

A deep learning–based score to evaluate multiple sequence alignments

Nimrod Serok ^{1†}, Ksenia Polonsky ^{1†}, Haim Ashkenazy ², Itay Mayrose ³, Jeffrey L. Thorne ^{4,5}, Tal Pupko ^{1*}

¹The Shmunis School of Biomedicine and Cancer Research, George S. Wise Faculty of Life Sciences, Tel Aviv University, Tel Aviv 69978, Israel

²Department of Molecular Biology, Max Planck Institute for Biology Tübingen, Tübingen, Germany

³The School of Plant Sciences and Food Security, George S. Wise Faculty of Life Sciences, Tel Aviv University, Tel Aviv 69978, Israel

⁴Department of Biological Sciences, North Carolina State University, Raleigh, NC 27695, USA

⁵Department of Statistics, North Carolina State University, Raleigh, NC 27695, USA

[†]These two authors contributed equally to this work.

*Corresponding author: E-mail: talp@tauex.tau.ac.il.

Associate editor: Russell Corbett-Detig

Abstract

Multiple sequence alignment (MSA) inference is a central task in molecular evolution and comparative genomics, and the reliability of downstream analyses, including phylogenetic inference, depends critically on alignment quality. Despite this importance, most widely used MSA methods optimize the sum-of-pairs (SoP) score, and relatively little attention has been paid to whether this objective function accurately reflects alignment accuracy. Here, we evaluate the performance of the SoP score using simulated and empirical benchmark alignments. For each dataset, we compare alternative MSAs derived from the same unaligned sequences and quantify the relationship between their SoP scores and their distances from a reference alignment. We show that the alignment with the optimal SoP score often does not correspond to the most accurate alignment. To address this limitation, we develop deep learning–based scoring functions that integrate a collection of MSA features. We first introduce *Model 1*, a regression model that predicts the distance of a given MSA from the reference alignment. Across simulated and empirical datasets, this learned score correlates more strongly with true alignment accuracy than the SoP score. However, *Model 1* is less effective at identifying the best alignment among alternatives. We therefore develop *Model 2*, which takes as input a set of alternative MSAs generated from the same sequences and predicts their relative ranking. *Model 2* more accurately identifies the top-ranking MSA than the SoP score, *Model 1*, and several widely used alignment programs. Using simulations, we show that selecting MSAs based on our approach leads to more accurate phylogenetic reconstructions.

Keywords phylogenomics, machine-learning, alignment scoring function, sum of pairs, alignment algorithms

Introduction

Sequence alignment involves inferring sequence homology at the residue level, meaning that when two residues are aligned, they hypothetically evolved from a common ancestral residue. Reliable alignment is therefore essential for accurately modeling evolutionary processes, such as substitutions and indels (Redelings et al. 2024). As a result, sequence alignments are imperative to a wide range of bioinformatics and evolutionary genomics analyses, including phylogenetic inference, molecular dating, ancestral sequence reconstruction, detection of selective forces, and identification of remote homology (Kemeny and Notredame 2009; Thompson et al. 2011). They also play a key role in structural biology—for example, in AI-based protein structure prediction tools, such as AlphaFold (Jumper et al. 2021). Beyond individual proteins or genes, sequence alignments are also used to align entire genomes and investigate large-scale

evolutionary events, such as lateral gene transfer (Roettger et al. 2009). Additionally, they play a crucial role in studying the genetic basis of various diseases, including coronavirus disease (COVID-19) (Pekar et al. 2022), speech and language disorders (Enard et al. 2002), and cancer (Tollis et al. 2017; Gassner et al. 2018).

For aligning pairs of sequences (pairwise alignment), dynamic programming algorithms are commonly used to efficiently explore the space of all possible alignments and identify those that maximize a predefined alignment score (Needleman and Wunsch 1970; Sankoff 1972). A widely adopted scoring scheme is the affine gap penalty model, which assigns distinct costs for substitutions, gap openings, and gap extensions (Gotoh 1982). For protein sequences, substitution scores are typically drawn from empirical matrices, such as PAM and BLOSUM (reviewed in Trivedi and Nagarajaram 2020). Classic dynamic programming algorithms are relatively efficient, requiring computational effort

Received: January 29, 2026. Revised: May 29, 2026. Accepted: June 18, 2026

© The Author(s) 2026. Published by Oxford University Press on behalf of Society for Molecular Biology and Evolution.

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted reuse, distribution, and reproduction in any medium, provided the original work is properly cited.

proportional to the product of the lengths of the two sequences being aligned. However, with the rapid growth of sequence databases and the increasing demand for fast and reliable alignment of query sequences against these datasets, substantial effort has gone into developing high-speed heuristic algorithms for pairwise alignment, such as BLAST (Altschul et al. 1990), MMseqs2 (Mirdita et al. 2019), and DIAMOND (Buchfink et al. 2015). Additionally, statistical alignment algorithms that incorporate explicit models of sequence evolution have been developed (Thorne et al. 1991; Hein et al. 2000). These methods generally offer higher accuracy compared to traditional dynamic programming but are computationally intensive, often several orders of magnitude slower, and are thus rarely used.

The generalization of pairwise alignment algorithms to multiple sequence alignment (MSA) necessitated updating the scoring function to accommodate multiple sequences. The most widely used scoring function for MSAs is the “sum-of-pairs” (SoP) score (Carrillo and Lipman 1988), which is simply the sum of the induced pairwise alignment scores over all pairs of sequences. Specifically, consider an MSA with N sequences, and let S_{ij} be the score of the induced pairwise alignment between sequences i and j . Then $SoP = \sum_{i=1}^{N-1} \sum_{j=i+1}^N S_{ij}$.

The dynamic programming algorithm for pairwise alignment can be extended to MSAs, optimizing the SoP score. However, its computational time increases exponentially with the number of sequences (Wang and Jiang 1994). These extensive computational demands have led to a large number of heuristic algorithms that implement various optimization strategies. Most common aligners, such as Clustal Omega (Sievers and Higgins 2021), MAFFT (Katoh and Standley 2013), and PRANK (Löytynoja 2021), include a progressive alignment component, in which the MSA is iteratively built by uniting sequences according to a guide tree (Feng and Doolittle 1987). The guide tree is reconstructed without relying on the MSA, eg by computing pairwise distances for each pair of sequences and applying distance-based tree reconstruction methods, such as the neighbor joining (Saitou and Nei 1987). Additional heuristic strategies applied to the problem of optimizing the SoP score include iterative refinement (Hirosawa et al. 1995), divide-and-conquer (Zaharias et al. 2023), simulated annealing (Kim et al. 1994), and genetic algorithms (Gondro and Kinghorn 2007).

While many studies have focused on optimizing the SoP score, relatively few have sought to replace it with more predictive alternatives. The SoP score considers only pairwise relationships between sequences, ignoring higher-order features that quantify the conservation of entire columns. The column score, which is the percentage of columns without any substitutions, is one such feature. The information-theoretical entropy of a column, defined as $-\sum p_i \log_2(p_i)$, in which p_i is the frequency of amino acid i in that column, captures more nuanced conservation patterns at each column. The sum of the column-entropy scores (ESs) over all alignment columns, denoted the ES, provides a global conservation across the entire MSA (Shenkin et al. 1991). Similarly, Hertz and Stormo (1999) proposed maximizing the information content of an MSA, defined as the extent to which observed residue frequencies in each column deviate from those expected under a background distribution, such as the overall character frequencies in the MSA. Likewise, norMD quantifies column conservation as an alternative MSA objective function (Thompson et al. 2001). Building on this concept, Nguyen and

Pan (2011) introduced a novel column-scoring method that demonstrated improved performance over previous approaches.

An alternative set of scores aimed to integrate biological understanding, including structural and functional characteristics of the sequences being aligned, eg accounting for contact residues within the 3D structure (Kemena et al. 2011) or proportion of amino acids located in transmembrane regions and functional annotations of the aligned proteins (Ortuño et al. 2015). Orobítz et al. (2013) developed an objective function that combines several previously suggested scores, eg SoP scores with different gap penalties and the Triplet score used by the T-Coffee aligner (Notredame et al. 2000) that measures the consistency between the MSA and a reference library of pairwise alignments. Finally, the ClustalW algorithm (Thompson et al. 1994) modifies the SoP score to better reflect biological context by reducing the gap penalties in regions enriched with hydrophilic amino acids, which are often located on protein surfaces, where indels are more frequent.

A critical challenge in developing novel MSA scoring functions lies in the methodology used to evaluate and compare competing functions. Effective benchmarking requires two key components: (i) a set of reference MSAs and (ii) a metric that quantifies the level of disagreement between a predicted MSA and the reference alignment. However, establishing reliable benchmarks for assessing the performance of alignment methods is inherently difficult (Warnow 2021), and these challenges similarly affect the evaluation of MSA scoring functions. Two main strategies are used for generating benchmark MSAs: (i) simulation-based datasets and (ii) empirical datasets derived from structural alignments. Each approach has its own strengths and limitations. Simulation-based benchmarking offers several advantages. Simulated MSAs can be produced in large quantities, the reference alignment is known, and a diverse range of evolutionary scenarios can be modeled (Iantorno et al. 2014). To ensure biological realism, simulations are typically parameterized using empirical data—for example, sequences are evolved along phylogenetic trees inferred from real datasets, with branch lengths, rate variation across sites, and indel processes modeled to reflect observed biological patterns. However, a significant limitation of simulation-based benchmarks is their reliance on simplified evolutionary models that may not accurately capture important aspects of molecular evolution. For instance, most simulations assume site independence given a phylogenetic tree, whereas in reality, protein structure and function often impose dependencies among sites.

The second benchmarking strategy involves the use of empirical protein alignments informed by structural data. While these datasets aim to reflect biologically meaningful alignments, they are still the result of inference and may not represent the true evolutionary history. Indeed, studies have shown that different structural alignment algorithms can yield inconsistent results (Hasegawa and Holm 2009; Sierk et al. 2010). Edgar (2010) noted that “questionable alignments are found in all benchmarks, especially in BALiBASE,” one of the most widely used reference databases (Thompson et al. 2005). Furthermore, structural similarity can arise through convergent evolution, leading to alignments of structurally similar—but not homologous—regions, which undermines the evolutionary validity of the benchmark (Iantorno et al. 2014). Structural benchmarks are also limited in scope: They primarily include proteins with

well-defined, conserved structural regions, excluding unstructured protein regions and noncoding DNA alignments. In addition, these datasets lack scalability, particularly for applications such as machine learning (ML), which require large and diverse training sets. Given the distinct advantages and limitations of both simulation-based and structure-based benchmarking strategies, it is advisable to evaluate alignment algorithms, including scoring functions, using both approaches.

A key component of benchmarking MSA methods is a metric that quantifies the disagreement between a predicted alignment and a reference alignment. The most commonly used metrics are the Total Column Similarity (TCS) score and the Residue-Pairs Similarity (RPS) score (Thompson et al. 1999). The TCS score measures the proportion of correctly recovered columns from the reference alignment, while the RPS score quantifies the proportion of correctly aligned residue pairs between the predicted and reference alignments (Thompson et al. 2005). Notably, neither the TCS nor RPS score is a true distance metric, as both fail to satisfy key mathematical properties of distance functions, such as symmetry and the triangle inequality. Consequently, several alternative metrics that more rigorously quantify similarity or dissimilarity between MSAs have been proposed (Blackburne and Whelan 2012).

ML has increasingly become a transformative tool in phylogenetics, enhancing both tree reconstruction and sequence alignment processes (Mo et al. 2024), as well as searching for computing branch support values (Ecker et al. 2024; Wiegert et al. 2024). For example, Leuchtenberger et al. (2020) employed neural networks to identify MSAs composed of four sequences that evolved under conditions of long-branch attraction or long-branch repulsion. Haag et al. (2022) used ML to quantify the “difficulty” of an MSA, ie to estimate how challenging it would be for a heuristic tree search to recover top-scoring tree topologies. BetaAlign, a deep learning-based method, utilizes transformer models and leverages natural language processing techniques to perform multiple sequence alignments (Dotan et al. 2024). However, this approach is currently limited in the number of sequences it can process effectively.

We introduce deep learning-based approaches for improved scoring of MSAs and for ranking alternative alignments. Our approach involves generating diverse alternative MSAs using standard alignment tools, followed by evaluation and scoring with trained deep learning models. We developed two types of models. *Model 1* provides an accuracy score for a single input MSA. Although it outperforms the SoP score, it does not perform well in identifying the top-ranking MSA from a set of alternatives. We therefore developed *Model 2*, which takes as input a set of alternative MSAs and predicts their relative ranking. We show that *Model 2* consistently identifies MSAs that are significantly more accurate than those selected by the SoP metrics, *Model 1*, as well as MSAs produced by widely used tools, such as MAFFT and PRANK. These findings are validated on both empirical and simulated datasets.

