BIOINFORMATICS ORIGINAL PAPER

Vol. 30 no. 24 2014, pages 3548–3555
doi:10.1093/bioinformatics/btu721

Sequence analysis

Advance Access publication October 29, 2014

TIPP: taxonomic identification and phylogenetic profiling

Nam-phuong Nguyen[1], Siavash Mirarab[1], Bo Liu[2], Mihai Pop[2] and Tandy Warnow[1,3,4,*]

[1]Department of Computer Science, The University of Texas at Austin, Austin, TX USA, [2]Center for Bioinformatics and Computational Biology, University of Maryland at College Park, College Park, MD USA, [3]Department of Bioengineering, The University of Illinois at Urbana-Champaign, Urbana, IL USA and [4]Department of Computer Science, The University of Illinois at Urbana-Champaign, Urbana, IL USA

Associate Editor: Gunnar Ratsch

ABSTRACT

Motivation: Abundance profiling (also called 'phylogenetic profiling') is a crucial step in understanding the diversity of a metagenomic sample, and one of the basic techniques used for this is taxonomic identification of the metagenomic reads.

Results: We present taxon identification and phylogenetic profiling (TIPP), a new marker-based taxon identification and abundance profiling method. TIPP combines SAT\'e-enabled phylogenetic placement a phylogenetic placement method, with statistical techniques to control the classification precision and recall, and results in improved abundance profiles. TIPP is highly accurate even in the presence of high indel errors and novel genomes, and matches or improves on previous approaches, including NBC, mOTU, PhymmBL, MetaPhyler and MetaPhlAn.

Availability and implementation: Software and supplementary materials are available at http://www.cs.utexas.edu/users/phylo/software/sepp/tipp-submission/.

Contact: warnow@illinois.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Received on April 5, 2014; revised on October 12, 2014; accepted on October 24, 2014

1 INTRODUCTION

Metagenomic studies of microbial communities commonly generate millions to hundreds of millions of sequencing reads. The assignment of accurate taxonomic labels to these sequences is a critical component in many analyses, but is complicated by the fact that the majority of the organisms found in environmental or host-associated communities cannot be easily cultured in a laboratory, and of the organisms that can be cultured, an even smaller number have been sequenced, even partially. Thus, these commonly encountered organisms are largely absent from existing databases of known genomes and genes. Providing taxonomic labels to metagenomic sequences requires extrapolating the knowledge contained in sequence databases to previously unseen deoxyribonucleic acid (DNA) strings. Simple similarity-based approaches (e.g. picking the best database hit as the best 'guess' at the taxonomic label) have been shown to be insufficiently accurate (Koski and Golding, 2001), leading to the development of new and more sophisticated methods.

Recently developed methods improving on the simple similarity-based approaches include (a) composition-based approaches that rely on various machine learning techniques (Support Vector Machines in PhyloPythia and PhyloPythiaS (McHardy *et al*., 2007; Patil *et al*., 2011), Interpolated Markov Models in Phymm (Brady and Salzberg, 2011), Bayesian models in NBC [Rosen *et al*., 2011), or neural networks (Abe *et al*., 2006)] to classify sequences based on their DNA composition (usually based on the frequency of short k-mers), (b) more sophisticated analyses of similarity search results [e.g. using lowest common ancestor aggregation in Megan (Huson *et al*., 2007), or classifiers built from similarity search results as done in MetaPhyler (Liu *et al*., 2010, 2011), MetaPhlAn (Segata *et al*., 2012) and mOTU (Sunagawa *et al*., 2013) or protein profiles in Carma (Gerlach and Stoye, 2011)], and (c) combinations of multiple approaches (e.g., composition and similarity based approaches in PhymmBL (Brady and Salzberg, 2009)). Some of these approaches (e.g. most of the composition-based approaches) can be applied to any DNA sequence. However, others are specific to a reference collection of carefully selected genes (e.g. MetaPhyler, MetaPhlAn and mOTU use a collection of universal or clade-specific marker genes), and so are called 'marker-based' methods.

