

1023

## Supplementary Material

1024

### A Supplementary Figures

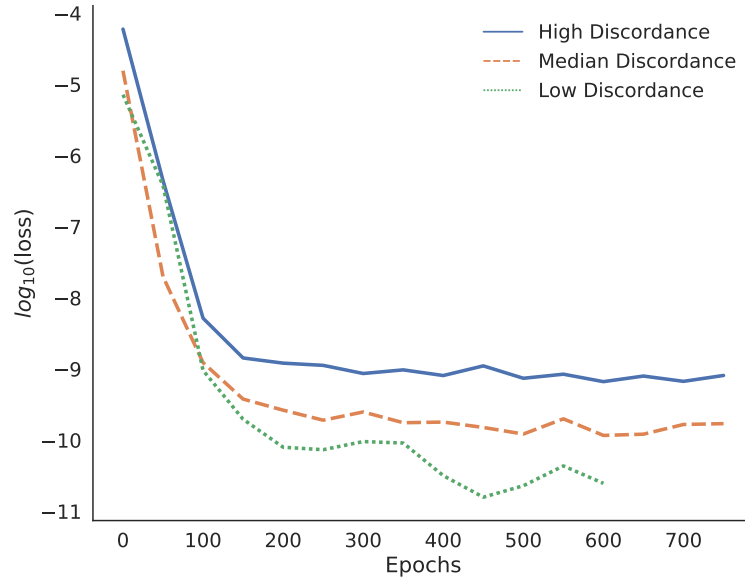


Figure S1: Train loss

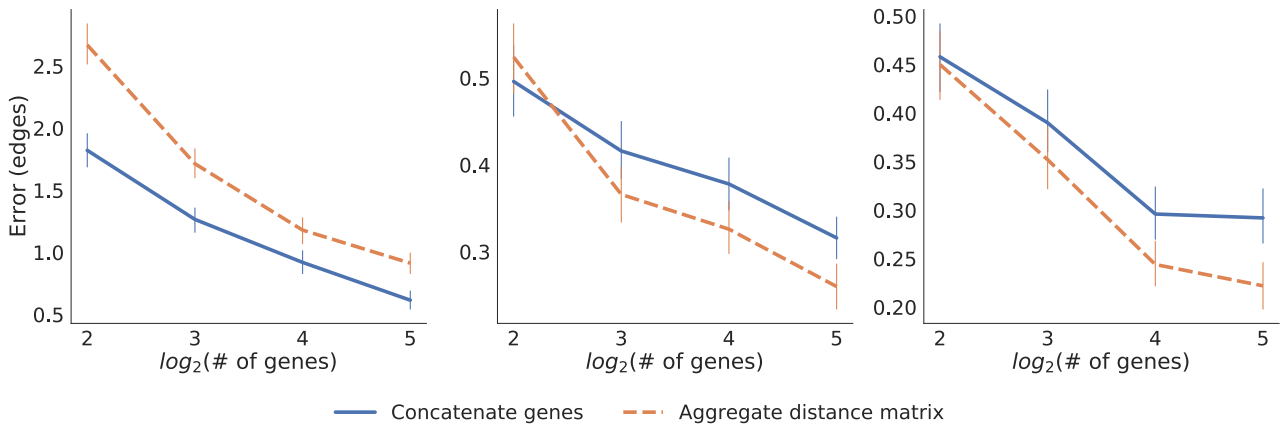


Figure S2: **Mean and standard error of placement error versus the number of genes for simulated data.** Blue: Results of single DEPP model trained by concatenating genes; Orange: Results of aggregating distance matrix from multiple DEPP models (one model per gene) by taking the mean distance among the interquartile range. Note that these analyses are using the older version of DEPP (v.0.1.56), and an older version of the dataset with ultrametric species trees used as the backbone (data release, v1.1.0). From left to right: high, medium, and low ILS.

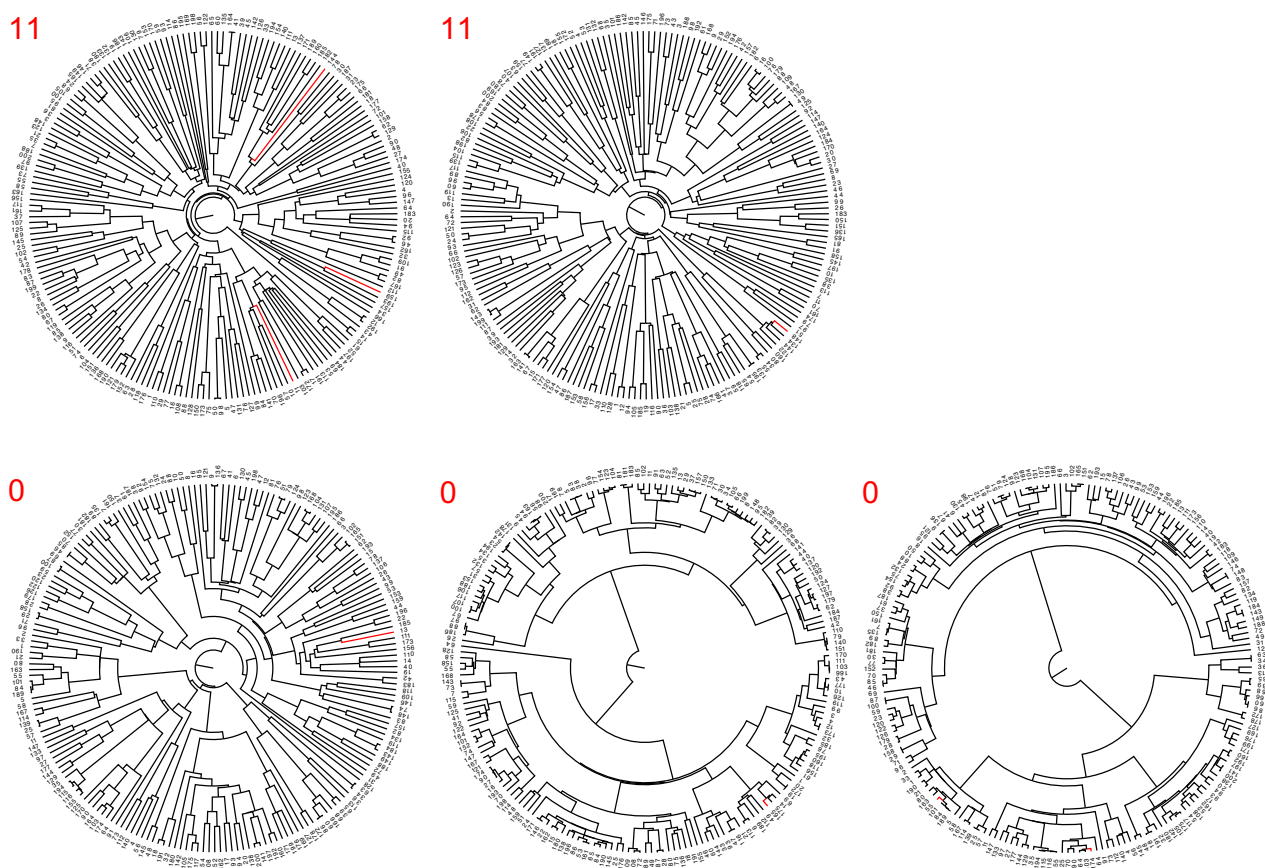


Figure S3: Examples of phylogenetic trees with high (top) and low (bottom) placement errors by DEPP. For each tree, we show the (correctly placed) query in red and show the error of DEPP on top in red.