Materials and methods

We first describe our methodology for protein sequences; modifications for DNA sequences are presented later. To compute the SoP score, one must specify the substitution matrix and the

affine gap penalty. In this paper, the default SoP score is calculated using the BLOSUM62 substitution matrix, with a gap opening penalty of -10 and a gap extension penalty of -0.5 . These parameters were selected because they are the default settings in the EMBOSS Needle program, a widely popular pairwise alignment tool that employs affine gap costs (Rice et al. 2000). We use the fast algorithm suggested by Ranwez (2016) to compute this score efficiently. Of note, additional SoP scores with an additional substitution matrix and with different gap costs are computed as features (see description of the features used below).

We use the d_{seq} measure to quantify dissimilarity between a pair of MSAs, because it is a proper mathematical distance metric (Blackburne and Whelan 2012). Let A denote an alignment of n sequences, and let S_j^i represent the j 'th character of the unaligned sequence S^i . Then, for a given sequence S^i and alignment position j , we identify the alignment column in which S_j^i appears and define $H_{seq}(A)_j^i$ as the collection of all other characters (plus gaps if present) that occupy the same column. Thus, $H_{seq}(A)_j^i$ contains exactly $n - 1$ elements and represents the set of characters (including gaps) that are homologous to S_j^i according to alignment A . Formally:

$$H_{seq}(A)_j^i = \{\chi_j^k \mid k = 1, 2, \dots, i-1, i+1, \dots, n\},$$

where

$$\chi_j^k = \begin{cases} S_m^k & \text{if residue } S_m^k \text{ is aligned with } S_j^i \\ G^k & \text{if sequence } S^k \text{ contains a gap in that column} \end{cases}$$

To illustrate, consider a toy alignment A of length two, with the following three sequences: MC, MY, and M—. For this example, $H_{seq}(A)_1^1 = \{S_1^2, S_1^3\}$, which represents the homology set of the first residue of the first sequence. Similarly, $H_{seq}(A)_2^1 = \{S_2^2, G^3\}$. The contribution of the j 'th position in sequence S^i to the total distance between MSAs A and B , denoted $d_{seq}(A, B)_j^i$, is defined as the dissimilarity between the set of characters aligned to S_j^i in the two alignments: $d_{seq}(A, B)_j^i = \frac{|H_{seq}(A)_j^i \Delta H_{seq}(B)_j^i|}{|H_{seq}(A)_j^i| + |H_{seq}(B)_j^i|}$, where Δ is the symmetric difference of two sets, ie the set of elements in either of two sets, but not in their intersection. The overall distance between the two alignments is computed as the average of these local dissimilarities across all characters: $d_{seq}(A, B) = \frac{1}{c} \sum_i \sum_j d_{seq}(A, B)_j^i$, where c is the total number of characters across all sequences, ie the sum of all sequence lengths: $c = \sum_i |S^i|$.

We aimed to simulate MSAs that resemble empirical MSAs in terms of topology, branch lengths, model parameters (including indel and substitution rates), and alignment length. To this end, we first inferred these attributes from empirical data and used them to generate simulated MSAs. Empirical MSAs and their corresponding trees were extracted from OrthoMaM v.12, a database of orthologous mammalian protein-coding genes (Ranwez et al. 2007; Allio et al. 2024). We first selected 340 datasets whose corresponding phylogenetic trees exhibited the greatest total branch lengths. This criterion has been chosen because we expect that aligning highly variable sequences is more challenging. Subsequently, for each MSA, we randomly sampled a subset of either 40, 50, 60, 70, or 80 taxa (~ 70 MSAs for each number of taxa). Each MSA is associated with the tree, and in each case, we pruned the tree to the corresponding taxa using the ETE3

package in Python (Huerta-Cepas et al. 2016). The tree branch lengths for OrthoMaM datasets were based on coding DNA alignments. Because we analyzed protein MSAs, we re-optimized the branch lengths of all the phylogenetic trees using IQ-TREE with *-s*, *-te*, and *-m LG+G8* parameters. The associated shape parameter of the gamma distribution (α) was extracted from the IQ-TREE log. We then inferred the indel parameters using SpartaABC v.0.6.0 (Ashkenazy et al. 2017). The program was executed with the following command: *sparta -i <path> -t AA -n 100000 -nc 300 -a mafft -s 12345*, and the *LG+G8{alpha}* model as indicated in the *model.bestModel* file.

We then simulated 340 MSAs (hereafter “reference MSAs”) and their corresponding unaligned sequence sets using INDELible v.1.03 (Fletcher and Yang 2009). For each dataset, INDELible was parameterized using the substitution and indel parameters, root sequence length, and tree topology with its associated branch lengths as inferred from the corresponding OrthoMaM dataset, thereby ensuring that simulations reflected empirically estimated evolutionary scenarios. These simulated data were subsequently used as reference datasets, which we denote D_{sim} .

We also benchmarked our algorithm against a set of empirical MSAs obtained from BALiBASE version 4 (Thompson et al. 2005). The benchmark included reference sets RV10, RV11, RV12, RV20, RV30, and RV50. These reference sets represent a wide variety of alignment challenges: Reference set RV10 includes large, complex protein families and is specifically designed to address the challenges of the MSA construction algorithms while working with less frequent patterns; reference sets RV11 and RV12 contain divergent sequences with less than 20% and 40% identity, respectively; reference set RV20 includes families aligned with a highly divergent “orphan” sequence; reference set RV30 contains subgroups with less than 25% residue identity between groups; and reference set RV50 includes families where one or more sequences contain long inserted segments relative to the others. Hence, a total of 294 empirical MSAs were extracted from this dataset, which we denote D_{emp} .

To train and evaluate the deep learning models, a large and diverse labeled dataset is required. Therefore, we generated 1,600 alternative alignments for each of the original reference MSAs derived from either the simulated or empirical datasets. The deep learning algorithms will learn to score the accuracy of a given MSA (*Model 1*) or to rank the accuracy of a set of alternative alignments (*Model 2*). An MSA accuracy is defined as its distance from the corresponding reference alignment (see below). Each reference MSA with its associated inferred alternative MSAs is called hereafter an MSA-batch. The final dataset comprised a total of $1600 \times 340 = 544000$ and $1600 \times 294 = 470400$ alternative MSAs for the simulated and the empirical datasets, respectively.

The following steps were performed for each reference MSA to generate 1,600 alternative MSAs by realigning their sequences: (i) 800 alternative alignments were generated using GUIDANCE (Sela et al. 2015) v.3.1 with default parameters: 400 using MAFFT and 400 using PRANK as the baseline alignment programs. GUIDANCE produces alternative alignments by introducing uncertainty in the guide trees and in the process of indel formation. (ii) An additional 400 alternative MSAs were generated by Muscle5 (Edgar 2022) version 5.2.linux64, which perturbs the guide tree and the hidden Markov model used for MSA construction and creates a so-called diversified ensemble of the alignments. Muscle5 alternative alignments were produced using

the *-diversified -replicates 400 -threads 8* parameters in the command line. (iii) Another 400 alternative MSAs were extracted from the Markov chain Monte Carlo iterations of each BALi-Phy (Gupta et al. 2021) version 3.6.1 run, which records a new alternative MSA every ten iterations. The BALi-Phy program was executed with the parameter *-iter = 4000*.

For each reference MSA, we also obtained four “default MSAs,” defined as the standard inferred alignments produced by four different alignment programs using their default settings. Specifically, we first unaligned the sequences. We next realigned them using each of the following alignment algorithms: MAFFT (Katoh 2002; Katoh and Standley 2013) version v7.525, PRANK (Löytynoja 2014; Löytynoja 2021) v.170427, Muscle5 (Edgar 2022) version 5.2.linux64, and BALi-Phy (Gupta et al. 2021) version 3.6.1. The default alignments of MAFFT, PRANK, and Muscle5 were produced with the default settings. The BALi-Phy *bp-analyze* command line was run to produce *P1.max.fasta* alignment, which was used as the default alignment for this algorithm.

A total of 153 input features were extracted from each simulated MSA. Two additional features related to 3D protein structures were extracted for empirical MSAs. The list of all 155 features is detailed in Information S1. Each of the 155 features can be assigned to one of the following categories: (i) unaligned sequence attributes, eg the length of the unaligned shortest sequence; (ii) MSA attributes, eg the total number of MSA columns; (iii) SoP-related features, eg the sum of pairs cost, using the PAM250 substitution matrix with a gap opening and gap extension penalty of -10 and -1 ; (iv) gap-related features, eg the total number of gaps of length one in the MSA; (v) tree-related features, eg the average branch length, when branch lengths are optimized on a neighbor-joining tree extracted from the MSA; (vi) entropy-related features, eg the average entropy of an alignment column; (vii) k-mer-related features, eg the average frequency of k-mer of length five found within the MSA; and (viii) structure-related features, eg the fraction of columns that are identical between the MSA and a FoldMason-based structural alignment of the same set of sequences.

One of the primary objectives of this study was to develop a novel deep learning-based scoring function for MSAs that correlates strongly with the quality of the inferred alignment, as measured by its d_{seq} distance to the reference alignment. To this end, we trained a deep learning regression model that receives as input the MSA features and predicts d_{seq} , referred to as *Model 1*. Figure S1 illustrates the architecture of our neural network model, and Table S1 provides additional network attributes and the optimal hyperparameters for this model. Hyperparameters were optimized using the Optuna package (Akiba et al. 2019), maximizing the correlation between the predicted and true d_{seq} values. The above D_{sim} and D_{emp} data were each split into train, validation, and test sets in a 64:16:20 ratio. To avoid data leakage, all alternative MSAs derived from the same original dataset and based on the same sets of sequences were assigned to the same partition (train, validation, or test).

We subsequently trained a second model (referred to as *Model 2*), which incorporates modifications to *Model 1* that focus specifically on selecting the most accurate MSA from a given collection of alternatives. To support this objective, the target label was changed to a rank-based normalized version of the original d_{seq} distance. Normalization was performed independently within an MSA-batch, ie the set of alternative MSAs derived from the

same set of sequences. For a given alternative MSA with index i within an MSA-batch, the scaled target value was computed as $Scaled_i = \frac{r_i - 1}{n - 1}$, where r_i represents the rank of the MSA with index i in this MSA-batch according to its d_{seq} distance value, and n denotes the total number of alternative MSAs in the MSA-batch. In case of ties, all entries with the same rank were given the average rank (eg if two MSAs had ranks 5 and 6 with the same score, both obtained the rank 5.5). Feature values were also normalized for all alternative MSAs within the same MSA-batch. For a given feature, all its feature values across the 1,600 alternative MSAs within an MSA-batch were ranked, and the scaled feature value was computed using the same rank-based approach described above for the target value.

When running *Model 2*, our objective is to distinguish between MSAs comprising an MSA-batch. During training and testing, a minibatch of MSAs is processed by the deep learning model (a minibatch is a standard term in learning algorithms, which is the number of samples processed in each learning iteration. Note that this is a different “batch” from the MSA-batch). We implemented a custom “batch generator” that ensured all MSAs within a minibatch belong to the same MSA-batch, ie *Model 2* “sees” each time it processes data only MSAs that belong to the same MSA-batch.

To emphasize the accurate ranking of the best alignments within each batch, the loss function was modified to penalize prediction errors more heavily for the top-performing MSAs. Specifically, the loss was defined as $Loss = MSE + \tau_1 \times MSE_{top \hat{n}}$, where MSE denotes the mean squared error across the entire batch, and $MSE_{top \hat{n}}$ represents the mean squared error computed over the $\hat{n}$ samples with the lowest (ie best) target labels within that batch. The free parameter τ_1 controls the relative contribution of the MSE versus the $MSE_{top \hat{n}}$. The network attributes and hyperparameters of *Model 2* are listed in [Table S1](#). The selected hyperparameters maximized the validation loss. Both models were trained using TensorFlow Keras ([Abadi et al. 2016](#); [Chollet 2018](#)).

To train the DNA version of *Models 1* and *2*, we generated 350 DNA MSA-batches. We constructed 70 MSA-batches, each containing 40, 50, 60, 70, or 80 sequences. Each MSA-batch included 400 alternative MSAs generated using GUIDANCE-MAFFT. Of the 350 MSA-batches, 80% were randomly selected for training the models, while the remaining 20% were reserved for testing.