Abundance profiling, also called 'phylogenetic profiling', seeks to estimate the relative abundance of the species (or genera, or families, etc.) within a sequence dataset. While many methods produce these estimates by characterizing most (or all) of the sequences in the dataset, marker-based methods produce these estimates by characterizing only those sequences that match the marker genes they rely on. The marker genes used by some of these methods (e.g. MetaPhyler and mOTU) are supposed to be single copy and universal; thus, the estimates produced using such markers can be used for profiling without needing to correct for copy number or missing data.

We present taxon identification and phylogenetic profiling (TIPP), a new marker-based method for taxon identification and abundance profiling. We explore TIPP for abundance profiling using MetaPhyler's collection of 30 marker genes, but TIPP can be used with any single-copy gene for which a dense enough sample of full-length sequences is available. TIPP is implemented in Python and can be run on Unix and Mac.

2 APPROACH

2.1 Overview

TIPP has three phases: pre-processing (Phase 1), taxon identification for reads mapped to its marker genes (Phase 2) and abundance profiling based on these taxon identifications (Phase 3).

*To whom correspondence should be addressed.

Phase 1 involves a sequence of preprocessing steps, based on its set S of reference sequences for its set of marker genes, and its input set Q of uncharacterized reads (the 'query sequences'). TIPP uses SATé (Liu *et al.*, 2009, 2012) to compute an alignment A and tree T for full-length sequences for each of its marker genes, using its reference set S; these are called the 'backbone alignment' and 'backbone tree'. Since the NCBI taxonomy is not fully resolved (i.e., bifurcating), TIPP uses the backbone alignment and RAxML (Stamatakis, 2006) to refine the NCBI taxonomy (restricted to the sequences in the backbone alignment) into a fully resolved tree, T^*. Thus, each marker gene has its own refinement of the taxonomy that is used in subsequent analyses. TIPP then uses BLAST (Altschul *et al.*, 1990) to determine which reads in Q map to its marker genes. At the end of this preprocessing, the set Q is partitioned into sets, with one set for each of its marker genes, and a final set for the reads that are not mapped to any marker gene

Phase 2, where reads that are mapped to marker genes are taxonomically characterized, is the most involved, and is described below (and also in Fig. 1).

Phase 3, which estimates the abundance profile, is quite simple. We pool all the reads that mapped to any marker gene and have been characterized at any taxonomic level into a single set. We compute the abundance profile as the relative distribution of the clades present in the pooled set. As the pooled reads are only required to be characterized at the phylum level, the abundance profiles can include a category of unclassified sequences. Sequences can be labeled as unclassified due to two reasons: there may not be sufficient support to classify a sequence at a given taxonomic level, and the taxonomy may not have a classification for a clade at a particular taxonomic level. For example, a species that does not have a family label would be unclassified at the family level, even if it was classified at a lower level; see Section S7 for an example.

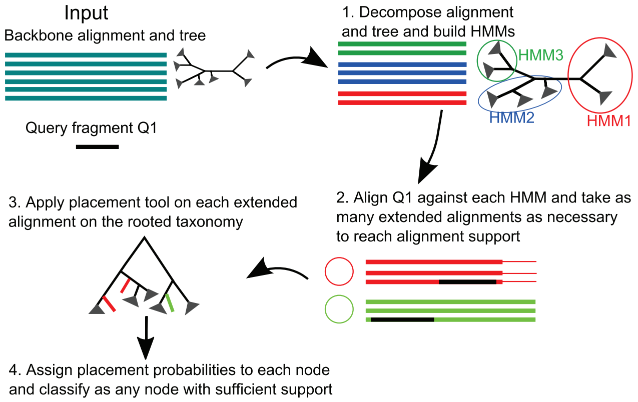
Input
Backbone alignment and tree
Query fragment Q1
1. Decompose alignment and tree and build HMMs
HMM3
HMM2
HMM1
2. Align Q1 against each HMM and take as many extended alignments as necessary to reach alignment support
3. Apply placement tool on each extended alignment on the rooted taxonomy
4. Assign placement probabilities to each node and classify as any node with sufficient support