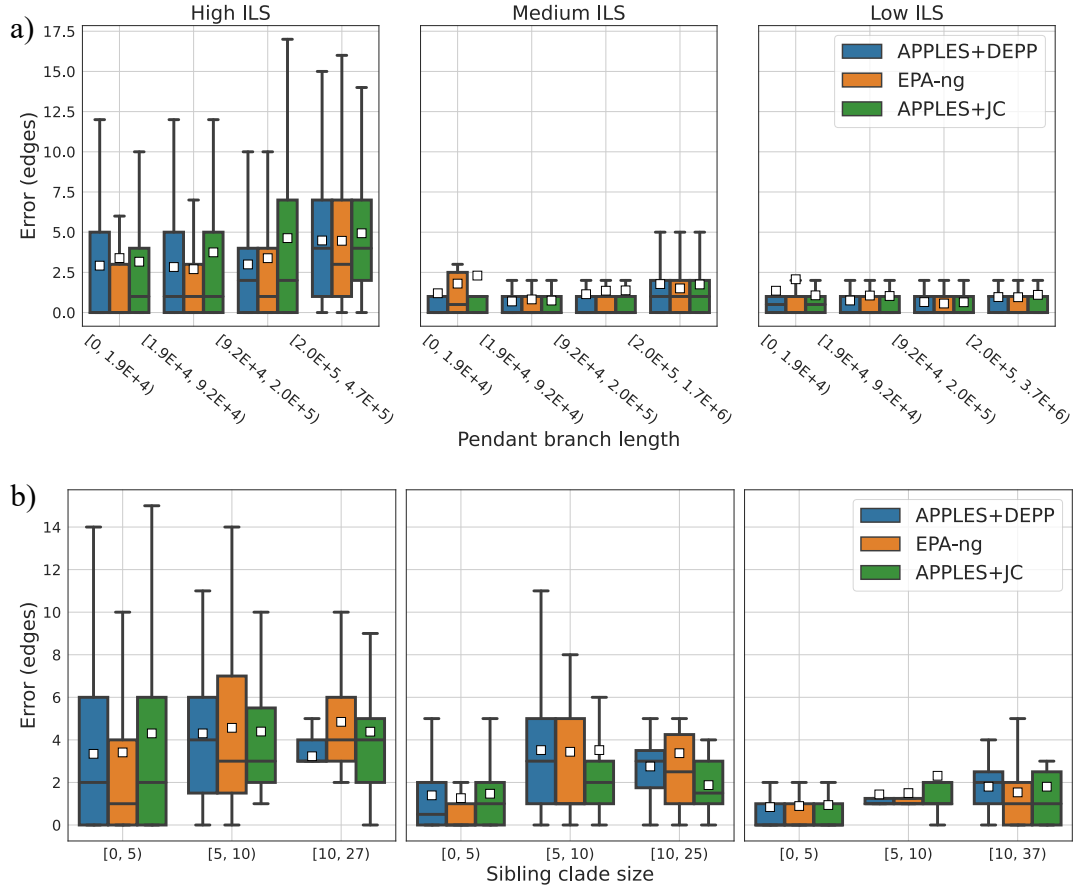


Figure S4: **Impact of query properties on error.** For each mode condition, we show the impact of the (a) length of the terminal branch of the query and (b) the number of leaves in the sibling clade of the query on the accuracy of all models when run with a single gene. The boxes correspond to levels of discordance, showing high, medium, and low from left to right.

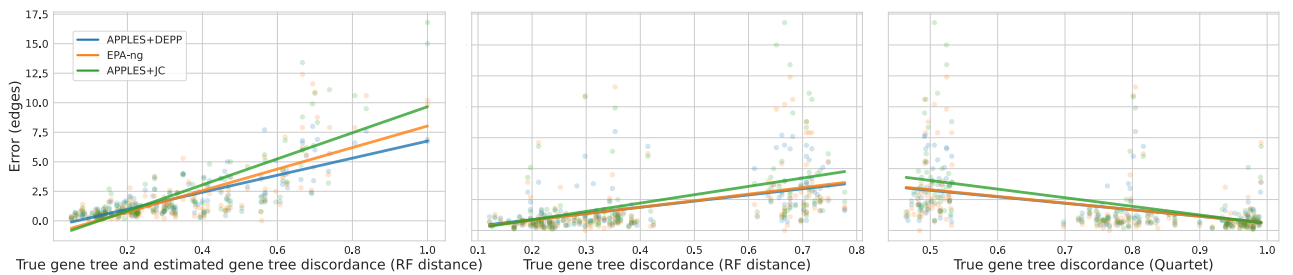


Figure S5: **The impact of true gene tree discordance and lack of signal** left: Impact of lacking signal measured as the RF distance between true gene trees and the estimated gene trees. Middle: Impact of true gene tree discordance measured as the RF distance between true gene trees and the species tree. Right: Impact of true gene tree discordance measured as the quartet score between true gene trees and the species tree, combining all discordance levels. Dots represent the mean of 10 queries for each of 148 replicate datasets.

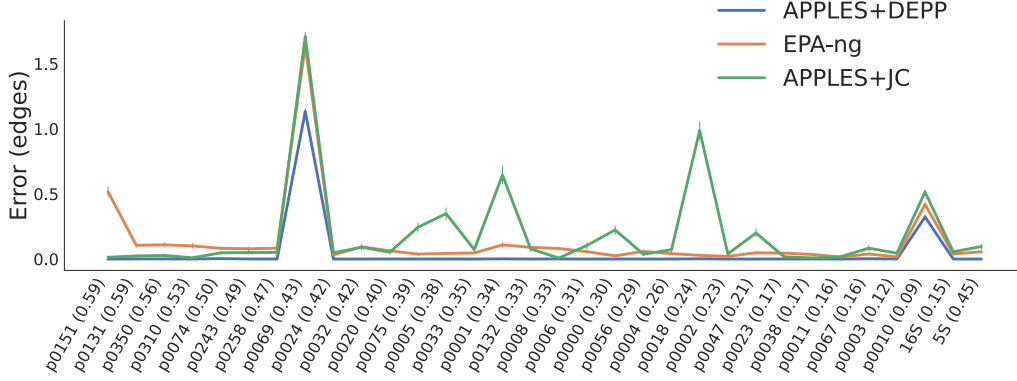


Figure S6: The mean and standard error of errors of APPLES+JC and EPA-ng versus DEPP when tested on known sequences (i.e., those in the reference set). Note that, unlike other genes, here, 16S and 5s have multiple copies (1.9 and 3.8 copies per species, respectively), causing the spike in the error.