To train these models, several modifications were introduced to the features. These include the following: (i) counting all possible substitutions ($A \leftrightarrow C$, $A \leftrightarrow G$, $A \leftrightarrow T$, $C \leftrightarrow G$, $C \leftrightarrow T$, $G \leftrightarrow T$); (ii) using a DNA-based substitution matrix derived from Kimura’s two-parameter model instead of the protein PAM and BLOSUM matrices when computing the SoP score; (iii) changing the gap penalty parameters when computing the SoP score; and (iv) increasing the length of the considered k-mer-based features (detailed in [Information S1](#)). For DNA sequences, we used the same deep network architecture, hyperparameters (eg τ_1), and train–test partition ratio as those used for protein sequences.

Our methodology takes as input a set of alternative MSAs. The first step consists of feature extraction across all MSAs in the batch. This step produces a CSV file summarizing the extracted features across all MSAs; this file is provided in the GitHub repository. The feature file is then used as input to the trained deep neural network model. Model inference outputs a score for each MSA, provided as an additional CSV file. The source code

used to generate all figures presented in this manuscript is also included in the repository.

Results

First, we aimed to assess the extent to which existing alignment quality measures are capable of predicting the distance from the reference alignment. Two alignment quality measures were considered: SoP and the entropy score (ES). The SoP quality measure is expected to be positively correlated with alignment quality (and negatively correlated with the distance from the reference MSA). The ES is expected to be negatively correlated with alignment quality (and positively correlated with the distance from the reference MSA). To demonstrate how we evaluate the performance of each of these MSA quality metrics, consider a single MSA-batch, eg empirical MSA BBA0169 from the BALiBASE dataset. This MSA-batch includes a known reference alignment and 1,600 inferred alternative alignments, generated by various alignment tools. For each alternative MSA, we measured the alignment quality (eg the SoP score) and its distance from the reference alignment using the d_{seq} distance metric, which has the advantage of being a mathematically valid distance measure that satisfies properties, such as symmetry and the triangle inequality ([Blackburne and Whelan 2012](#)). We examined the Pearson correlation between the min–max normalized quality metric values and the distance from the reference alignment. For example, in the case of the SoP score, we expect higher values to correspond to smaller distances from the reference alignment and vice versa. The Pearson correlation for this MSA-batch (BBA0169) was -0.91 ([Fig. 1a](#)). The correlation coefficient for the ES was -0.61 . We repeated the same analysis for a single simulated MSA-batch ([Fig. 1b](#)). In this case, SoP performed similarly to ES, with correlation coefficients of -0.78 and 0.77 , respectively.

We next repeated this analysis across all 294 empirical BALiBASE MSA-batches ([Fig. 1c](#)) and 340 simulated MSA-batches generated through a simulation process based on the OrthoMaM database ([Fig. 1d](#)). On the empirical datasets, the SoP metric showed higher predictive accuracy compared to ES in 285 out of 294 cases (ie higher absolute Pearson coefficients). For the simulated data, SoP performed better than ES in 187 out of 340 cases. SoP correlated similarly with other MSA similarity metrics, ie when Residues-Pairs and Total Column were used instead of the d_{seq} metric ([Fig. S2](#)). Since the SoP metric performed better than the ES metric, and is the most commonly used approach to score MSAs, we decided to use SoP in all subsequent analyses.

Using the above approach, we next evaluated our deep learning–based scoring scheme, hereafter referred to as *Model 1*, and compared its performance with that of the SoP metric on the same representative cases shown in [Fig. 1](#). For the empirical MSA-batch BBA0169, the Pearson correlation coefficients were 0.95 for *Model 1* and -0.91 for SoP ([Fig. 2a](#)). *Model 1* also outperformed SoP on the simulated MSA-batch, with Pearson coefficients of 0.90 and -0.79 , respectively ([Fig. 2b](#)). Importantly, these test examples were not used during model training. A similar trend was observed when a larger set of test examples was analyzed ([Fig. 2c and d](#)). As these panels display only the test data, excluding the training data, the number of datasets corresponds to one-fifth of the total: 59 empirical and 68 simulated

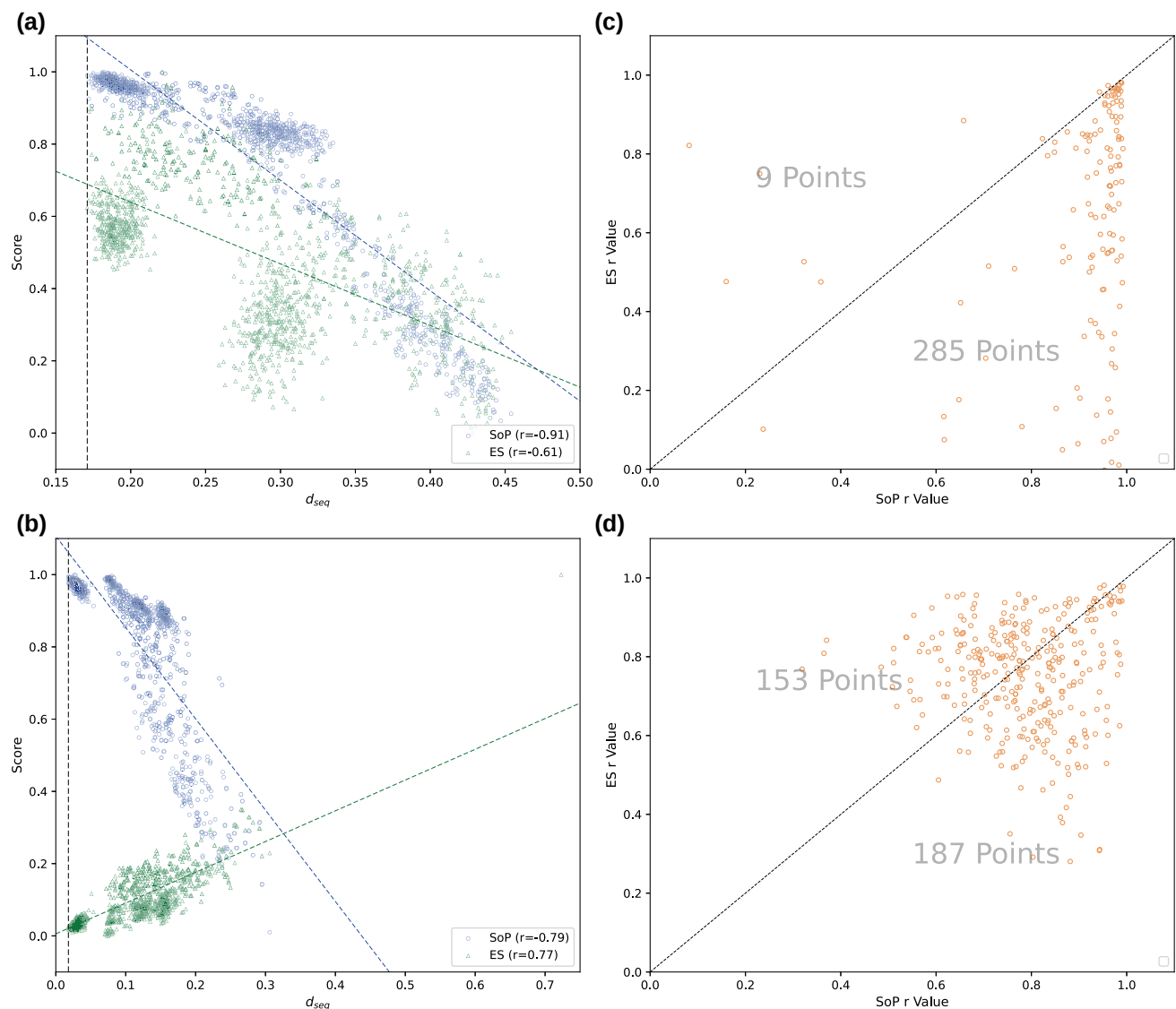


Figure 1 a) Empirical MSA-batch BBA0169. Each dot represents the score obtained for one alternative MSA. For each alternative MSA, we computed two alignment quality scores: SoP and Entropy-Sum (ES), as well as d_{seq} , the distance of the inferred MSA from the reference MSA. The Pearson correlation coefficient between each MSA quality score and d_{seq} is shown at the lower right. The quality scores were normalized to be between zero and one, using the formula $\frac{x - \min}{\max - \min}$, where x , $\min$, and $\max$ correspond to the scores of the alternative MSA, the minimal score among all alternative MSAs, and the highest score, respectively. b) Similar analysis on a simulated MSA-batch. c) Distribution of Pearson correlation coefficients across 294 empirical BAliBASE MSA-batches. Data points above the $y = x$ line correspond to cases in which ES outperformed SoP, ie higher correlation coefficients. d) Distribution of absolute values of Pearson correlation across 340 simulated MSA-batches, each reflecting the evolutionary dynamics of an OrthoMaM alignment.

datasets. For these datasets, *Model 1* outperformed SoP in 52 (88.1%) and 59 (86.8%) of the cases, respectively. Overall, these results indicate that across both empirical and simulated datasets, the two variants of *Model 1* (one for empirical and one for simulated data) consistently predict the distance from the reference alignment more accurately than the SoP metric.

Model 1 predicts multiple MSAs close to the reference MSA, which appears as a scatter of dots in Fig 2a and b. These MSAs are characterized by similar small distances to the reference MSA. *Model 1* is unable to accurately identify the most accurate MSA out of this scatter. The same limitation is even more emphasized when the SoP metric is used.

This outcome is not surprising because the correlation calculation includes all evaluated alignments, including those that are very distant from the reference alignment, and *Model 1* was trained implicitly assigning all of them with the same importance. Therefore, high overall performance across the entire dataset does not necessarily translate into accurate identification of the alignments closest to the reference MSA. However, researchers are often most interested in identifying the best alignment, rather than in accurately predicting distances for distant, low-quality alignments. In light of this, we proceeded to evaluate how well SoP and *Model 1* perform in the task of identifying the alignment closest to the reference alignment.

