of similarity between two independent analyses. In the experiments, the sampling chains are stopped when the value was less than 0.01. Hence, \( ngen \) for these data sets is set to 12000, 20000, 50000, 70000, 400000, respectively.

B. PERFORMANCE COMPARISONS

Fig. 6 illustrates the comparisons between \( tc \) and \( tp \) when analyzing realistic data sets. The nucleotide substitution models are set to GTR, K80, JC69 and HKY, respectively. Apparently, \( tc \) is dramatically increased with the increase of the number of taxa, which causes a relatively large consumption of time and hinders the analysis of very large species or sequences to some extent. Precisely, NeoPLE outstrips MrBayes with speedup factors from 2.1 to 3.5 when using GTR model, and the speedup factors are similar for the other three models. Consequently, the experimental results demonstrate that NeoPLE outstrips MrBayes in all cases. Moreover, Fig. 7 illustrates the performance comparisons between \( tc \) and \( tp \) when fixing the evolutionary model to GTR and the number of taxa to \textit{nucle12}, \textit{nucle17}, \textit{nucle29}, \textit{nucle37}, and \textit{nucle54} respectively. Note that the experiments use the simulated data sets with different number of alignment sites ranging from 1,000 to 8,000. As shown in the figure, \( tc \) is significantly increased with the increase of alignment sites. Nonetheless, the result produced by NeoPLE is pleasantly surprised as the time cost is constant no matter how the number of alignment sites changes. For instance, \( tc \) is markedly growing with the increase of the \( nsite \) when the data set is \textit{nucle12} in Fig. 7A, but \( tp \) almost remains the same. The more obvious is that \( tc \) increases rapidly from 495 to 5,300s when increasing the number of taxa to \textit{nucle54} depicted in Fig. 7E, while \( tp \) still remains 222s. Under these circumstances, the time gap between \( tp \) and \( tc \) is amazing.

C. CONSENSUS TREE COMPARISONS

The aim of Phylogenetic analysis is to obtain reasonable molecular evolutionary trees, thus a consensus tree needs to be established at the end of each analysis to verify whether
FIGURE 6. In the case of four nucleotide substitution models, the trend of time complexity $t_p$ and $t_c$ changes with the number of taxa. The x-axis represents the name of taxa. T on the y-axis represents time cost in seconds. The elapsed time of NeoPLE is defined as $t_p$ and the elapsed time to estimate the likelihood using MrBayes is defined as $t_c$. In addition, the titles is different nucleotide substitution models, such as GTR, JC69, K80, and HKY. The number at the top of the rectangle is the specific value of time complexity.

D. DISCUSSIONS
Beside evaluating the accuracy of NeoPLE by the comparisons of consensus trees, the relationships between model, $nsite$ and $acc$ is included. Fig. 9A illustrates the relationships between model and $acc$ when setting the number of alignment sites to 1000 and the number of generations to the maximum. The curves of different colors represent the different data set. The results indicate that the evolutionary models have no evident effect on the accuracy of likelihood predictions as the ordinate values of different color curves basically fluctuate between 0.93 and 0.94. Fig. 9B illustrates the relationships between $nsite$ and $acc$ when setting the nucleotide substitution models to GTR and the number of generations to the maximum. Experiments show that the number of alignment sites has obvious effect on $acc$, as the value of $acc$ increases with the increase of $nsite$.

The relationship between $ngen$ and $acc$ is illustrated in Fig 10. In order to make the results more comparative, the number of tree generations is gradually increased in each time, in which $nucle12$, $nucle17$, $nucle29$, $nucle37$, $nucle54$ is increased by 1000, 2000, 5000, 7000, 40000, respectively. The experimental results demonstrate that the prediction accuracy of the model is greatly improved with the number of tree generations increases. The curves of different colors represent different number of alignment sites. It can be seen that different curves show an upward trend. Therefore, increasing the number of generations can improve the accuracy.

Overall, the cases of deep neighbor information, likelihood and random forest algorithm are employed to train a novel non-linear prediction model, called NeoPLE, for phylogenetic inference in this study. NeoPLE is successfully integrated in MrBayes, which brings performance improvements by two major benefits relative to the traditional approaches based on exhaustive CLP calculations.