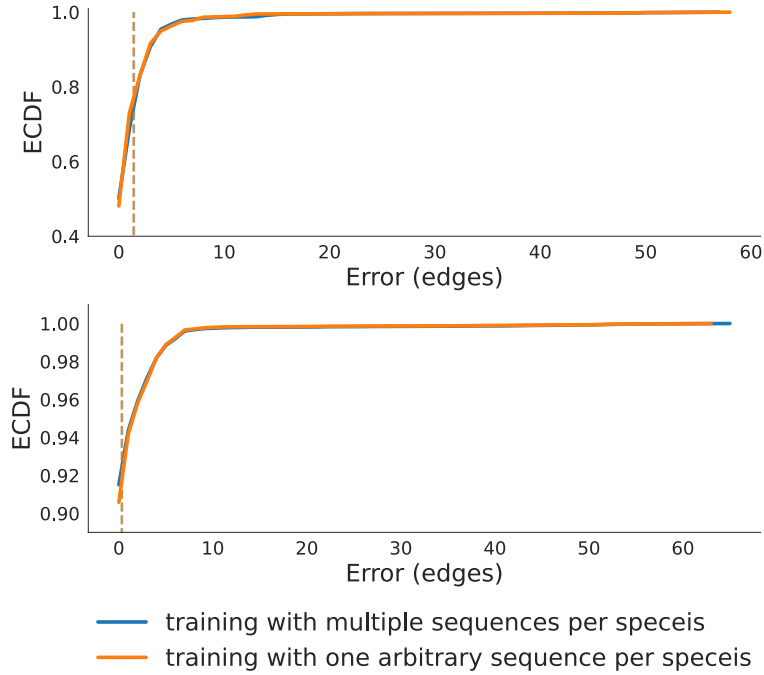


Figure S7: **Impact of weighted encoding on WoL 16S data.** The figure shows the comparison between training the model using all the available copies (using weighted encoding described in the text) versus selecting one arbitrary copy to represent each species. With multiple copies, sequenced are encoded by the frequency of the nucleotide character at each site. The results show that using more sequences per species has a very small impact on accuracy. Nevertheless, we use all the available sequences for training in all other experiments under the assumption that using all the sequences can reduce the variance of the DNN model. Results are based on version 1.0.0 of the DEPP WoL reference library.

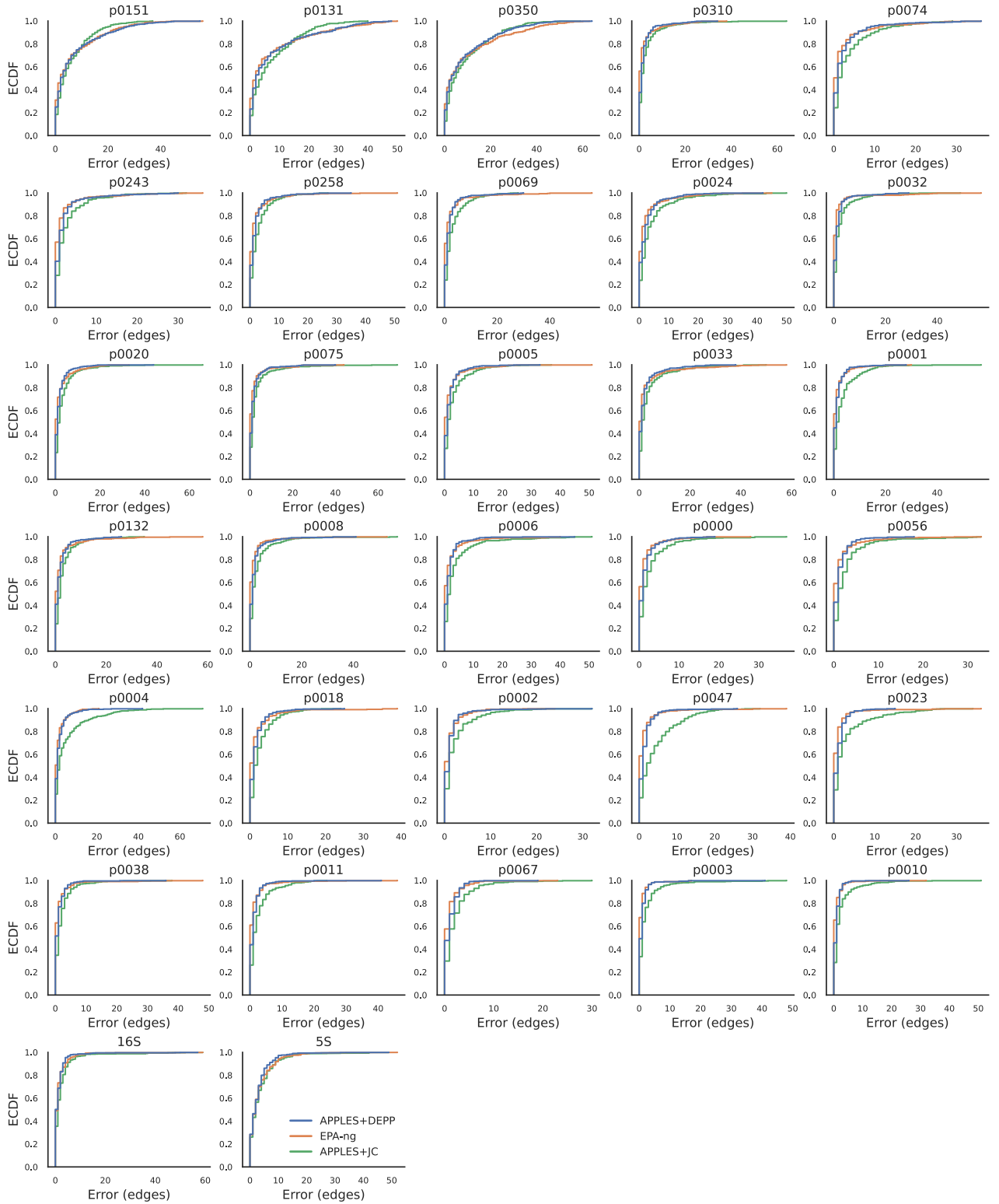


Figure S8: **Placement error on real microbial data in leave-out-experiments** We show the Empirical Cumulative Distribution (ECDF) of the placement error for JC and DEPP per gene case. The top two rows correspond to genes with high discordance, the next two rows medium discordance, and the next two rows low discordance.