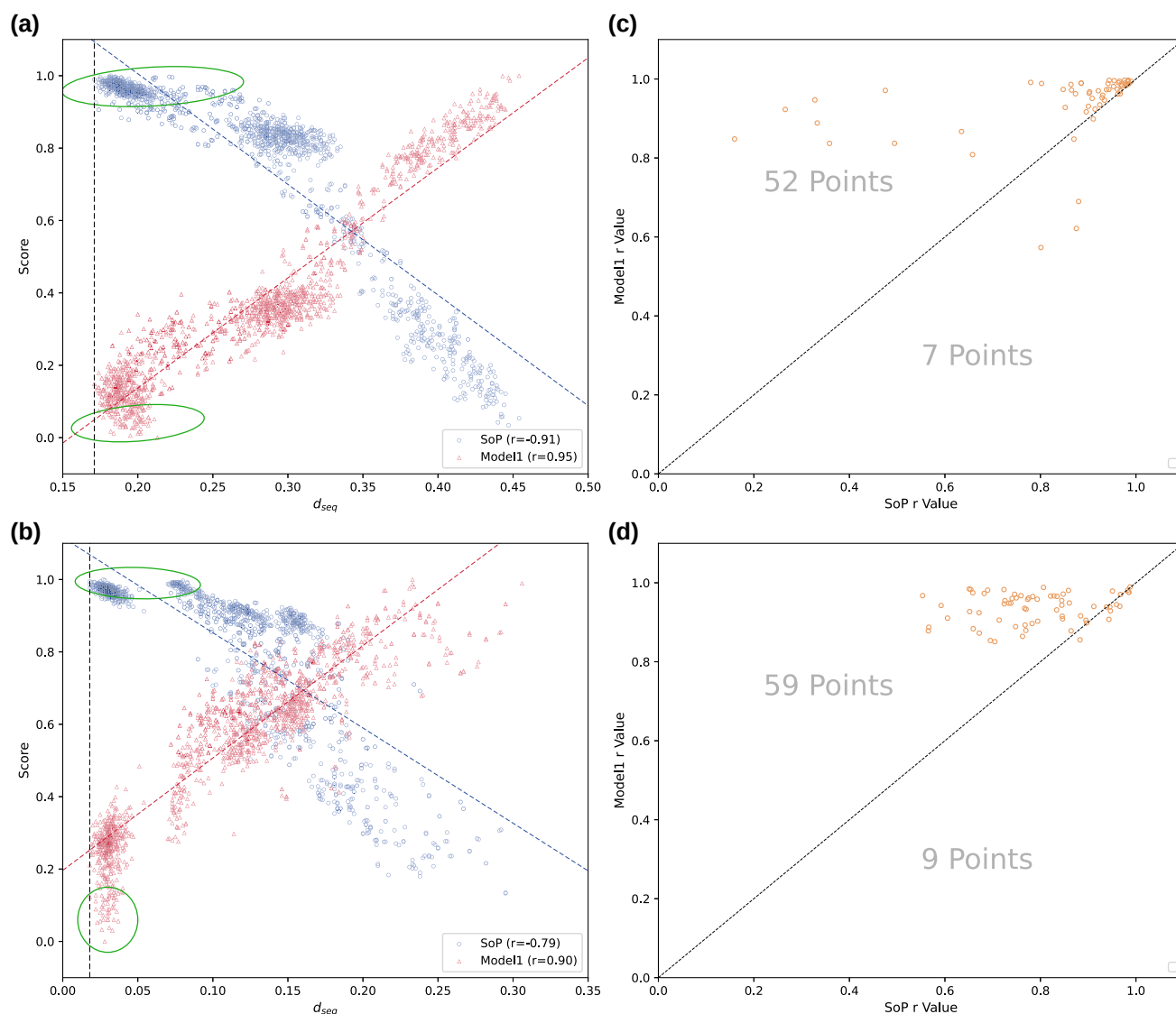


Figure 2 a) Empirical MSA-batch BBA0169. The Pearson correlation coefficients between an MSA quality score (either SoP or *Model 1*) and d_{seq} were computed, similar to Fig. 1a. A vertical dashed line indicates the d_{seq} value of the most accurate MSA among the alternatives. The oval shapes highlight top-scoring MSAs that both SoP and *Model 1* struggle to rank correctly. b) Similar analysis on a simulated MSA-batch. c) Distribution of absolute values of Pearson correlation across 59 empirical test MSA-batches for SoP and *Model 1*. Data points above the $y = x$ line correspond to cases in which *Model 1* outperformed SoP, ie higher correlation coefficients. d) Similar analysis across 68 simulated test MSA-batches.

To further exemplify this issue, we analyzed the distribution of the distance from the reference MSA (d_{seq}) over all 1,604 alternative and default MSAs, for a single empirical dataset (BBA0169) (Fig. 3a). In this plot, the alternative MSA that was closest to the reference alignment has a d_{seq} of 0.171. Note that this distance is nonzero, as the reference alignment itself was not among the alternative alignments generated by the inference tools. The d_{seq} values associated with the default MSAs produced by the various aligners were 0.182, 0.218, 0.285, and 0.293 for Muscle, MAFFT, PRANK, and BALI-Phy, respectively. The inferred MSA with the highest (best) SoP score yielded a d_{seq} value of 0.182, while selecting the MSA with the lowest (best) *Model 1* score resulted in a d_{seq} value of 0.213. These findings highlight the limited effectiveness of *Model 1* in identifying the optimal MSA from a set of alternatives, despite its ability to predict with high accuracy the distance

from the reference MSA when a large number of alternative MSAs are considered.

As stated above, among the alternative MSAs shown in Fig. 3a, the one that was closest to the reference MSA had a d_{seq} value of 0.171. Using the SoP score, the (normalized) score of the closest MSA was 0.99, and 33 out of the 401 top-quartile alternative MSAs had SoP values higher than this MSA (Fig. 3b). When the goal is to find the optimal MSA, these are errors of the SoP scoring scheme. *Model 1* predicted the (normalized) score of this optimal alternative MSA to be 0.149. Among the 1,604 scores, 313 out of 401 MSAs had predicted scores below this score (Fig. 3c). These are clear errors of *Model 1*. These results highlight the need for an alternative score to discriminate among MSAs that are closest to the reference one.

To better discriminate between alternative MSAs that are similar to the reference one, we tested several alternatives, including

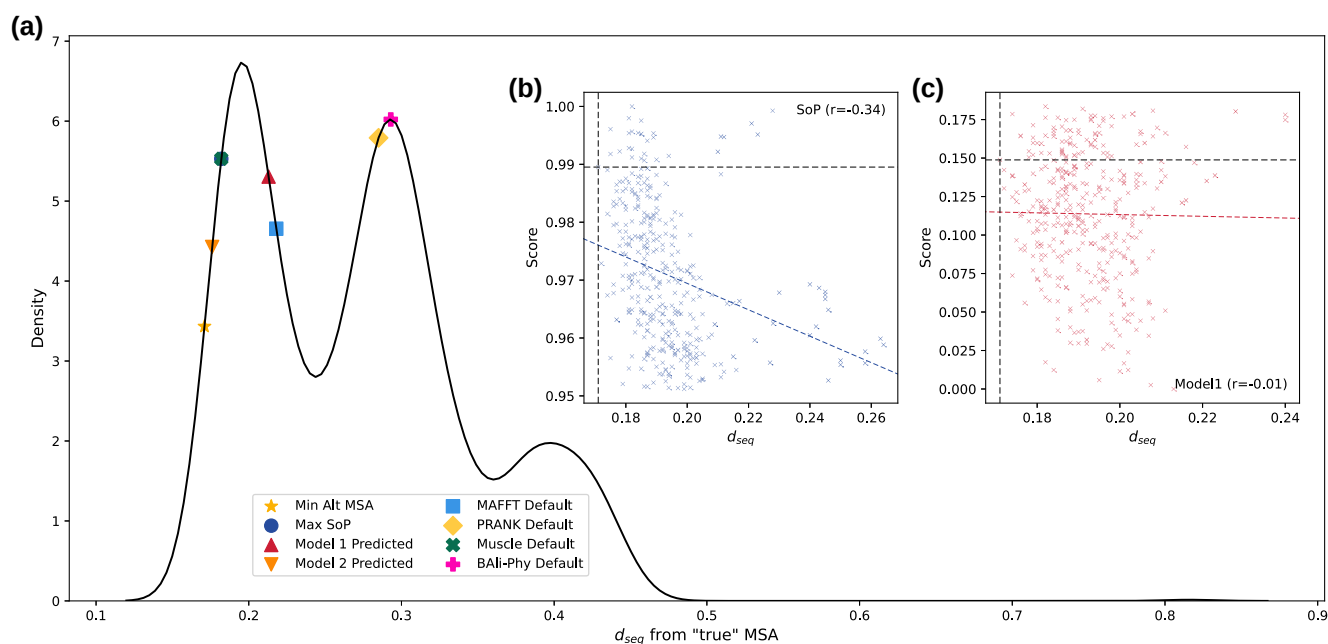


Figure 3 Analysis of empirical MSA-batch BBA0169. a) Probability density function of d_{seq} value for 1,604 alternative and default MSAs. Markers indicate the positions of the default alignments (MAFFT, PRANK, BALI-Phy, Muscle), the highest SoP-scoring MSA ("max SoP"), the MSA selected using *Models 1* and 2, and the alternative MSA closest to the reference MSA ("Min Alt MSA"). The density observed to the left of the Min Alt MSA marker reflects a smoothing artifact introduced by the kernel density estimation and does not correspond to additional alignments. Note that the Max-SoP and Muscle-Default markers overlap. b) d_{seq} and SoP predicted scores for the (top quartile) 401 alternative MSAs with the lowest d_{seq} . The d_{seq} and predicted score of the alternative MSA that is closest to the true MSA are indicated by vertical and horizontal lines, respectively. Alternative MSAs above the horizontal lines were predicted to be superior to the closest MSA and hence represent clear errors of SoP. c) d_{seq} and *Model 1* predicted scores for the 401 alternative MSAs with the lowest d_{seq} . Alternative MSAs below the horizontal lines represent clear errors of *Model 1*.

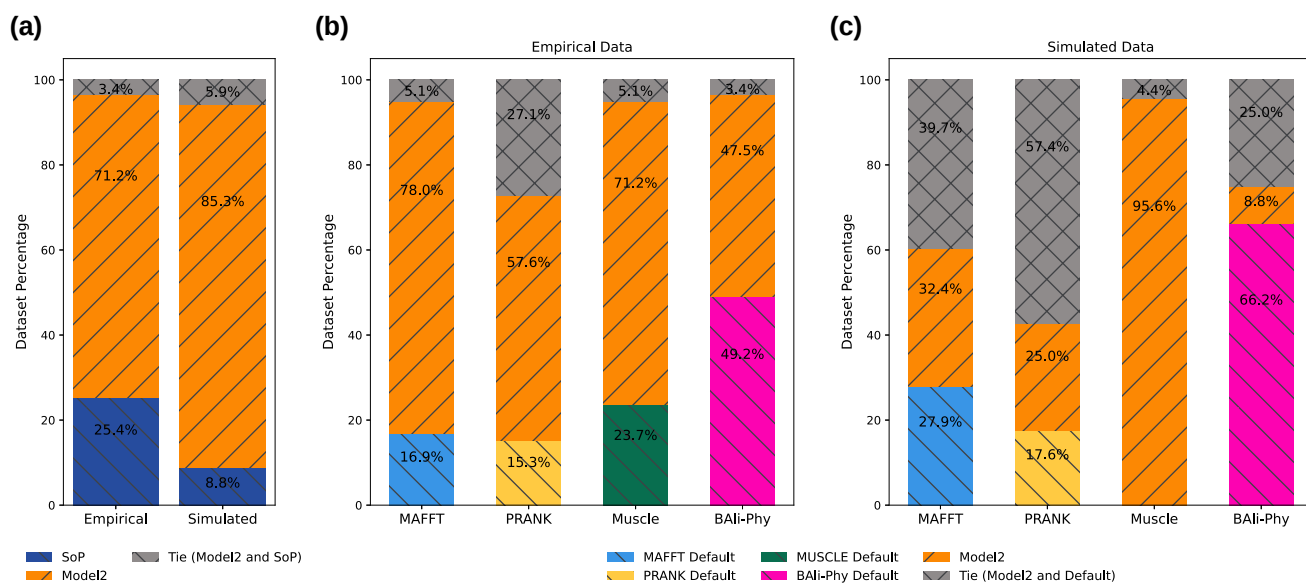


Figure 4 Performance of *Model 2* on the "pick-the-best-MSA" task. For each MSA-batch, each method (*Model 2*, the SoP score, or an aligner's default output) selected a single MSA from the set of alternative alignments, and performance was assessed by comparing the distance of the selected MSA from the reference MSA. a) Proportion of MSA-batches in which *Model 2* or the SoP score identified the more accurate alignment, or in which both methods tied; shown separately for the empirical (59 batches) and simulated (68 batches) datasets. b) For the empirical dataset, proportion of cases, stratified by alignment program, in which *Model 2* selected a more accurate MSA than the default output of that program, the default output was more accurate, or the selections tied. c) Same as b), but for the simulated test set.

the following: (i) training on the subset of MSAs with the lowest d_{seq} , (ii) training on augmented data of MSAs with the lowest d_{seq} , (iii) alternative loss functions that penalize more heavily MSAs with low d_{seq} , and (iv) several ranking functions, such as RankNet (Burgess et al. 2005), Kendall loss (Kendall 1938; Brinker and Hüllermeier 2020), and NDCG (Chen et al. 2009). *Model 2* (discussed below) outperformed these approaches, and hence we only show *Model 2* results.

Model 2 was trained to rank alternative MSAs and, accordingly, receives an MSA-batch as input. This is in contrast to *Model 1*, which receives as input a single MSA and predicts its accuracy. *Model 2* was trained using rank-normalized target values, and its loss function penalizes errors in the relative ranking of accurate MSAs more heavily (see Materials and methods). For the example analyzed in Fig. 3, *Model 2* selected an inferred MSA with $d_{seq} = 0.176$, better than the MSA predicted by *Model 1* ($d_{seq} = 0.213$) and only 2.9% greater distance than the best possible alignment ($d_{seq} = 0.171$). The prediction of *Model 2* was better than those selected by all default aligners. Specifically, the distance from the best alternative alignment was 6.4% to 71.3% higher for the default aligners compared to *Model 2* alignments (Fig. 3a). We next tested the hypothesis that *Model 2* outperforms *Model 1* in terms of correlation with d_{seq} . This was not the case (Fig. S3), highlighting the differences between the two models: While *Model 1* can predict MSA quality without requiring alternative MSAs, *Model 2* is trained to pick the top-scoring MSA among alternative MSAs, a task it performs better than *Model 1*.