First, a new abstract relational schema describing the relationship between tree structure and its likelihood probability transfers the original likelihood estimate of a tree into regression-based prediction so that the likelihood of each individual sampling tree in the process of phylogenetic inference can be obtained directly and quickly, which substantially...
FIGURE 7. The trend of time complexity \( t_p \) and \( t_c \) with the number of alignment sites. The \( n_{site} \) represents the number of nucleotide sites. \( T \) on the y-axis represents time complexity in seconds. The titles represent different name of taxa, such as nucle12, nucle17, nucle29, nucle37 and nucle54. Blue stands for time complexity \( t_p \). Red stands for time complexity \( t_c \). The elapsed time of NeoPLE is defined as \( t_p \) and the elapsed time to estimate the likelihood using MrBayes is defined as \( t_c \). When the number of sites increases, the time cost remains unchanged.

FIGURE 8. The comparisons between MrBayes and NeoPLE in constructing consensus trees. (A) depicts consensus trees with clade credibility values from MrBayes, while (B) depicts consensus trees from NeoPLE. Percentages of different colors represent the credibility of the clades, and the number at the right hand side of each sub-graph denotes the taxon ID.

reduces the analysis cost. Second, the time cost of completing a phylogenetic inference depends on two key factors, which are the number of taxa and the alignment sites per each taxon. The former determines the sampling size, while the latter adds the burden of time cost calculating the CLPs for sites. Under the new schema, the process of training model, as well as performing a prediction, only relates the number of taxa, which is independent of data set length however. This is due to the efficient and effective feature extraction strategy based on deep neighbor information. The prediction model established using tree branch length between arbitrary tree nodes demonstrates the robust adaptation even if changing
data set length. The benefit in Fig. 7 illustrates increasing data set length does not incur overhead as the time complexity O(1) is achieved.

IV. CONCLUSION
To conclude, the model-based assessment of likelihood usually leads to undesirably long execution time. To overcome,
this study makes the full use of candidate trees and successfully establishes a model that exactly describes the relationships between the likelihood and the extracted features through exploiting the deep neighbor information of each individual tree. Instead of having to depend on the recursive calculations of CLPs node by node in traditional phylogenetic inference tools, the proposed NeoPLE mines knowledge from trees calculated and trains a prediction model, thus the likelihood estimates of the rest trees are directly determined from the model. A significant performance improvement in reducing time complexity demonstrates that NeoPLE is a powerful tool for biological evolutionary research. The use of this method is expected to accelerate the progress of phylogenetic research and help uncover the secrets behind complex biodiversity. Future work on NeoPLE will comprise the achievement of performance and accuracy improvements, as well as the integration of more phylogenetic inference tools.

REFERENCES

[1] M. Kimura, “Evolutionary rate at the molecular level,” J. Genet. Mol. Biol., vol. 18, no. 4, pp. 219–225, 2007.

[2] A. Lele and J. Ottenburghs, “Digest: A single genetic origin and a role for bone development pathways in repeated losses of flight in steamer ducks,” Evolution, vol. 73, no. 9, pp. 2030–2032, Sep. 2019.

[3] Y. Q. Wang and B. Su, “Molecular evolution of microcephalin, a gene determining human brain size,” Human Mol. Genet., vol. 13, no. 11, pp. 1131–1137, Jun. 2004.

[4] T. J. Ward and R. L. J. N. Honeycutt Derr, “Nucleotide sequence evolution at the k-casen locus: Evidence for positive selection within the family bovidae,” Genetics, vol. 147, no. 4, pp. 1863–1872 1997.

[5] G. Stecher, L. Liu, M. Sanderford, D. Peterson, K. Tamura, and S. Kumar, “MEGA-MD: Molecular evolutionary genetics analysis software with mutational diagnosis of amino acid variation,” Bioinformatics, vol. 30, no. 9, pp. 1305–1307, May 2014.

[6] Z. Yang and B. Rannala, “Bayesian estimation of species divergence times under a molecular clock using multiple fossil calibrations with soft bounds,” Mol. Biol. Evol., vol. 23, no. 1, pp. 212–226, Jan. 2006.

[7] M. J. Benton and M. R. Wills Hitchin, “Quality of the fossil record through time,” Nature, vol. 40, pp. 534–537, Feb. 2000.

[8] C. A. Russo, N. Takezaki, and M. Nei, “Efficiencies of different genes and different tree-building methods in recovering a known vertebrate phylogeny,” Mol. Biol. Evol., vol. 13, no. 3, pp. 525–536, Mar. 1996.

[9] J. L. Thorne and H. Kishino, “Divergence time and evolutionary rate estimation with multilocus data,” Systematic Biol., vol. 51, no. 5, pp. 689–702, Sep. 2002.