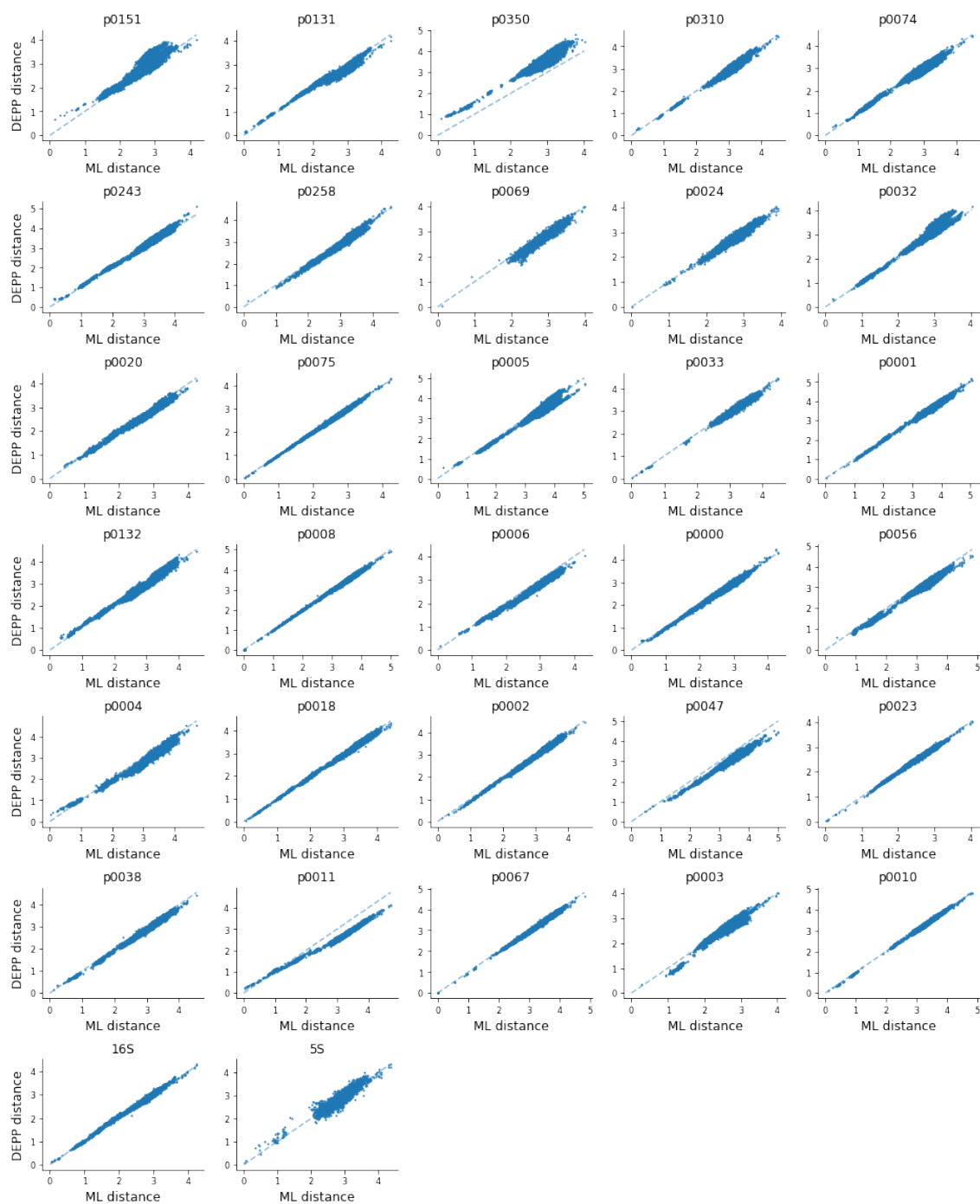


Figure S9: **Examples of distance calculation using DEPP with errors equal to zero** x-axis: the true phylogenetic distance between the query; y-axis: distance calculated by DEPP. Each panel is a query arbitrarily chosen from the experiment corresponding to the gene indicated by the panel's title. See Table S3 for gene properties.

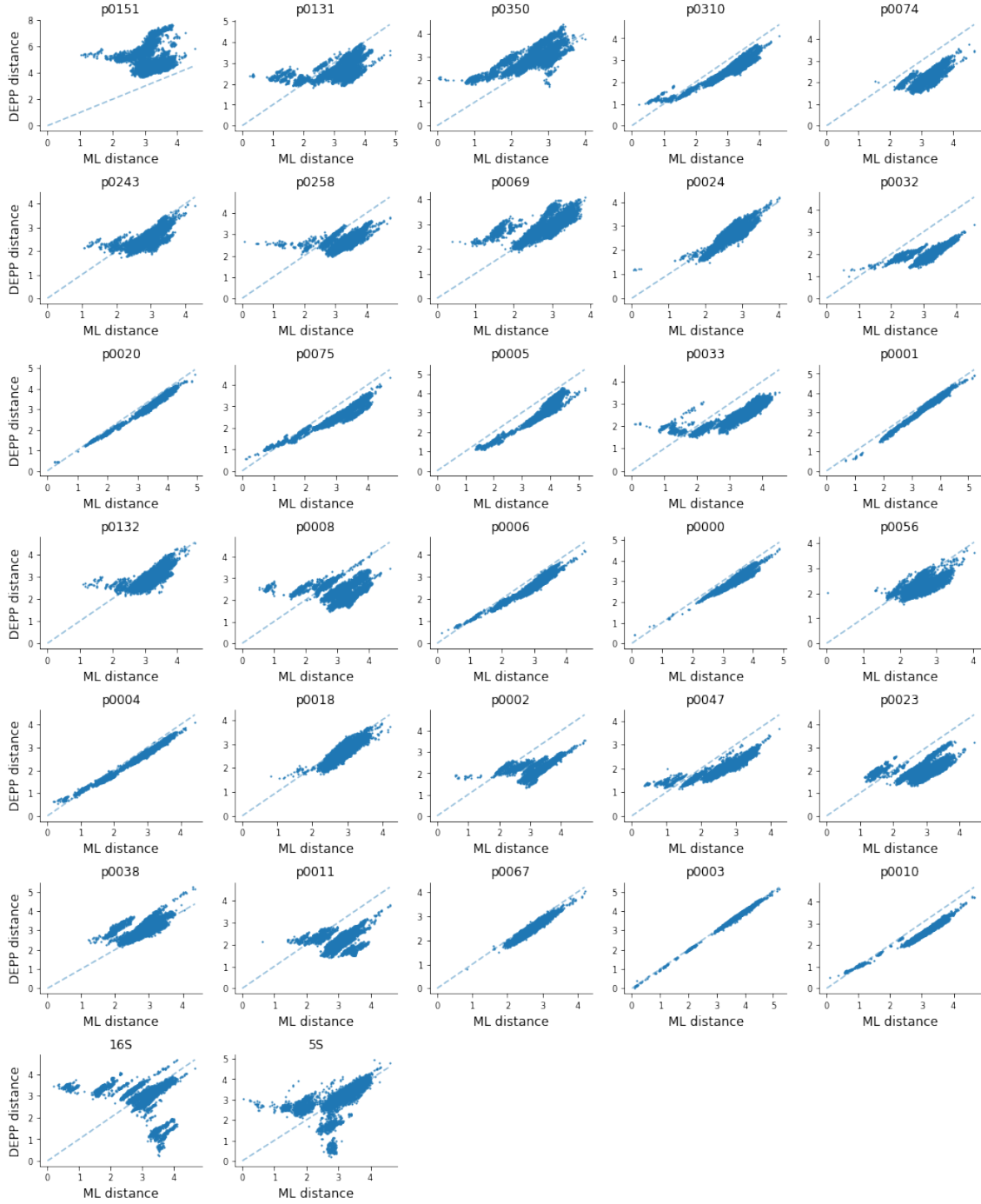


Figure S10: **Examples of distance calculation using DEPP with errors larger than 15** x-axis: the true phylogenetic distance between the query; y-axis: distance calculated by DEPP. Each panel is a query arbitrarily chosen from the experiment corresponding to the gene indicated by the panel's tile. See Table S3 for gene properties.