We next evaluated the performance of *Model 2* on a large set of datasets using the “pick-the-best-MSA” task, designed to quantify the ability to identify the most accurate alignment from a set of alternative MSAs. In each instance, the model was tasked with selecting a single alignment, and this choice was compared against selections made by the SoP score and the default outputs of the alignment algorithms. As shown in Fig. 4a, *Model 2* selected alignments that were closer to the reference alignment than those chosen by SoP in 71.2% and 85.3% of the cases in empirical and simulated sets, respectively ($P < 0.0004$, chi-square test with $df = 1$, in both cases). Except for the case of BALI-Phy, *Model 2* also outperformed the default aligners, consistently choosing better MSAs, both for empirical and simulated datasets (Fig. 4b and c, respectively). These results underscore the practical effectiveness and value of *Model 2* in real-world scenarios and demonstrate its superiority over both traditional scoring methods and alignment tool outputs.

We next examined the extent to which the performance of *Model 2* is affected by total sequence divergence, as measured by the sum of branch lengths in the underlying tree. Specifically, we randomly selected three simulated tree/parameter combinations from the test set and scaled each tree by a factor of 1, 2, 4, 8, or 16. We then simulated “true” MSAs on these trees under the corresponding model parameters, yielding 15 tree-MSA pairs. For each pair, we unaligned the sequences and realigned them using MAFFT, PRANK, or Muscle and generated alternative MSAs using GUIDANCE (for MAFFT and PRANK) or internally within Muscle5. For each dataset and branch-length scaling factor, we ranked all alternative MSAs by accuracy and determined the rank of the alternative MSA selected by *Model 2* and by SoP. As sequence divergence increased, the ability of both *Model 2* and SoP to identify the best MSA decreased. For example, in one simulated MSA-batch using the original branch

lengths, *Model 2* selected the fifth-ranked MSA among the alternatives, whereas SoP selected the alignment ranked 71st. When all branch lengths were multiplied by 16, *Model 2* selected the highest-ranking alignment, whereas SoP selected the MSA ranked 296th (Table S2). Across 15 comparisons, *Model 2* selected more accurate MSAs than SoP in five cases, six comparisons resulted in a tie, and SoP selected a more accurate MSA in four cases. These results suggest that *Model 2* retains an advantage over SoP in identifying high-quality alignments even under conditions of substantial sequence divergence.

We next examined whether our models can accurately rank MSAs containing substantially larger numbers of sequences, although the model had not been trained on MSAs of this scale. To this end, we evaluated the ability of *Model 2* to rank alternative MSAs comprising 1,000 and 5,000 sequences, while the model was trained on MSAs with less than 100 sequences. Because generating alternative MSAs for such large datasets is computationally demanding, we evaluated only a single set of 400 simulated alternative MSAs for the 1,000-sequence dataset and 148 alternative MSAs for the 5,000-sequence dataset.

For the 1,000-sequence dataset, the rank correlation between the scores assigned by *Model 2* and the distance from the true MSA was $r = 0.91$, indicating strong agreement between the model ranking and MSA accuracy. This performance exceeded that of SoP, which achieved a rank correlation of $r = -0.81$ (Fig. S4). Moreover, the MSA selected by *Model 2* was closer to the true MSA than the one selected by SoP ($d_{seq} = 0.080$ and 0.087 , respectively). Similar trends were observed for the 5,000-sequence dataset, for which the rank correlations were $r = 0.96$ for *Model 2* and $r = -0.84$ for SoP (Fig. S4). In this case, both methods selected the MSA closest to the true MSA.

We next evaluated an alternative approach for scoring large MSAs. Given an MSA with 1,000 sequences, rather than applying *Model 2* directly, we randomly sampled a subset of sequences and constructed a corresponding sub-MSA comprising only the selected rows and columns that contained at least one nongap character among them. *Model 2* was then applied to this sub-MSA. In our experiments, we sampled 60 sequences—within the range used for training *Model 2*—and computed a predicted score for each sub-MSA. This procedure was repeated multiple times (five replicates), and scores were averaged to obtain a final estimate for each alternative MSA. All MSAs were subsequently ranked according to their average *Model 2* scores. The same sampling strategy was also applied to SoP. Applying this approach to a simulated dataset of 1,000 sequences with 400 alternative MAFFT alignments, we obtained rank correlations of $r = 0.89$ (*Model 2*) and $r = -0.77$ (SoP). These values closely match those obtained when scoring the full MSAs directly ($r = 0.91$ and $r = -0.81$ for *Model 2* and SoP, respectively). Together, the results of both approaches indicate that a model trained on only a few dozen sequences can accurately rank alternative MSAs containing 1,000 and even 5,000 sequences.

The sampling strategy may improve performance when evaluating large MSAs, as the resulting sub-MSAs more closely resemble the training regime of *Model 2*. However, subsampling inevitably discards information, which could reduce accuracy. The high concordance between sampling-based and full-MSA evaluations suggests that both approaches provide reliable rankings of large alternative MSAs.

We next examined how the number of alternative MSAs per batch affects the performance of *Model 2*. Specifically, for each

of 68 held-out test MSAs (ie simulated MSA-batches not used for training), we randomly sampled 20, 50, 100, 200, 400, 800, or 1,600 alternative MSAs. *Model 2* outperformed SoP across all sample sizes, and both approaches benefited from having at least 100 alternatives available (Fig. S5). In the “pick-the-best-MSA” task, *Model 2* benefited from larger candidate sets, increasing its advantage over the default alignments produced by MAFFT, Muscle, and PRANK and partially narrowing the gap to BALi-Phy (Fig. S6), which nonetheless remained the most accurate method. For example, when 50 alternatives were considered, *Model 2* selected an alignment that was more accurate than the default Muscle alignment in 70.6% of cases, whereas Muscle was more accurate than *Model 2* in 5.9% (ties occurred in 23.5%). When the candidate set was expanded to 1,600 alternatives, *Model 2* selected a more accurate alignment than the default Muscle alignment in all test cases. Finally, we quantified how often increasing the number of alternatives beyond 20 improved the best attainable alignment (Fig. S7). This analysis indicated that larger candidate sets more frequently contained an alignment closer to the true MSA; for instance, increasing the number of alternatives from 20 to 1,600 yielded a more accurate selected alignment in 52.9% of batches. Taken together, these results show that *Model 2* can already outperform default alignments when as few as 50 alternatives are available, but its performance improves further as more alternative MSAs are considered.

We next assessed whether our approach could accurately rank DNA-based MSAs. To this end, we trained DNA-specific versions of *Model 1* and *Model 2* using DNA alignments (see [Materials and methods](#)). In a representative example comprising 40 simulated sequences and 400 alternative MSAs, the score produced by *Model 1* showed a strong correlation with the distance to the true simulated MSA ($r = 0.95$), outperforming SoP ($r = -0.89$; Fig. S8a). Extending this analysis to 70 independent DNA MSA-batches not used for training, *Model 1* outperformed SoP in 66 cases (94.3%; Fig. S8b).

We then evaluated the DNA-trained version of *Model 2* for its ability to rank alternative alignments by accuracy. In a “pick-the-best-MSA” test across the same 70 MSA-batches, the top-ranked MSA identified by *Model 2* was more accurate than that selected by SoP in 81.4% of cases, matched SoP in 10.0%, and was less accurate in 8.6% of cases. Together, these results indicate that our approach generalizes effectively to noncoding DNA sequences.

To gain insights into the models’ decision-making, we analyzed the 40 most influential features for the regression-based *Model 1* and the ranking-optimized *Model 2*. Feature importance was quantified using SHAP (SHapley Additive exPlanations) values (Lundberg and Lee 2017), which quantify the contribution of each feature to the final prediction (Table 1). Notably, the sets of top features differed between *Model 1* and *Model 2*, consistent with their distinct training objectives. For *Model 2*, SoP-based features were strongly enriched, including SoP scores computed under multiple substitution matrices (eg BLOSUM62 and PAM250) and a range of gap opening and gap extension penalty combinations. This pattern suggests that no single parameter set is universally optimal and that performance benefits from integrating signals across diverse parameterizations. Accordingly, rather than relying on a fixed configuration, *Model 2* effectively exploits a broader landscape of scoring regimes simultaneously, enabling adaptation to dataset-specific

characteristics. This advantage is not available to non-BALI-Phy approaches that depend on a single parameter set. In addition, protein structural features were highly important for empirical benchmarks (eg FoldMason residue pairs score and FoldMason residue pairs score_rank). Finally, entropy-based features (entropy_sum and entropy_sum_rank) appeared to be the most contributing features for the simulated benchmarks, but did not rank among the top contributors in the empirical benchmarks, indicating that they capture complementary aspects of MSA quality not reflected by SoP alone.

To evaluate the effect of the underlying MSA on phylogenetic accuracy, we compared the normalized Robinson-Foulds (RF) distances between inferred and true trees across 68 test simulated datasets. Trees reconstructed from the reference MSAs and those from the MSAs with the best *Model 2* predicted scores exhibited very similar mean RF distances (0.316 vs. 0.305, respectively; $P > 0.05$, paired one-tailed t -test). In contrast, trees reconstructed from MAFFT default MSAs exhibited significantly reduced topological accuracy compared to those inferred from the best predicted MSAs (mean RF distance of 0.336 vs. 0.316; $P < 0.05$, paired two-tailed t -test). Trees inferred from the MSAs generated using the default Muscle and PRANK aligners exhibited even lower RF distance than those inferred from MAFFT (not shown). Overall, these results demonstrate that phylogenies inferred from MSAs selected using the trained *Model 2* ranking function achieved accuracy comparable to those based on the true alignments and outperformed trees reconstructed from standard MAFFT default alignments.

We next evaluated the computational cost of our approach. Given a trained *Model 2*, the first step is to generate a set of alternative MSAs. In this study, we generated 1,600 alternative protein MSAs for each MSA-batch. For the empirical dataset BBA0169 (Fig. 2), this included 400 GUIDANCE-MAFFT alternatives (1 min 5.28 s; four processes), 400 GUIDANCE-PRANK alternatives (29 min 15.68 s; four processes), 400 Muscle-based alternatives (13 min 34.3 s; four threads), and 400 posterior BALi-Phy samples (over 1 week). Feature extraction on these alternative MSAs required 1 min 9.6 s, and ranking all 1,600 MSAs using the trained *Model 2* took 1.7 s. Thus, the overwhelming majority of the computational cost is devoted to generating alternative MSAs. Similarly, during training, most of the runtime was spent on generating alternative MSAs for each batch, whereas simulation, feature extraction, and parameter tuning contributed negligibly to the total runtime.

Discussion

In this work, we developed two deep learning models that address distinct aspects of MSA evaluation. *Model 1* acts as a scoring function, mapping any given MSA to a single numerical value that serves as a proxy for its likelihood or overall quality. Of note, *Model 1* is not intended as a diagnostic tool for assessing absolute MSA quality, as it lacks an interpretable reference scale. For example, an MSA with a predicted *Model 1* score of 0.12 may represent a high-quality MSA in the context of highly diverged, gappy sequences, or a poor-quality MSA for closely related sequences. *Model 2*, in contrast, was trained to directly compare alternative MSAs generated from the same set of sequences, particularly to discriminate among the top-scoring

Table 1 Mean absolute value of the top ten features impact on model output using SHAP values.

Empirical data			
Model 1		Model 2	
Feature	Mean absolute SHAP value	Feature (rank-normalized)	Mean absolute SHAP value
1 FoldMason residue pairs score	0.078	FoldMason residue pairs score_rank	0.092
2 seq_max_len	0.026	sp_BLOSUM62_GO_-10_GE_-0.2_rank	0.039
3 seq_min_len	0.023	sp_go_rank	0.028
4 sp_go_norm	0.016	sp_mismatch_count_rank	0.024
5 sp_ge_norm	0.015	sp_PAM250_GO_-10_GE_-0.5_rank	0.021
6 n_unique_sites	0.011	sp_norm_PAM250_GO_-6_GE_-1_rank	0.020
7 gaps_len_one	0.009	sp_ge_rank	0.018
8 k_mer_average_K5	0.009	sp_norm_BLOSUM62_GO_-10_GE_-1_rank	0.018
9 parsimony_max	0.009	FoldMason column score_rank	0.015
10 gaps_len_four_plus	0.009	sp_HENIKOFF_with_gaps_BLOSUM62_GO_-6_GE_-0.2_rank	0.015

Simulated data			
Model 1		Model 2	
Feature	Mean absolute SHAP value	Feature (rank-normalized)	Mean absolute SHAP value
1 constant_sites_pct	0.013	entropy_sum_rank	0.049
2 entropy_sum	0.013	sp_ge_rank	0.029
3 num_cols_all_gaps_except1	0.012	sp_BLOSUM62_GO_-6_GE_-0.2_rank	0.029
4 parsimony_mean	0.010	sp_mismatch_count_rank	0.028
5 k_mer_average_K5	0.009	sp_match_count_rank	0.022
6 sp_mismatch_norm_PAM250	0.009	sp_PAM250_GO_-6_GE_-0.2_rank	0.018
7 sp_go_norm	0.009	sp_HENIKOFF_with_gaps_BLOSUM62_GO_-6_GE_-0.2_rank	0.017
8 n_unique_sites	0.007	sp_go_norm_rank	0.016
9 bl_mean	0.007	gaps_1seq_len4plus_rank	0.015
10 seq_min_len	0.006	sp_match_BLOSUM62_rank	0.014

The SHAP values were extracted based on the random subset of 500 MSAs from the test set batches.

alignments. In principle, a scoring function can also be used to rank alignments by sorting their predicted scores. However, our results show that while *Model 1* correlates strongly with overall alignment quality, it did not perform well in distinguishing among top-scoring MSAs. This limitation likely arises because *Model 1* was not explicitly trained to discriminate between closely related, high-quality alignments. Consequently, the correlation between predicted and true quality decreases substantially when only top-ranked MSAs are considered. These observations motivated the development of *Model 2*, which uses a dedicated loss function that optimizes relative ordering among MSAs and assigns a higher weight to MSAs that are close to the reference alignment. This approach reflects the practical goal of most applications: to identify a single high-quality MSA suitable for downstream analyses, rather than to assign absolute scores to all possible alignments. A key innovation of our framework, therefore, is that unlike SoP and related metrics, *Model 2* explicitly optimizes for ranking accuracy rather than for absolute scoring.