[10] J. L. Thorne and H. Kishino, “Estimation of divergence times from molecular data sequence,” in Statistical Methods in Molecular Evolution, R. Nielsen, Ed. New York, NY, USA: Springer-Verlag, 2005, pp. 235–256.

[11] W. Zheng, Q. Mao, R. J. Genco, J. Wactawski-Wende, M. Buck, Y. Cai, and Y. Sun, “A parallel computational framework for ultra-large-scale sequence clustering analysis,” Bioinformatics, vol. 35, no. 3, pp. 380–388, Feb. 2019.

[12] R. H. Waterston, “Initial sequencing and comparative analysis of the mouse genome,” Nature, vol. 420, no. 6915, pp. 520–562, 2002.

[13] A. Cooper and R. Fortey, “Evolutionary explosions and the phylogenetic fuse,” Trends Ecol. Evol., vol. 13, no. 4, pp. 151–156, Apr. 1998.

[14] M. D. Hendy, “Hadamard conjugation: An analytical tool for phylogenetics,” in Mathematics of Evolution and Phylogeny, O. Gascuel, Ed. Oxford, U.K: Oxford Univ. Press, 2005, pp. 143–177.

[15] I. K. Jordan, F. A. Kondrashov, I. A. Adzhubei, and Y. I. Wolf, “A universal trend of amino acid gain and loss in protein evolution,” Nature, vol. 433, pp. 633–638, Jan. 2005.

[16] S. Kumar, “MEGA3: Integrated software for molecular evolutionary genetics analysis and sequence alignment,” Briefings Bioinf., vol. 5, no. 2, pp. 150–163, Jan. 2004.

[17] A. Stamatakis, “RAxML version 8: A tool for phylogenetic analysis and post-analysis of large phylogenies,” Bioinformatics, vol. 30, no. 9, pp. 1312–1313, May 2014.

[18] B. S. Everitt, S. Landau, and M. Leese, Cluster Analysis. London, U.K.: Arnold, 2001.

[19] M. N. Wei, “Phylogenetic analysis in molecular evolutionary genetics,” Annu. Rev. Genet., vol. 30, no. 1, pp. 371–403, 1996.

[20] Z. DiNardo, K. Tomlinson, A. Ritz, and L. Oesper, “Distance measures for tumor evolutionary trees,” Bioinformatics, vol. 36, no. 7, pp. 2090–2097, Apr. 2020.

[21] W. Hordijk and O. Gascuel, “Improving the efficiency of SPR moves in phylogenetic tree search methods based on maximum likelihood,” Bioinformatics, vol. 21, no. 24, pp. 4338–4347, Dec. 2005.

[22] B. Chor and T. Tuller, “Maximum likelihood of evolutionary trees: Hardness and approximation,” Bioinformatics, vol. 21, no. 1, pp. 117–118, Jun. 2005.

[23] J. Felsenstein, “Evolutionary trees from DNA sequences: A maximum likelihood approach,” J. Mol. Evol., vol. 17, pp. 386–376, Nov. 1981.

[24] X. Meyer, B. Chopard, and N. Salamin, “Accelerating Bayesian inference for evolutionary biology models,” Bioinformatics, vol. 33, no. 5, pp. 669–676, Mar. 2017.

[25] J. P. Huelsenbeck and F. Ronquist, “MRBAYES: Bayesian inference of phylogenetic trees,” Bioinformatics, vol. 17, no. 8, pp. 754–755, Aug. 2001.

[26] S. Li and D. H. Pearl Doss, “Phylogenetic tree construction using Markov chain Monte Carlo,” J. Amer. Stat. Assoc., vol. 95, no. 450, pp. 493–508, 2000.

[27] B. Mau and M. A. Newton, “Phylogenetic inference for binary data on dengdums using Markov chain Monte Carlo,” J. Comput. Graph. Statist., vol. 6, no. 1, pp. 122–131, 1997.

[28] B. Rannala and Z. Yang, “Probability distribution of molecular evolutionary trees: A new method of phylogenetic inference,” J. Mol. Evol., vol. 43, pp. 304–311, Sep. 1996.

[29] Z. Yang and A. D. Yoder, “Comparison of likelihood and Bayesian methods for estimating divergence times using multiple gene loci and calibration points, with application to a radiation of cute-looking mouse lemur species,” Systematic Biol., vol. 52, no. 5, pp. 705–716, Oct. 2003.