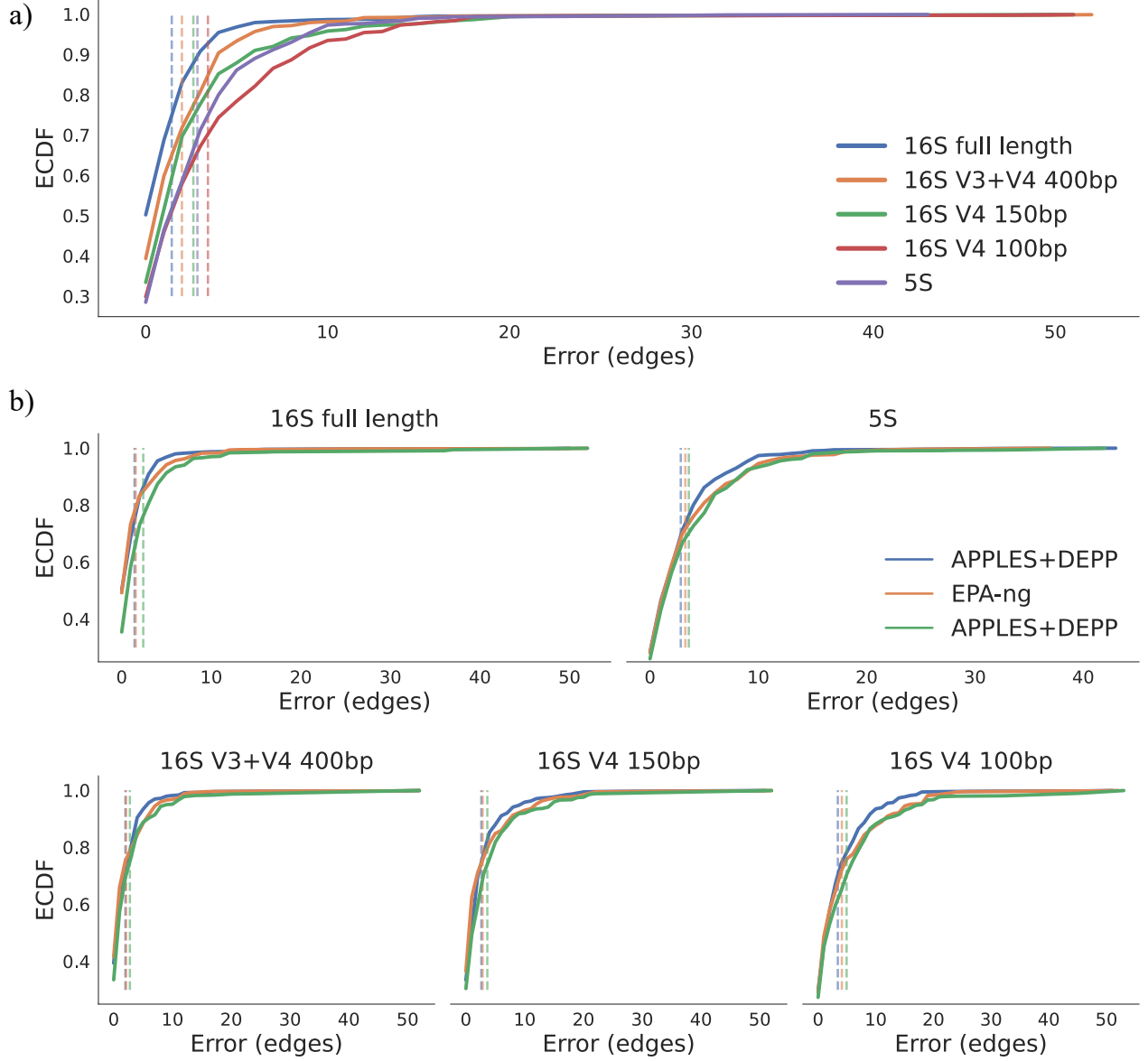


Figure S11: **Complete Empirical CDF on the WoL dataset.** In this figure we show the complete ECDF figure on WoL data; a truncated version is shown Fig. 4b.



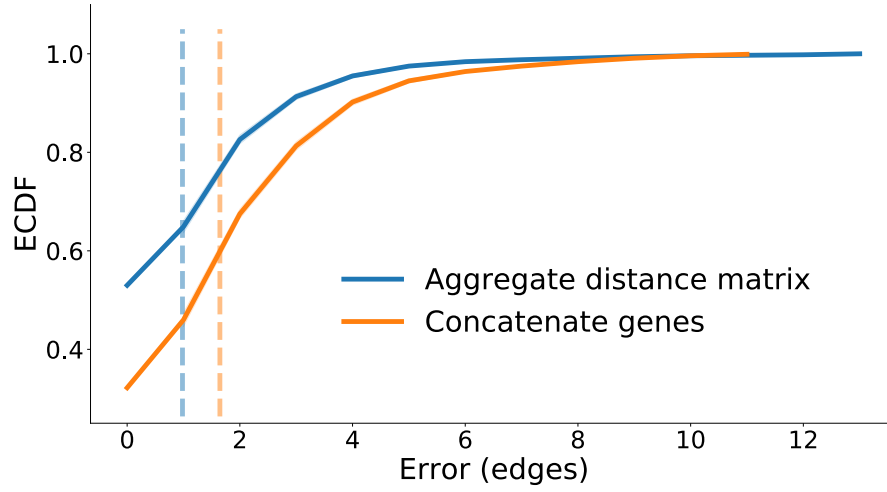


Figure S12: **Results of multiple genes on WoL dataset summarized using two different ways.** We show results of using 50 randomly selected genes in a leave-out experiment. Blue: We train 50 models which gives us 50 distance matrices. We then aggregate the 50 distance matrices by taking the mean distances in the interquartile range. Orange: We concatenate the sequences from 50 genes and train a model on the concatenated sequences. Note that these analyses are using the older version of DEPP (v.0.1.56), and an older version of the dataset with ultrametric species trees used as the backbone (data release, v1.1.0).

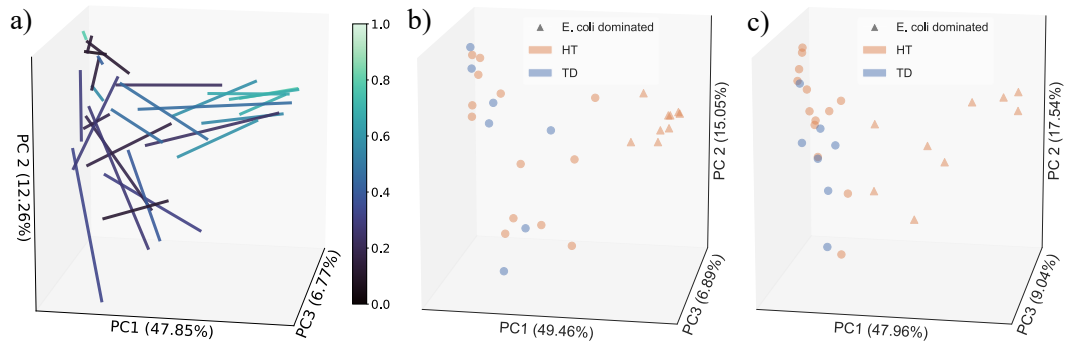


Figure S13: **Visualization of beta diversity of Traveler's Diarrhea (TD) gut microbiomes.** We plot samples using PCoA run on weighted UniFrac distances. a) The PCoA visualization of both MAG and ASV profiles. Profiles from the same sample are linked by an edge, and the color of the edge shows the percentage of sequencing data covered by MAGs. Note that lighter edges, corresponding to samples with a higher MAG coverage, tend to have shorter lengths. b,c) The PCoA visualization of MAG profiles (b) and ASV profiles (c) separately. *E. coli*-dominated samples are more easily delineated using MAGs.