To our knowledge, this study is the first to employ deep learning to learn an MSA ranking function. A major distinction between our approach and classic scoring schemes, such as SoP

and its variants, is that we use supervised learning to directly train the ranking model. This allows the model to capture complex dependencies among features, thus integrating over multiple characteristics of an MSA and not just the inventory of the number of gaps and substitutions. ML methods are increasingly being applied to algorithmic challenges in molecular phylogenetics (Azouri et al. 2021, 2024; Smith and Hahn 2023; Mo et al. 2024). Here, we demonstrate that such approaches can also address one of the field's most persistent problems, MSA inference. Notably, combining our trained model with the Guidance-based generation of alternative MSAs enabled the selection of alignments that are, on average, more accurate than those produced by widely used programs (in their default settings), including MAFFT, PRANK, and Muscle5, across both simulated and empirical datasets.

We designed over 150 handcrafted features that capture diverse structural and statistical properties of both the MSA and its underlying phylogenetic tree. These features allow our models to overcome several limitations of traditional scoring methods, such as SoP, which depend on a single affine-gap and substitution model to evaluate pairwise alignments. Many of our features generalize this concept by varying substitution matrices and gap

penalties. For example, Feature 27 (“sp_BLOSUM62_GO_6_GE_-0.2”) represents the SoP score computed using the BLOSUM62 matrix with a gap opening penalty of -6 and a gap extension penalty of -0.2 , whereas Feature 37 (“sp_PAM250_GO_10_GE_-1”) uses the PAM250 matrix with a gap opening penalty of -10 and extension penalty of -1 (see [Information S1](#) for details). By learning dependencies among such features, the model can adaptively downweigh Feature 27 when few long indels are present while assigning greater importance to Feature 37. The model learns whether there are few or many long indels, from features, such as “gaps_len_four_plus,” which counts gaps of length ≥ 4 , and “avg_unique_gap_length,” which quantifies the average length of unique indels. As part of feature extraction, we also compute a neighbor-joining tree and calculate parsimony scores for each alignment column, eg “parsimony_mean,” which measures the mean parsimony score across all columns. Together, our features capture both parsimony-based concepts underlying early alignment algorithms and SoP-based measures that characterize most modern MSA inference methods.

In this study, we trained each model twice: once using simulated datasets and once using empirical datasets. We observed that models trained on simulated data performed less effectively when applied to empirical data and vice versa. Additionally, *Model 2* outperformed all aligners, except BALi-Phy. These findings are consistent with previous studies demonstrating fundamental differences between simulated and empirical datasets. For example, [Trost et al. \(2024\)](#) showed that a ML classifier can readily distinguish between empirical and simulated MSAs. Another important distinction between the simulation-trained and empirically-trained models lies in the features used during training. The empirical models incorporated structure-based features, which are irrelevant to the simulation-trained models because the simulated datasets did not account for protein structural constraints. Finally, some of the apparent prediction errors observed for the empirical models may stem from inaccuracies in the empirical reference alignments themselves. Unlike simulated MSAs, which have a known ground truth, the reference alignments in datasets, such as BALiBASE, are inferred and therefore cannot be regarded as genuine reference MSAs ([Edgar 2010](#); [Iantorno et al. 2014](#)). Despite these differences between models trained on simulated versus empirical MSAs, our results clearly show that accounting for multiple features capturing global aspects of the MSA within a deep learning inference framework can be highly efficient in ranking both types of alternative MSAs and thus help improve the accuracy of MSA inference.

We have shown that *Model 2* enables selecting a single MSA that is more accurate than several widely used alignment programs. From the practical aspect, however, this necessitates generating alternative MSAs and running the trained model on each of them to select the best one, which is computationally costly. Relatively straightforward modifications can potentially reduce running times, eg discarding some of the alternative MSAs based on a subset of features and extracting the entire set of informative features only from those alternative MSAs that pass the first filtering step.

Additional types of features could further enhance model performance. For instance, when analyzing empirical data, one could incorporate features derived from predicted three-dimensional protein structures generated by tools, such as AlphaFold3 ([Abramson et al. 2024](#)), though this approach would

incur substantial computational cost. A promising alternative is to use embedding vectors from protein language models, which capture rich contextual information about amino acid sequences ([Rao et al. 2021](#); [Rives et al. 2021](#); [Brandes et al. 2022](#); [Lin et al. 2023](#); [Heinzinger et al. 2024](#); [Hayes et al. 2025](#)). Because several features in our current approach depend on the assumed phylogenetic tree, a fully end-to-end framework would also require direct embedding of the tree. Nevertheless, handcrafted features retain important advantages in interpretability. They facilitate the identification of systematic failure modes and provide biological insight into model behavior. For example, in cases of long-branch attraction ([Susko and Roger 2021](#)), deviations in tree-based features can reveal the source of alignment error, an effect that might be obscured when using purely learned representations.

A potential limitation of our framework is its assumption that MSAs are constructed from correctly inferred homologous sequences, whereas in practice, many alignments are derived from predicted gene models. Errors in gene prediction, such as incorrect exon boundaries, exon fragmentation, or inclusion of nonhomologous regions, can introduce systematic biases into the input alignments. These errors may lead to misleading alignment features that are unrelated to true evolutionary signal, potentially affecting model predictions. While our approach remains applicable in such settings, its performance will depend on the quality of the underlying gene annotations, highlighting the importance of careful pre-processing and orthology validation when applying MSA-ranking methods to genome-derived datasets.

Our simulations did not explicitly incorporate low-complexity regions ([Enright et al. 2023](#)). To our knowledge, there is currently no widely used evolutionary simulator that realistically models the emergence and evolution of low-complexity sequence features along a phylogeny. As a result, such regions were not represented in the training data for our models. Low-complexity regions are known to challenge alignment methods and may introduce patterns that differ from those captured during model training, potentially affecting prediction accuracy. We therefore expect that model performance in the presence of such regions could differ from that reported here. Incorporating low-complexity evolution into training data would likely improve robustness, but this will require the development of appropriate simulation frameworks or curated empirical benchmarks.

Another potential limitation of our simulation-based *Model 2* is that it is trained and tested on MSAs generated under independent and substitution stochastic models that fail to capture several key aspects of molecular sequence evolution. For instance, the substitution process assumes a constant rate within each site, even though it is well established that evolutionary rates can vary over time at the same site, a phenomenon known as heterotachy ([Lopez et al. 2002](#); [Galtier and Jean-Marie 2004](#)). Similarly, given the tree topology and site-specific rates, our simulator models substitutions at different alignment positions as independent events, thereby ignoring potential co-evolution between residues. When the three-dimensional structure of a protein is known, incorporating structural constraints into the simulations would provide a more biologically realistic representation of sequence evolution ([Robinson et al. 2003](#); [Rodrigue et al. 2005](#); [Echave and Wilke 2017](#); [Ferreiro et al. 2026](#)). In addition, although the indel model used here allows for distinct insertion and deletion rates ([Loewenthal et al. 2021](#)), it overlooks several well-documented aspects of indel evolutionary dynamics ([Redelings](#)

et al. 2024). For example, the current model assumes that indel probability is uniform across all alignment positions, implying no among-site variation in indel rate. This assumption is clearly oversimplified, as indels are typically rare in the structurally constrained core of proteins and more frequent on protein surfaces (Kim and Guo 2010). Overall, increasing the biological realism of simulated MSAs represents an important direction for future work. Such improvements are likely to enhance not only the accuracy of deep learning-based MSA inference but also the performance of downstream deep learning-based analyses in molecular evolution, including phylogenetic tree reconstruction, estimation of phylodynamic model parameters (Voznica et al. 2022; Lambert et al. 2023), and model selection (Abadi et al. 2020; Burgstaller-Muehlbacher et al. 2023; Kulikov et al. 2024; Braichenko et al. 2025).

For training the deep learning models, we applied fully connected networks with a handful of hidden layers. The loss function for *Model 2* was relatively ad hoc, aiming to give more weight to erroneous ranking of alternative MSAs that are close to the reference one than the rest of the alternative alignments. A few networks dedicated to ranking were previously suggested (Burges et al. 2005; Chen et al. 2009; Brinker and Hüllermeier 2020; He et al. 2022). In addition, transformers were shown to be a promising architecture for models that use representation learning. We predict that a substantial increase in model performance can be obtained by exploring alternative network configurations and loss functions, following innovations in the very fast-developing field of AI research.

Further research is required to fully characterize and optimize the proposed methodology. For instance, when training an MSA ranking model, it remains unclear whether performance would benefit more from increasing the number of alternative MSAs within each MSA-batch or from expanding the total number of MSA-batches. Another open question concerns dataset composition: Should “easy” cases, such as alignments with low indel rates, be excluded from the training and validation sets to focus the model on more challenging examples? Furthermore, we generated alternative MSAs in equal proportions using the alignment tools PRANK, MAFFT, Muscle, and BALi-Phy. It is possible, however, that some alignment methods should be more heavily represented within each training batch than others, depending on their error characteristics or similarity to the true alignment. Systematic exploration of these design parameters will be valuable for improving both training efficiency and model generalizability.

Model 2 currently ranks a limited set of alternative MSAs for each group of unaligned sequences, leaving many potentially high-scoring alignments unexplored. In this study, alternative MSAs were generated, for example, using GUIDANCE2 (Sela et al. 2015) by perturbing guide trees, adjusting gap penalties, and sampling co-optimal solutions within progressive alignment frameworks. Each resulting MSA therefore represents a distinct combination of these alignment parameters. This procedure could be extended into an iterative refinement framework. Specifically, one could begin by sampling alternative MSAs, use *Model 2* to identify the top candidates, and then generate new alternatives by applying targeted perturbations to these top-performing MSAs (eg small modifications to guide trees or gap penalties). This cycle would iteratively refine the candidate pool while balancing exploration and exploitation. Additional promising MSAs might also be generated by combining

information from several top-ranking alignments (Collingridge and Kelly 2012). Alternative refinement algorithms can also be used to search for neighboring MSAs with a high *Model 2* score (Gotoh 1996; Thompson et al. 2003; Gotoh 2014; Katoh and Standley 2014; Mokaddem et al. 2019). Such iterative processes terminate once no additional high-scoring MSAs are found.

In this work, we focused on identifying a single MSA that best represents the underlying homology relationships among the analyzed sequences. However, when the ultimate goal is phylogenetic reconstruction, it is often advantageous to account for alignment uncertainty rather than relying on a single, fixed MSA (Wheeler et al. 1995; Sela et al. 2015). More generally, the Bayesian framework for phylogenetic inference advocates integrating, or averaging, over sources of uncertainty, such as model parameters and alignment uncertainty, rather than conditioning on point estimates (Lunter et al. 2005; Hohna et al. 2016; Baele et al. 2017; Nascimento et al. 2017). A promising future direction is the development of generative deep learning models that take as input a set of unaligned sequences and generate samples of alternative alignments according to their posterior probabilities.

Acknowledgments

K.P. and N.S. were supported by fellowships from the Edmond J. Safra Center for Bioinformatics at Tel Aviv University. K.P. was also supported by the Khazanov scholarship program and the TAD Excellence Program for Doctoral Students in Artificial Intelligence and Data Science.