Fig. 1. TIPP's algorithm to taxonomically characterize the query sequence q that is already mapped to a marker gene. The backbone alignment and tree are for the marker gene, and an HMM Family is computed for the backbone alignment. The HMM Family is then used to compute one or more 'extended alignments' including q. The phylogenetic placement method pplacer produces a probability distribution for the placements of q into the reference taxonomy. Each node in the taxonomy is labeled with the sum of the probabilities of any placements at or below the edge above the node. TIPP classifies the query sequence at any node with sufficient statistical support

2.2 Phase 2: taxonomic characterizations of reads

We describe how TIPP performs its taxonomic characterization of a single read mapped to a single marker gene; see Figure 1. The input to TIPP includes the maximum alignment subset size (m) and the statistical support thresholds ($s_{\rm a}$ and $s_{\rm p}$) for alignment and placement support, respectively, which can be set by the user. We set default values as follows: $m = 100$ and $s_{\rm a} = s_{\rm p} = 95\%$.

Step 1: Decomposition. TIPP decomposes the set of leaves in the backbone tree T into subsets using the decomposition technique in SEPP. TIPP finds a centroid edge in T (one that separates the leaf set into two sets of approximately equal size), breaks the tree accordingly into two subtrees, and recurses on each subtree that contains more than m leaves. This produces a partition of the leaves of T into alignment subsets $S_1, S_2, \ldots, S_k$, each of size at most m.

Step 2: Compute Extended Alignment(s). We define the alignment A_i by restricting the alignment A to S_i, for every $1 \le i \le k$; these are the 'subset alignments'. TIPP uses the HMMER software suite (Eddy, 1998) to compute an HMM H_i on each A_i. Thus, TIPP represents the backbone alignment with a set of HMMs, a technique we call an 'HMM Family'. HMMER produces bit scores (discussed below), which are measures of the fit between each H_i and each query sequence $q \in Q$. Then, for the query sequence q, one or more subset alignments are selected so that the total statistical support for the alignments is at least $s_{\rm a}$ (see below for how TIPP computes statistical support). TIPP uses HMMER to align q to each of the selected subset alignments, and thus, produces extensions of each subset alignment A_i that include q (called an 'extended alignment'). Thus, each query sequence q gives rise to at least one and potentially many extended alignments of the reference dataset, each with $|S| + 1$ sequences.

Step 3: Placement. For each query sequence q and each extended alignment containing q, we use pplacer (Matsen *et al.*, 2010), a maximum likelihood phylogenetic placement method, to insert q into T^*. The output of pplacer provides multiple placements and their likelihood weight ratios (the maximum likelihood values for all placement locations, normalized to sum to 1) for each extended alignment. We combine all placements resulting from the extended alignments into a single collection of placements, re-normalize their likelihood weight ratios to sum up to 1 (weighting each placement by the corresponding alignment support), and treat the normalized likelihood weight ratios as probabilities; see Section S8 for an example.

Step 4: Classification. For each query sequence, we assign statistical support to each node in the taxonomy by adding the probabilities of all placements at or below the edge above the node; thus, the statistical support monotonically increases as one traverses the tree from any leaf to the root, and this allows us to classify the query sequence at all taxonomic levels for which it has support of at least $s_{\rm p}$. The query sequence is left unclassified at levels where the support of $s_{\rm p}$ is not reached. A unique taxon identification is produced whenever $s_{\rm p} > 0.5$; otherwise, TIPP outputs all identifications that meet the support threshold of $s_{\rm p}$, along with their support values. Because our default setting has $s_{\rm p} = 0.95$, TIPP produces a unique taxon identification for each query sequence.

2.3 Alignment support calculation

To take alignment uncertainty into account, we take a large enough number of extended alignments to reach the alignment support threshold s_a. To determine the number of extended alignments we need, we use HMMER's output to define the probability that a given query sequence is generated by a given HMM from the set of HMMs computed for the different subset alignments. These calculations are based on the assumptions that (1) the subsets are disjoint, so that at most one HMM generates the query sequence and (2) the query sequence is generated by some HMM.