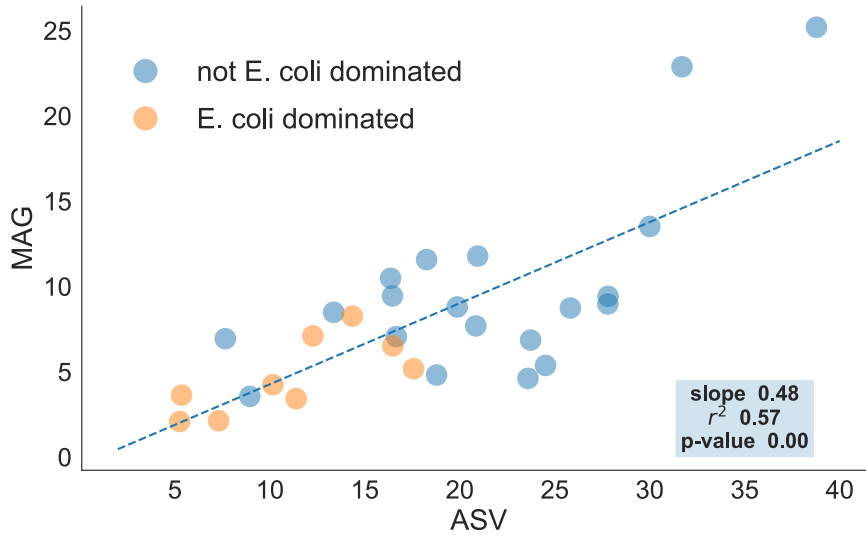


Figure S14: **Correlation of Alpha diversity measured using ASVs and MAGs on Traveler's Diarrhea dataset.** We show the alpha diversity of MAGs versus ASVs, measured using Faith's phylogenetic diversity (PD). Since this measure ignores abundance, to remove the impact of noise, we only consider the most abundant species that cumulatively cover at least 95% of the placed sequences for each sample. The dotted line shown in the figure is fitted using alpha diversity from all 29 samples.

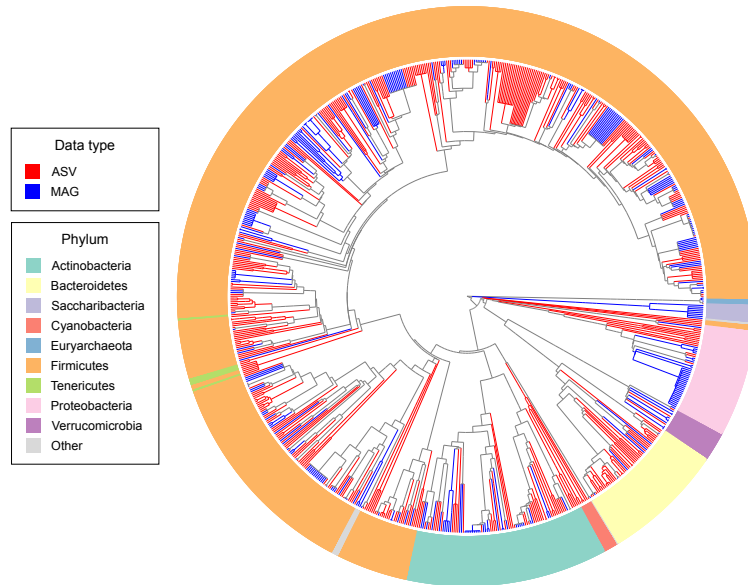


Figure S15: ASV and MAG placement on the WoL phylogenetic tree (no backbone species included)

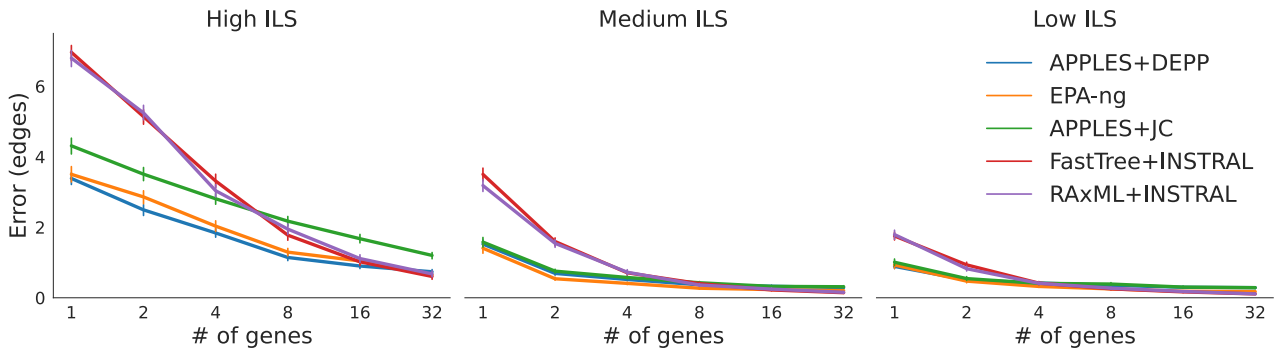


Figure S16: **Mean and standard error of placement error versus the number of genes**  
Input tree to INSTRAL is estimated using FastTree and RAXML

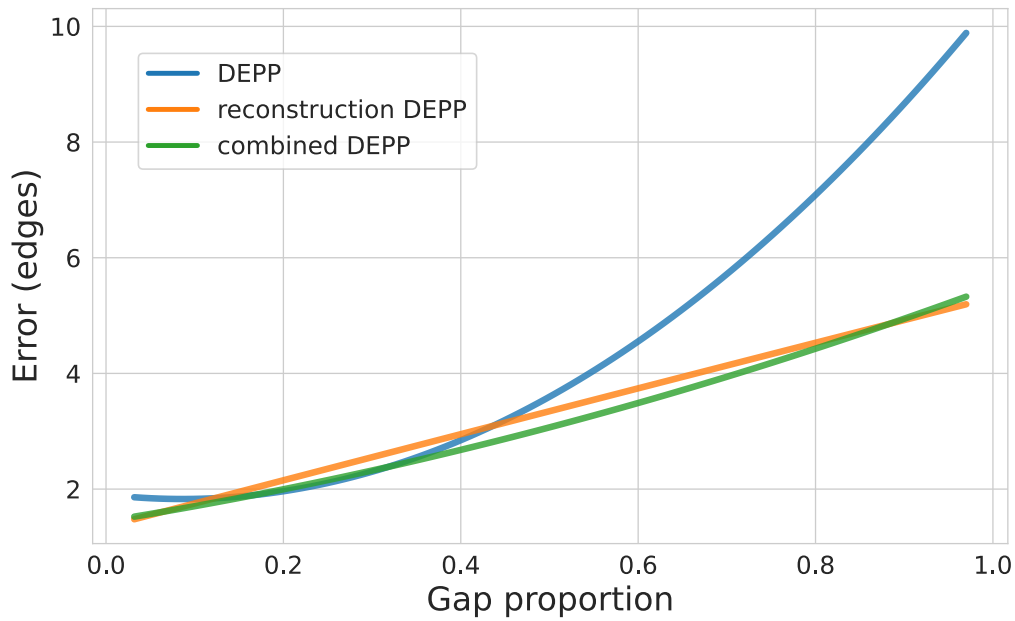


Figure S17: **Impact of reconstruction network** The x-axis is the proportion of gaps in the alignments of the queries; Data is from 30 marker genes in the WoL dataset

## B Supplementary Tables

Table S1: ANOVA statistical tests of the impact of gene tree discordance (RF between true gene trees and the species tree) and lack of gene signal (RF between true gene trees and estimated gene trees), both log transformed for better linearization, on the placement error of DEPP and APPLES+JC.