Supplementary material

Supplementary material is available at *Molecular Biology and Evolution* online.

Funding

T.P. was supported by Israel Science Foundation grants 2818/21 and 1789/26.

Conflicts of interest

None declared.

Data availability

All analyses were implemented in Python (v3.10), except for the feature extraction, which was also implemented in C++. The source code and the OrthoMaM and BALiBASE dataset entries used in this study are publicly available on GitHub (https://github.com/nimrodSerokTAU/sp_alternative and https://github.com/nimrodSerokTAU/sp_alternative_cpp, for Python and C++ implementations, respectively). In the Python GitHub, we provide detailed, step-by-step instructions for reproducing a representative example corresponding to simulated MSA-batch 23410, which was used to generate Fig. 1b and 2b. This dataset contains 1,604 alternative MSAs, along with the corresponding

true (simulated) MSA and the phylogenetic tree used to generate them.

References

- Abadi M *et al.* TensorFlow: a system for large-scale machine learning. In: *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation. OSDI'16.* USENIX Association; 2016. p. 265–283.
- Abadi S, Avram O, Rosset S, Pupko T, Mayrose I. ModelTeller: model selection for optimal phylogenetic reconstruction using machine learning. *Mol Biol Evol.* 2020;37:3338–3352. <https://doi.org/10.1093/molbev/msaa154>.
- Abramson J *et al.* Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature.* 2024;630:493–500. <https://doi.org/10.1038/s41586-024-07487-w>.
- Akiba T, Sano S, Yanase T, Ohta T, Koyama M. Optuna: a next-generation hyperparameter optimization framework. In: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* Association for Computing Machinery (ACM), New York, NY, USA; 2019. p. 2623–2631.
- Allio R *et al.* Orthomam v12: a database of curated single-copy ortholog alignments and trees to study mammalian evolutionary genomics. *Nucleic Acids Res.* 2024;52:D529–D535. <https://doi.org/10.1093/nar/gkad834>.
- Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. *J Mol Biol.* 1990;215:403–410. [https://doi.org/10.1016/S0022-2836\(05\)80360-2](https://doi.org/10.1016/S0022-2836(05)80360-2).
- Ashkenazy H, Levy Karin E, Mertens Z, Cartwright RA, Pupko T. SpartaABC: a web server to simulate sequences with indel parameters inferred using an approximate Bayesian computation algorithm. *Nucleic Acids Res.* 2017;45:W453–W457. <https://doi.org/10.1093/nar/gkx322>.
- Azouri D *et al.* The tree reconstruction game: phylogenetic reconstruction using reinforcement learning. *Mol Biol Evol.* 2024;41:msae105. <https://doi.org/10.1093/molbev/msae105>.
- Azouri D, Abadi S, Mansour Y, Mayrose I, Pupko T. Harnessing machine learning to guide phylogenetic-tree search algorithms. *Nat Commun.* 2021;12:1983. <https://doi.org/10.1038/s41467-021-22073-8>.
- Baele G, Lemey P, Rambaut A, Suchard MA. Adaptive MCMC in Bayesian phylogenetics: an application to analyzing partitioned data in BEAST. *Bioinformatics.* 2017;33:1798–1805. <https://doi.org/10.1093/bioinformatics/btx088>.
- Blackburne BP, Whelan S. Measuring the distance between multiple sequence alignments. *Bioinformatics.* 2012;28:495–502. <https://doi.org/10.1093/bioinformatics/btr701>.
- Braichenko S, Borges R, Kosiol C. Phylogenetic methods meet deep learning. *Genome Biol Evol.* 2025;17:evaf177. <https://doi.org/10.1093/gbe/evaf177>.
- Brandes N, Ofer D, Peleg Y, Rappoport N, Linial M. ProteinBERT: a universal deep-learning model of protein sequence and function. *Bioinformatics.* 2022;38:2102–2110. <https://doi.org/10.1093/bioinformatics/btac020>.
- Brinker K, Hüllermeier E. A reduction of label ranking to multi-class classification. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).* Springer; 2020. p. 204–219.
- Buchfink B, Xie C, Huson DH. Fast and sensitive protein alignment using DIAMOND. *Nat Methods.* 2015;12:59–60. <https://doi.org/10.1038/nmeth.3176>.
- Burges CJC, Shaked T, Renshaw E, Lazier A, Deeds M, Hamilton N, Hullender, GN. Learning to rank using gradient descent. In: *ICML 2005 - Proceedings of the 22nd International Conference on Machine Learning.* Association for Computing Machinery (ACM), New York, NY, USA; 2005. p. 89–96.
- Burgstaller-Muehlbacher S *et al.* ModelRevelator: fast phylogenetic model estimation via deep learning. *Mol Phylogenet Evol.* 2023;188:107905. <https://doi.org/10.1016/j.ympev.2023.107905>.
- Carrillo H, Lipman D. The multiple sequence alignment problem in biology. *SIAM J Appl Math.* 1988;48:1073–1082. <https://doi.org/10.1137/0148063>.
- Chen W, Liu TY, Lan Y, Ma Z, Li H. Ranking measures and loss functions in learning to rank. In: *Advances in Neural Information Processing Systems 22 - Proceedings of the 2009 Conference,* Curran Associates, Inc; 2009. p. 315–323.
- Chollet F. 2018. Keras: the Python deep learning API. Astrophysics Source Code Library, record ascl:1806.022 [Internet]. Available from: <https://ui.adsabs.harvard.edu/abs/2018ascl.soft06022C>. Accessed September 11, 2024
- Collingridge PW, Kelly S. MergeAlign: improving multiple sequence alignment performance by dynamic reconstruction of consensus multiple sequence alignments. *BMC Bioinf.* 2012;13:117. <https://doi.org/10.1186/1471-2105-13-117>.
- Dotan E *et al.* BetaAlign: a deep learning approach for multiple sequence alignment. *Bioinformatics.* 2024;41:btaf009. <https://doi.org/10.1093/bioinformatics/btaf009>.
- Echave J, Wilke CO. Biophysical models of protein evolution: understanding the patterns of evolutionary sequence divergence. *Annu Rev Biophys.* 2017;46:85–103. <https://doi.org/10.1146/annurev-biophys-070816-033819>.
- Ecker N, Huchon D, Mansour Y, Mayrose I, Pupko T. A machine-learning-based alternative to phylogenetic bootstrap. *Bioinformatics.* 2024;40:i208–i217. <https://doi.org/10.1093/bioinformatics/btae255>.
- Edgar RC. Quality measures for protein alignment benchmarks. *Nucleic Acids Res.* 2010;38:2145–2153. <https://doi.org/10.1093/nar/gkp1196>.
- Edgar RC. Muscle5: high-accuracy alignment ensembles enable unbiased assessments of sequence homology and phylogeny. *Nat Commun.* 2022;13:6968. <https://doi.org/10.1038/s41467-022-34630-w>.
- Enard W *et al.* Molecular evolution of FOXP2, a gene involved in speech and language. *Nature.* 2002;418:869–872. <https://doi.org/10.1038/nature01025>.
- Enright JM, Dickson ZW, Golding GB, Ozkan B. Low complexity regions in proteins and DNA are poorly correlated. *Molec Biol Evol.* 2023;40:549. <https://doi.org/10.1093/molbev/msad084>.
- Feng DF, Doolittle RF. Progressive sequence alignment as a prerequisite to correct phylogenetic trees. *J Mol Evol.* 1987;25:351–360. <https://doi.org/10.1007/BF02603120>.
- Ferreiro D, Pazos E, Arenas M. Trends in substitution models of protein evolution for phylogenetic inference. *Mol Phylogenet Evol.* 2026;214:108473. <https://doi.org/10.1016/j.ympev.2025.108473>.

- Fletcher W, Yang Z. INDELible: a flexible simulator of biological sequence evolution. *Mol Biol Evol*. 2009;26:1879–1888. <https://doi.org/10.1093/molbev/msp098>.
- Galtier N, Jean-Marie A. Markov-modulated Markov chains and the covarian process of molecular evolution. *J Comput Biol*. 2004;11:727–733. <https://doi.org/10.1089/cmb.2004.11.727>.
- Gassner FJ et al. Imprecision and DNA break repair biased towards incompatible end joining in leukemia. *Mol Cancer Res*. 2018;16:428–438. <https://doi.org/10.1158/1541-7786.MCR-17-0373>.
- Gondro C, Kinghorn BP. A simple genetic algorithm for multiple sequence alignment. *Genet Mol Res*. 2007;6:964–982.
- Gotoh O. An improved algorithm for matching biological sequences. *J Mol Biol*. 1982;162:705–708. [https://doi.org/10.1016/0022-2836\(82\)90398-9](https://doi.org/10.1016/0022-2836(82)90398-9).
- Gotoh O. Significant improvement in accuracy of multiple protein sequence alignments by iterative refinement as assessed by reference to structural alignments. *J Mol Biol*. 1996;264:823–838. <https://doi.org/10.1006/jmbi.1996.0679>.
- Gotoh O. Heuristic alignment methods. In: Russell DJ, editors. *Multiple sequence alignment methods*. Humana Press; 2014. p. 29–43.
- Gupta M, Zaharias P, Warnow T. Accurate large-scale phylogeny-aware alignment using BALI-Phy. *Bioinformatics*. 2021;37:4677–4683. <https://doi.org/10.1093/bioinformatics/btab555>.
- Haag J, Hühler D, Bettisworth B, Stamatakis A. From easy to hopeless-predicting the difficulty of phylogenetic analyses. *Mol Biol Evol*. 2022;39:msac254. <https://doi.org/10.1093/molbev/msac254>.
- Hasegawa H, Holm L. Advances and pitfalls of protein structural alignment. *Curr Opin Struct Biol*. 2009;19:341–348. <https://doi.org/10.1016/j.sbi.2009.04.003>.
- Hayes T et al. Simulating 500 million years of evolution with a language model. *Science*. 2025;387:850–858. <https://doi.org/10.1126/science.ads0018>.
- He Y et al. GNNRank: learning global rankings from pairwise comparisons via directed graph neural networks. In: *Proceedings of Machine Learning Research*. 2022. p. 8581–8612.
- Hein J, Wiuf C, Knudsen B, Møller MB, Wibling G. Statistical alignment: computational properties, homology testing and goodness-of-fit. *J Mol Biol*. 2000;302:265–279. <https://doi.org/10.1006/jmbi.2000.4061>.
- Heinzinger M et al. Bilingual language model for protein sequence and structure. *NAR Genom Bioinform*. 2024;6:lqae150. <https://doi.org/10.1093/nargab/lqae150>.
- Hertz GZ, Stormo GD. Identifying DNA and protein patterns with statistically significant alignments of multiple sequences. *Bioinformatics*. 1999;15:563–577. <https://doi.org/10.1093/bioinformatics/15.7.563>.
- Hirosawa M, Totoki Y, Hoshida M, Ishikawa M. Comprehensive study on iterative algorithms of multiple sequence alignment. *Comput Appl Biosci*. 1995;11:13–18. <https://doi.org/10.1093/bioinformatics/11.1.13>.
- Hohna S et al. RevBayes: Bayesian phylogenetic inference using graphical models and an interactive model-specification language. *Syst Biol*. 2016;65:726–736. <https://doi.org/10.1093/sysbio/syw021>.
- Huerta-Cepas J, Serra F, Bork P. ETE 3: reconstruction, analysis, and visualization of phylogenomic data. *Mol Biol Evol*. 2016;33:1635–1638. <https://doi.org/10.1093/molbev/msw046>.
- Iantorno S, Gori K, Goldman N, Gil M, Dessimoz C. Who watches the watchmen? An appraisal of benchmarks for multiple sequence alignment. *Methods Mol Biol*. 2014;1079:59–73. https://doi.org/10.1007/978-1-62703-646-7_4.
- Jumper J et al. Highly accurate protein structure prediction with AlphaFold. *Nature*. 2021;596:583–589. <https://doi.org/10.1038/s41586-021-03819-2>.
- Katoh K. MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Res*. 2002;30:3059–3066. <https://doi.org/10.1093/nar/gkf436>.
- Katoh K, Standley DM. MAFFT multiple sequence alignment software version 7: improvements in performance and usability. *Mol Biol Evol*. 2013;30:772–780. <https://doi.org/10.1093/molbev/mst010>.
- Katoh K, Standley DM. MAFFT: iterative refinement and additional methods. In: Russell DJ, editors. *Multiple sequence alignment methods*. Humana Press; 2014. p. 131–146.
- Kemena C, Notredame C. Upcoming challenges for multiple sequence alignment methods in the high-throughput era. *Bioinformatics*. 2009;25:2455–2465. <https://doi.org/10.1093/bioinformatics/btp452>.
- Kemena C, Taly J-F, Kleinjung J, Notredame C. STRIKE: evaluation of protein MSAs using a single 3D structure. *Bioinformatics*. 2011;27:3385–3391. <https://doi.org/10.1093/bioinformatics/btr587>.
- Kendall MG. A new measure of rank correlation. *Biometrika*. 1938;30:81–93. <https://doi.org/10.1093/biomet/30.1-2.81>.
- Kim J, Pramanik S, Chung MJ. Multiple sequence alignment using simulated annealing. *Comput Appl Biosci*. 1994;10:419–426. <https://doi.org/10.1093/bioinformatics/10.4.419>.
- Kim R, Guo J. Systematic analysis of short internal indels and their impact on protein folding. *BMC Struct Biol*. 2010;10:24. <https://doi.org/10.1186/1472-6807-10-24>.
- Kulikov N, Derakhshandeh F, Mayer C. Machine learning can be as good as maximum likelihood when reconstructing phylogenetic trees and determining the best evolutionary model on four taxon alignments. *Mol Phylogenet Evol*. 2024;200:108181. <https://doi.org/10.1016/j.ympev.2024.108181>.
- Lambert S, Voznica J, Morlon H. Deep learning from phylogenies for diversification analyses. *Syst Biol*. 2023;72:1262–1279. <https://doi.org/10.1093/sysbio/syad044>.
- Leuchtenberger AF et al. Distinguishing Felsenstein zone from Farris zone using neural networks. *Mol Biol Evol*. 2020;37:3632–3641. <https://doi.org/10.1093/molbev/msaa164>.
- Lin Z et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science* (1979). 2023;379:1123–1130. doi:<https://doi.org/10.1126/science.ade2574>.
- Loewenthal G et al. A probabilistic model for indel evolution: differentiating insertions from deletions. *Mol Biol Evol*. 2021;38:5769–5781. <https://doi.org/10.1093/molbev/msab266>.
- Lopez P, Casane D, Philippe H. Heterotachy, an important process of protein evolution. *Mol Biol Evol*. 2002;19:1–7. <https://doi.org/10.1093/oxfordjournals.molbev.a003973>.
- Löytynoja A. Phylogeny-aware alignment with PRANK. In: Russell DJ, editor. *Multiple sequence alignment methods*. 1079: Humana Press; 2014. p. 155–170. https://doi.org/10.1007/978-1-62703-646-7_10.
- Löytynoja A. Phylogeny-aware alignment with PRANK and PAGAN. In: Katoh K, editor. *Multiple sequence alignment*:

- methods and protocols. 2231: Springer US; 2021. p. 17–37. https://doi.org/10.1007/978-1-0716-1036-7_2.
- Lundberg SM, Lee SI. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*; 2017. p. 4768–4777.
- Lunter G, Drummond AJ, Miklós I, Hein J. *Statistical alignment: recent progress, new applications, and challenges*. Statistical Methods in Molecular Evolution; 2005. p. 375–405.
- Mirdita M, Steinegger M, Söding J. MMseqs2 desktop and local web server app for fast, interactive sequence searches. *Bioinformatics*. 2019;35:2856–2858. <https://doi.org/10.1093/bioinformatics/bty1057>.
- Mo YK, Hahn MW, Smith ML. Applications of machine learning in phylogenetics. *Mol Phylogenet Evol*. 2024;196:108066. <https://doi.org/10.1016/j.ympev.2024.108066>.
- Mokaddem A, Bel Hadj A, Elloumi M. Refin-Align: new refinement algorithm for multiple sequence alignment. *Informatica*. 2019;43:527–534. <https://doi.org/10.31449/inf.v43i4.2621>.
- Nascimento FF, Reis MD, Yang Z. A biologist's guide to Bayesian phylogenetic analysis. *Nat Ecol Evol*. 2017;1:1446–1454. <https://doi.org/10.1038/s41559-017-0280-x>.
- Needleman SB, Wunsch CD. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol*. 1970;48:443–453. [https://doi.org/10.1016/0022-2836\(70\)90057-4](https://doi.org/10.1016/0022-2836(70)90057-4).
- Nguyen KD, Pan Y. An improved scoring method for protein residue conservation and multiple sequence alignment. *IEEE Trans Nanobioscience*. 2011;10:275–285. <https://doi.org/10.1109/TNB.2011.2179553>.
- Notredame C, Higgins DG, Heringa J. T-coffee: a novel method for fast and accurate multiple sequence alignment. *J Mol Biol*. 2000;302:205–217. <https://doi.org/10.1006/jmbi.2000.4042>.
- Orobítg M, Cores F, Guirado F, Roig C, Notredame C. Improving multiple sequence alignment biological accuracy through genetic algorithms. *J Supercomput*. 2013;65:1076–1088. <https://doi.org/10.1007/s11227-012-0856-9>.
- Ortuño FM et al. Comparing different machine learning and mathematical regression models to evaluate multiple sequence alignments. *Neurocomputing (Amst)*. 2015;164:123–136. <https://doi.org/10.1016/j.neucom.2015.01.080>.
- Pekar JE et al. The molecular epidemiology of multiple zoonotic origins of SARS-CoV-2. *Science*. 2022;377:960–966. <https://doi.org/10.1126/science.abp8337>.
- Ranwez V et al. Orthomam: a database of orthologous genomic markers for placental mammal phylogenetics. *BMC Evol Biol*. 2007;7:241. <https://doi.org/10.1186/1471-2148-7-241>.
- Ranwez V. Two simple and efficient algorithms to compute the SP-score objective function of a multiple sequence alignment. *PLoS One*. 2016;11:e0160043. <https://doi.org/10.1371/journal.pone.0160043>.
- Rao R et al. MSA transformer. In: *Proceedings of the 38th International Conference on Machine Learning*. PMLR; 2021. p. 8844–8856.
- Redelings BD, Holmes I, Lunter G, Pupko T, Anisimova M. Insertions and deletions: computational methods, evolutionary dynamics, and biological applications. *Mol Biol Evol*. 2024;41:msae177. <https://doi.org/10.1093/molbev/msae177>.
- Rice P, Longden I, Bleasby A. EMBOSS: the European molecular biology open software suite. *Trends Genet*. 2000;16:276–277. [https://doi.org/10.1016/S0168-9525\(00\)00204-2](https://doi.org/10.1016/S0168-9525(00)00204-2).
- Rives A et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc Natl Acad Sci U S A*. 2021;118:e2016239118. <https://doi.org/10.1073/pnas.2016239118>.
- Robinson DM, Jones DT, Kishino H, Goldman N, Thorne JL. Protein evolution with dependence among codons due to tertiary structure. *Mol Biol Evol*. 2003;20:1692–1704. <https://doi.org/10.1093/molbev/msg184>.
- Rodrigue N, Lartillot N, Bryant D, Philippe H. Site interdependence attributed to tertiary structure in amino acid sequence evolution. *Gene*. 2005;347:207–217. <https://doi.org/10.1016/j.gene.2004.12.011>.
- Roettger M, Martin W, Dagan T. A machine-learning approach reveals that alignment properties alone can accurately predict inference of lateral gene transfer from discordant phylogenies. *Mol Biol Evol*. 2009;26:1931–1939. <https://doi.org/10.1093/molbev/msp105>.
- Saitou N, Nei M. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol Biol Evol*. 1987;4:406–425. <https://doi.org/10.1093/oxfordjournals.molbev.a040454>.
- Sankoff D. Matching sequences under deletion-insertion constraints. *Proc Natl Acad Sci U S A*. 1972;69:4–6. <https://doi.org/10.1073/pnas.69.1.4>.
- Sela I, Ashkenazy H, Katoh K, Pupko T. GUIDANCE2: accurate detection of unreliable alignment regions accounting for the uncertainty of multiple parameters. *Nucleic Acids Res*. 2015;43:W7–W14. <https://doi.org/10.1093/nar/gkv318>.
- Shenkin PS, Erman B, Mastrandrea LD. Information-theoretical entropy as a measure of sequence variability. *Proteins*. 1991;11:297–313. <https://doi.org/10.1002/prot.340110408>.
- Sierk ML, Smoot ME, Bass EJ, Pearson WR. Improving pairwise sequence alignment accuracy using near-optimal protein sequence alignments. *BMC Bioinf*. 2010;11:146. <https://doi.org/10.1186/1471-2105-11-146>.
- Sievers F, Higgins DG. The clustal omega multiple alignment package. *Methods Mol Biol*. 2021;2231:3–16. https://doi.org/10.1007/978-1-0716-1036-7_1.
- Smith ML, Hahn MW. Phylogenetic inference using generative adversarial networks. *Bioinformatics*. 2023;39:btad543. <https://doi.org/10.1093/bioinformatics/btad543>.
- Susko E, Roger AJ. Long branch attraction biases in phylogenetics. *Syst Biol*. 2021;70:838–843. <https://doi.org/10.1093/sysbio/syab001>.
- Thompson JD, Higgins DG, Gibson TJ. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res*. 1994;22:4673–4680. <https://doi.org/10.1093/nar/22.22.4673>.
- Thompson JD, Koehl P, Ripp R, Poch O. BALIBASE 3.0: latest developments of the multiple sequence alignment benchmark. *Proteins*. 2005;61:127–136. <https://doi.org/10.1002/prot.20527>.
- Thompson JD, Linard B, Lecompte O, Poch O. A comprehensive benchmark study of multiple sequence alignment methods: current challenges and future perspectives. *PLoS One*. 2011;6:e18093. <https://doi.org/10.1371/journal.pone.0018093>.
- Thompson JD, Plewniak F, Poch O. A comprehensive comparison of multiple sequence alignment programs. *Nucleic Acids Res*. 1999;27:2682–2690. <https://doi.org/10.1093/nar/27.13.2682>.

- Thompson JD, Plewniak F, Ripp R, Thierry JC, Poch O. Towards a reliable objective function for multiple sequence alignments. *J Mol Biol.* 2001;314:937–951. <https://doi.org/10.1006/jmbi.2001.5187>.
- Thompson JD, Thierry JC, Poch O. RASCAL: rapid scanning and correction of multiple sequence alignments. *Bioinformatics.* 2003;19:1155–1161. <https://doi.org/10.1093/bioinformatics/btg133>.
- Thorne JL, Kishino H, Felsenstein J. An evolutionary model for maximum likelihood alignment of DNA sequences. *J Mol Evol.* 1991;33:114–124. <https://doi.org/10.1007/BF02193625>.
- Tollis M, Schiffman JD, Boddy AM. Evolution of cancer suppression as revealed by mammalian comparative genomics. *Curr Opin Genet Dev.* 2017;42:40–47. <https://doi.org/10.1016/j.gde.2016.12.004>.
- Trivedi R, Nagarajaram HA. Substitution scoring matrices for proteins - an overview. *Protein Sci.* 2020;29:2150–2163. <https://doi.org/10.1002/pro.3954>.
- Trost J *et al.* Simulations of sequence evolution: how (un)realistic they are and why. *Mol Biol Evol.* 2024;41:msad277. <https://doi.org/10.1093/molbev/msad277>.
- Voznica J *et al.* Deep learning from phylogenies to uncover the epidemiological dynamics of outbreaks. *Nat Commun.* 2022;13:3896. <https://doi.org/10.1038/s41467-022-31511-0>.
- Wang L, Jiang T. On the complexity of multiple sequence alignment. *J Comput Biol.* 1994;1:337–348. <https://doi.org/10.1089/cmb.1994.1.337>.
- Warnow T. Revisiting evaluation of multiple sequence alignment methods. *Methods Mol Biol.* 2021;2231:299–317. https://doi.org/10.1007/978-1-0716-1036-7_17.
- Wheeler WC, Gatesy J, DeSalle R. Elision: a method for accommodating multiple molecular sequence alignments with alignment-ambiguous sites. *Mol Phylogenet Evol.* 1995;4:1–9. <https://doi.org/10.1006/mpev.1995.1001>.
- Wiegert J, Höhler D, Haag J, Stamatakis A. Predicting phylogenetic bootstrap values via machine learning. *Mol Biol Evol.* 2024;41:msae215. <https://doi.org/10.1093/molbev/msae215>.
- Zaharias P, Smirnov V, Warnow T. Large-scale multiple sequence alignment and the maximum weight trace alignment merging problem. *IEEE/ACM Trans Comput Biol Bioinform.* 2023;20:1700–1712. <https://doi.org/10.1109/TCBB.2022.3191848>.