[30] C. D. Bustamante, R. Nielsen, and D. L. Hartl, “Maximum likelihood and Bayesian methods for estimating the distribution of selective effects among classes of mutations using DNA polymorphism data,” Theor. Population Biol., vol. 63, no. 2, pp. 91–103, Mar. 2003.

[31] H.-G. Stratigopoulos et al. and Y. Makris, “Error moderation in low-cost Machine-Learning-Based Analog/RF testing,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 27, no. 2, pp. 339–351, Feb. 2008.

[32] L. J. Tao, Y. R. Feng, and Z. H. Liang, “Statistical machine learning,” J. Henan Normal Univ., vol. 38, no. 6, pp. 35–40, Jun. 2010.

[33] H. Bhaskar, D. C. Hoyle, and S. Singh, “Machine learning in bioinformatics: A brief survey and recommendations for practitioners,” Comput. Biol. Med., vol. 40, no. 10, pp. 1104–1125, Oct. 2006.

[34] C. M. Bishop, Pattern Recognition and Machine Learning, New York, NY, USA: Springer, 2006.

[35] S. Ross, “Efficient inference on known phylogenetic trees using Poisson regression,” Bioinformatics, vol. 23, no. 2, pp. e142–e147, Jan. 2007.

[36] J. Chifman and L. Kubatko, “Quartet inference from SNP data under rate matrices under the GTR model,” Bioinformatics, vol. 21, no. 6, pp. 708–715, Mar. 2006.
[43] C. Cai, K. Yan, and H. Lu, “Intelligent placement model based on decision tree,” in Proc. 9th Int. Conf. Inf. Technol. Med. Educ. (ITME), Oct. 2018, pp. 837–841.

[44] H. Nishimura, M. Kobayashi, and M. Maruyama, “Off-line character recognition using HMM by multiple directional feature extraction and voting with bagging algorithm,” in Proc. 5th Int. Conf. Document Anal. Recognit., 1999, pp. 49–52.

[45] G. Izmirlian, “Application of the random forest classification algorithm to a SELDI-TOF proteomics study in the setting of a cancer prevention trial,” Ann. New York Acad. Sci., vol. 1020, no. 1, pp. 154–174, 2006.

[46] A.-L. Boulesteix, S. Janitza, J. Kruppa, and I. R. König, “Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics,” Wiley Interdiscip. Rev.: Data Mining Knowl. Discovery, vol. 2, no. 6, pp. 493–507, Nov. 2012.

[47] S. Han and H. Kim, “On the optimal size of candidate feature set in random forest,” Appl. Sci., vol. 9, no. 5, p. 898, Mar. 2019.

[48] Y. Lisha, L. Mengying, and L. Yaqin, “Palm Vein classification based on deep neural network and random forest,” Laser Optoelectron. Prog., vol. 56, no. 10, 2019, Art. no. 101010.

[49] V. K. Ayyadevara, “Random forest,” in Pro Machine Learning Algorithms. Bethel Island, CA, USA: Springer, 2018, pp. 105–116.

[50] A. Rambaut and N. C. Grass, “Seq-Gen: An application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees,” Comput. Appl. Biosci., vol. 13, pp. 235–238, Apr. 1997.

CHENG LING received the Ph.D. degree in electronic engineering from Edinburgh University, in 2012. He is currently an Associate Professor with the Beijing University of Chemical Technology. His research interests include high-performance GPU computing on computational biology and bioinformatics.

HAOYU ZHANG received the Ph.D. degree in electronic engineering from Edinburgh University, in 2015. He is currently a Senior Lecturer with Zhejiang Ocean University. His research interests include microwave imaging based cancer detection and airborne SAR based remote sensing.

WENHAO CHENG received the bachelor’s degree from Shanxi University, in 2017. He is currently pursuing the master’s degree with Beijing University of Chemical Technology. His research interest includes machine learning based method for molecular phylogenetic analysis.

HANHAO ZHU received the Ph.D. degree from Harbin Engineering University, in 2014. He is currently an Associate Professor with Zhejiang Ocean University. His research interests include underwater acoustic signal processing and underwater based remote sensing.

HUA ZHANG received the master’s degree in control theory and control engineering from the Zhejiang University of Technology, in 2009. He is currently an Associate Professor with Zhejiang Ocean University. His research interests include wireless sensor networks and underwater wireless sensor networks.

***