Method	Variable	Df	Sum Sq	Mean Sq	F value	p-value	Explained %
DEPP	log(1 - gt. err)	1	618.71	618.71	85.00	0	4.8
	log(1- discord)	1	1325.80	1325.80	182.14	0	10.4
	Residuals	1486.0	10816.49	7.27			84.8
JC	log(1 - gt. err)	1	4252.44	4252.44	384.01	0	18.7
	log(1- discord)	1	1973.77	1973.77	178.24	0	8.7
	Residuals	1486.0	16455.38	11.07			72.6
EPA-ng	log(1 - gt. err)	1	1596.27	1596.27	140.09	0	8.0
	log(1- discord)	1	1328.98	1328.98	116.63	0	6.7
	Residuals	1486.0	16932.23	11.39			85.3
INSTRAL	log(1 - gt. err)	1	1541.83	1541.83	99.43	0	5.1
	log(1- discord)	1	5658.22	5658.22	364.92	0	18.7
	Residuals	1486.0	23040.67	15.50			76.2
Using quartet score (for true discordance)							
DEPP	log(1 - gt. err)	1	618.71	618.71	81.40	0	4.9
	log(1- discord)	1	848.12	848.12	111.59	0	6.6
	Residuals	1486.0	11294.16	7.60			88.5

Table S2: Impact of hyperparameters on the error of DEPP. We test on the single-gene high discordance 200-taxon simulated dataset. Bold: the default setup, used for the results reported in the paper.  $k$ : embedding size,  $\delta$ : square root of the tree distance. To change the architecture, we change the number of resblocks. Various weighting schemes are incorporated in (1). Error: mean placement error (edges). Statistical tests of significance compare each condition to the default condition using a two-sided paired Student’s t test ( $p$ -values).

	32	64	128	128	<b>128</b>	128	128	128	256
resblocks	5	5	1	3	<b>5</b>	5	5	5	5
weighting	$\frac{1}{\delta^2}$	$\frac{1}{\delta^2}$	$\frac{1}{\delta^2}$	$\frac{1}{\delta^2}$	$\frac{1}{\delta^2}$	1	$\frac{1}{\delta}$	$\frac{1}{\delta^4}$	$\frac{1}{\delta^2}$
error	3.836	3.380	3.734	3.534	<b>3.442</b>	3.374	3.516	3.520	3.302
$p$ -values	0.002	0.544	0.023	0.359	N/A	0.472	0.454	0.442	0.161

*Observations:*

- Increasing embedding size ( $k$ ) tend to reduce the error.
- More residual blocks lead to better performance.
- DEPP is robust to weighting schema. Different weighting schema do not have statistically significant difference in error.

Table S3: Properties of selected genes from the WoL dataset Zhu et al., 2019. Length indicates the length of the gene’s protein alignment. Occupancy shows the number of species that has the each gene in their genome. QD: Quartet distance between the published gene tree and the species tree. Finally, group shows the assignment of genes to the three groups of low, medium, and high discordance based on the QD.

marker	UniProt ID	gene	length	occupancy	QD	group
p0010	C9RQN5	<i>rpoB</i>	1429	9911	0.09324	low
p0003	A0A0T5XBS3	<i>rpoC</i>	1652	10249	0.11948	low
p0067	P62663	<i>rpsC</i>	239	10295	0.16015	low
p0011	Q1IJG7	<i>alaS</i>	1098	10253	0.16170	low
p0038	K9TTL6	<i>rplB</i>	287	10208	0.16852	low
p0023	D3R080	<i>infB</i>	1155	10197	0.17352	low
p0002	B1AI94	<i>ftsH</i>	721	10243	0.22639	low
p0047	D2PUL3	<i>Kfla (0407)</i>	1113	10201	0.20781	low
p0018	B5I9V0	<i>pgk</i>	409	9969	0.23642	low
p0004	P19486	<i>tuf</i>	424	10054	0.25889	low
p0056	B9CN99	<i>rimI</i>	816	10047	0.28959	mid
p0000	B3PLT3	<i>oppF-valS</i>	1631	9985	0.29736	mid
p0006	Q7US70	<i>cysS</i>	537	9977	0.31252	mid
p0008	D5WTU4	<i>serS</i>	428	9944	0.32622	mid
p0132	A0A0P0KSS3	<i>metAP1b</i>	333	9745	0.33467	mid
p0001	A0A173SLF7	<i>fusA1</i>	880	10237	0.34126	mid
p0033	D3LW72	<i>HMPREF0889 (1213)</i>	550	10315	0.35079	mid
p0005	E1GXM3	<i>ileS</i>	1220	10187	0.37819	mid
p0075	E8X3N9	<i>AciX9 (1167)</i>	848	10135	0.38996	mid
p0020	D8PAC1	<i>thrS</i>	651	10063	0.39864	mid
p0032	Q9X3X7	<i>topA</i>	1212	10101	0.41792	high
p0024	A5F9G1	<i>epd</i>	341	9337	0.42137	high
p0069	B1GZ24	<i>fold</i>	296	9389	0.43362	high
p0258	E1WYY2	<i>tyrS</i>	423	9238	0.46740	high
p0243	D5E5H2	<i>hisRS</i>	478	9769	0.48789	high
p0074	A0A173ZQ45	<i>rhIE</i>	747	8977	0.50476	high
p0310	P75510	<i>trpS</i>	346	9368	0.53021	high
p0350	B8J115	<i>metN</i>	348	9536	0.56286	high
p0131	A0A1C6GRF3	<i>potA1</i>	525	9110	0.58579	high
p0151	I5AQ83	<i>EubceDRAFT1 (0090)</i>	1320	10070	0.59077	high

Table S4: Number of species left after filtering out sequences that is shorten than roughly half the average length.

Dataset	Unfiltered	full length	V3+V4	V4 150bp	V4 100bp
# of species	7813	7797	7730	7731	7740
Threshold	N/A	800	200	75	50

Table S5: Running time and peak memory usage for three experiments. All of the methods are run on a single CPU. First three columns are DEPP variants: DEPP; DEPP with missing data reconstruction; DEPP with backbone embeddings saved. On the WoL dataset, DEPP is the fast method without the reconstruction step. Adding reconstruction network halves the speed of DEPP and makes it slower than EPA-ng. However, if we saved the backbone embeddings before teasing, DEPP can be 2-times faster than EPA-ng and 5-times faster than APPLES+JC.

		DEPP	DEPP (recon)	DEPP (B.S.)	EPA-ng	JC
ILS simulated data	time	6m36s	-	6m11s	1m17s	2m56s
	memory	0.42G	-	0.30G	0.22G	0.05G
HGT simulated data	time	18m16s	-	13m08s	37m01s	108m04s
	memory	1.52G	-	1.14G	10.59G	0.12G
WoL 30 marker genes	time	31m55s	79m36s	16m48s	43m25s	97m09s
	memory	3.23G	4.31G	2.38G	29.37G	0.19G

1026

## C Background on CNNs

**Convolutional layers.** Denote an input feature map of the intermediate layer of a model as  $X \in \mathbb{R}^{L \times c_{in}}$ , where  $L$  is the length of the feature map (here, sequence length), and  $c_{in}$  is the number of input channels (here, 4 for the initial layer). Also given is a filter  $F \in \mathbb{R}^{k \times c_{in} \times c_{out}}$  where  $k$  is the kernel size and  $c_{out}$  is the number of output channels. The convolutional operation is defined as:

$$Y(l, t) = \sum_{i=0}^k \sum_{c=0}^{c_{in}} X(l+i, c) \times F(i, c, t)$$

1027

1028

1029

One-dimensional convolutional layer use sliding windows to process the input sequence. Each site in the output is the weighted sum of the fragment in the corresponding windows plus the bias.

1030

1031

1032

1033

1034

In a neural network, convolutional layers are usually used as feature extractors. Multiple convolutional layers are able to detect high-level abstraction from the input. One-dimensional convolutional layer use sliding windows to process the input sequence. Each site in the output is the weighted sum of the fragment in the corresponding windows plus the bias. The weights are determined by a learnable kernel.

1035

1036

1037

1038

**Fully Connected Layer.** In a neural network, fully connected layers are usually used as the last few layers to aggregate information and get the final output. In the fully connected layer, each activation in the output has connections with all the input. Specifically, each output dimension is a weighted sum of all the input, and the weights are optimized during training.

1039

1040

1041

1042

**Nonlinear layer.** Convolutional layers and fully connected layers consist of only linear operations. Introducing nonlinear layers can give the model the ability to capture nonlinear relations. While many nonlinear kernels exist, in this work, we use Continuously Differentiable Exponential Linear Units (CELU) as the nonlinear layer applied at the end of each layer.



<sup>1043</sup> CELU has the following form:

$$\text{CELU}(x, \alpha) = \begin{cases} x, & x \leq 0, \\ \alpha(e^{\frac{x}{\alpha}} - 1), & \text{otherwise} \end{cases} \quad (\text{S1})$$

## 1044 **D Exact commands and versions**

### 1045 **Software version**

- 1046 • APPLES 2.0.0
- 1047 • DEPP 0.2.2
- 1048 • EPA-ng 0.3.8
- 1049 • INSTRAL 5.13.5
- 1050 • RAxML-ng 1.0.1
- 1051 • RAxML 8.2.12
- 1052 • UPP 4.3.10
- 1053 • Qiime2 2020.11.1
- 1054 • Prodigal v2.6.3
- 1055 • PhyloPhlAn commit 2c0e61a
- 1056 • Simphy 1.0
- 1057 • Indelible V1.03

### 1058 **Branch length reestimation**

- 1059 • `raxml-ng --evaluate --msa $sequence_file --tree $tree --model JC --threads 2 --blopt`  
 1060 `nr_safe`
- 1061 • `raxmlHPC-PTHREADS -s $sequence_file -w $outdir -n run -p 12345 -T 32 -m GTRCAT`  
 1062 `-g $tree`

### 1063 **Detecting Genes**

- 1064 • Predict open reading frames (ORFs) and translate DNA sequences into amino acid sequences
- 1065   `— prodigal -p $mode -f gff -g 11 -i $g.fna -o $g.gff -a $g.faa -d $g.ffn`  
 1066    \$g: genome ID, \$mode: "single" for single genomes (e.g., WoL) and "meta" for metagenomes  
 1067    (e.g., TD)
- 1068 • Identify the 400 marker genes from ORFs
- 1069   `— phylophlan.py`  
 1070    `--c_dat $tmpdir --c_in $indir`

```
1071 --c_out $outdir --nproc $cpus -u $rundir
```

## 1072 Gene alignment

1073 Genes were aligned using UPP using the following commands for aligning the backbone se-  
 1074 quences(first command), and aligning the novel queries to the existing backbone alignments  
 1075 (second command).

```
1076 • run_upp.py -s $input_seq -B 2000 -M -1 -T 0.33 -A 200
```

```
1077 • run_upp.py -s $query_seq -a $backbone_seq -t $backbone_tree -A 100 -d $outdir
```

## 1078 Placement

```
1079 • APPLES+DEPP
```

```
1080 – Training
```

```
1081 * Simulated dataset
```

```
1082 train.depp.py
```

```
1083 backbone_tree_file=$backbone_tree
```

```
1084 backbone_seq_file=$backbone_seq
```

```
1085 patience=5 lr=1e-4 embedding_size=128
```

```
1086 * Marker genes in WoL dataset
```

```
1087 train.depp.py
```

```
1088 backbone_tree_file=$backbone_tree
```

```
1089 backbone_seq_file=$backbone_seq
```

```
1090 patience=5 lr=1e-4 embedding_size=512
```

```
1091 * 5S/16S in WoL dataset
```

```
1092 train.depp.py
```

```
1093 backbone_tree_file=$backbone_tree
```

```
1094 backbone_seq_file=$backbone_seq
```

```
1095 patience=5 lr=1e-4 embedding_size=512 replicate_seq=True
```

```
1096 * Note: for version 1.0.0 of this reference DB, we used DEPP 0.1.13, for version 1.1.0  

  1097 (used everywhere unless otherwise specified), we use DEPP 0.1.54 for version 1.2.0, we
```

```

1098         use DEPP 0.2.2.
1099     - Query time
1100     * Calculating distance matrix
1101         · depp_distance.py
1102         query_seq_file=$query_seq
1103         backbone_seq_file=$backbone_seq
1104         model_path=$model_path
1105         outdir=$out_dir
1106     * Placement using APPLES
1107         · run_apples.py -d $distance_file -t $backbone_tree -o -f 0 -b 5
1108     • APPLES+JC
1109     run_apples.py -s $backbone_tree -q $query_seq -t $backbone_tree -f 0 -b 5
1110     • EPA-ng
1111     - raxml-ng
1112         --evaluate
1113         --msa $backbone_seq
1114         --tree $backbone_tree
1115         --prefix info
1116         --model GTR+G+F
1117         --threads 2 --blopt nr_safe
1118     - epa-ng
1119         --ref-msa $backbone_seq
1120         --tree $backbone_tree
1121         --query $query_seq
1122         --model $GTR_info
1123     • INSTRAL
1124     - java -Djava.library.path=./ -jar __instral.jar__
1125         -i $gene_trees
1126         -f $backbone_tree

```

1127     --placement \$query

1128     -o \$placement\_tree -C

## 1129   **Weighted UniFrac**

1130   • qiime diversity beta-phylogenetic

1131     --p-metric weighted\_unifrac

1132     --i-table \$feature\_table

1133     --i-phylogeny \$placement\_tree

1134     --o-distance-matrix \$unifrac\_distance\_matrix

## 1135   **PERMANOVA**

1136   • qiime diversity beta-group-significance

1137     --i-distance-matrix \$unifrac\_distance\_matrix

1138     --m-metadata-file \$metadata

1139     --o-visualization \$permanova\_output

1140     --m-metadata-column group

1141     --p-permutations 999999

## 1142   **PCoA**

1143   • qiime diversity pcoa

1144     --i-distance-matrix \$unifrac\_distance\_matrix --o-pcoa \$pcoa\_matrix

1145 **D.1 HGT simulation**

```
1146 • simphy -rs 10 -rl f:500 -rg 1 -sb f:0.0000005 -sd f:0.000000416666667
1147 -st f:100000000 -sl f:10000 -si f:1 -sp f:500000 -su f:4e-08 -hh f:1
1148 -hs ln:1.5,1 -hl ln:1.3692114,0.6931472 -hg ln:1.5,1 -cs 14907 -gt n:-18,0.4
1149 -lt ln:gt,0.75 -lk 1 -lb f:0 -ld f:0 -v 3
1150 -o model.10000.100000000.0.0000005.norm-lognormal -ot 0 -op 1 -od 1 > log.txt
```