For a given HMMER model H and query sequence q, HMMER calculates a bit-score (which we denote BS(H)), defined by:

$$\mathrm{BS}(H) = \log_2 \frac{P(q|H)}{P(q|R)} \tag{1}$$

where $P(q|H)$ is the probability of model H generating query sequence q, and $P(q|R)$ is the probability of a random model R generating query sequence q. Assuming that the query sequence q is generated by exactly one of the HMMs (H_1 to H_n, each corresponding to a different subset alignment), the probability that H_i generated q is:

$$P(H_i|q) = \frac{P(q|H_i)P(H_i)}{\sum_{j=1}^{n} P(q|H_j)P(H_j)}. \tag{2}$$

Since our subsets all have roughly equal size, we make the simplifying assumption that the *a priori* probabilities of any two HMMs generating any given query sequence are equal. We can rewrite Equation (2) as:

$$P(H_i|q) = \frac{1}{\sum_{j=1}^{n} \frac{P(q|H_j)}{P(q|H_i)}} \tag{3}$$

By Equation (1),

$$\mathrm{BS}(H_j) - \mathrm{BS}(H_i) = \log_2 \frac{P(q|H_j)}{P(q|R)} - \log_2 \frac{P(q|H_i)}{P(q|R)} \tag{4}$$

$$= \log_2 \frac{P(q|H_j)}{P(q|H_i)} \tag{5}$$

Hence, the probability of H_i using bit-scores is given by:

$$P(H_i|q) = \frac{1}{\sum_{j=1}^{n} 2^{\mathrm{BS}(H_j) - \mathrm{BS}(H_i)}} \tag{6}$$

Thus, assuming that the bit-scores are sorted such that $\mathrm{BS}(H_i) \geq \mathrm{BS}(H_{i+1})$ $(i = 1, 2, \ldots, n-1)$, to reach a specified threshold s_a, we find the smallest m such that $\sum_{k=1}^{m} P(H_k|q) \geq s_a$.

3 EXPERIMENTAL DESIGN

3.1 Overview

We evaluated TIPP in comparison to other phylogenetic profiling methods under three different conditions. Experiment 1 compared performance under easy conditions, where the genomes are known and the reads (i.e. query sequences) have low rates of sequencing errors. Experiment 2 examined datasets with high rates of insertion and deletion errors (collectively known as 'indels'). Experiment 3 examined performance where the query sequences come from 'novel' genomes (defined below).

3.1.1 Methods studied We compared TIPP, MetaPhyler, MetaPhlAn, PhymmBL, mOTU and NBC. Three of these methods (TIPP, mOTU and Metaphyler) are marker-based methods, and use universal housekeeping genes that are unlikely to undergo duplication or horizontal gene transfer. Both TIPP and MetaPhyler use the same collection of 30 marker genes, and mOTU uses a reduced set of 10 marker genes selected from Mende *et al.* (2013). MetaPhlAn, conversely, selects markers that uniquely identify specific taxonomic groups. Assigning query sequences to genes (i.e. binning) is performed internally by each of the marker-based methods; see Section S4 for the BLAST settings used within TIPP for binning.

Phred quality scores (Ewing and Green, 1998) are needed for each read, in order for mOTU to run. Since the datasets used in our study do not have the quality scores, we assigned Phred quality scores of 33 to all the bases of all the reads (99.95% probability that the base is correct). MetaPhyler allows the user to input a confidence threshold, and query sequences are only classified at a given level if this confidence threshold is met. We use the confidence level of 90% suggested by the authors of MetaPhyler. For PhymmBL, we classify a query sequence at the most specific classification yielding a confidence score of 95% or higher; however, PhymmBL does not give confidence scores at the species level, and thus, cannot be used to perform abundance profiling at the species level. Finally, NBC gives a confidence score of the query sequence matching a taxon. We accept the classification if the confidence score is above the species threshold formula given by the NBC authors (which is a function of read-length; see Section S4); thus, a query sequence will either be classified at the species level or be completely unclassified. See Section S4 for version numbers and commands used.

3.1.2 Reference marker datasets TIPP uses the reference sequence dataset obtained from Liu *et al.* (2010, 2011), consisting of 30 phylogenetic marker genes that span the Bacteria and Archaea domains. The marker genes selected were believed to be single copy genes, universally present across the Bacteria domain, and resistant to horizontal gene transfer. Only species whose genomes have been sequenced were present in the reference dataset. The number of sequenced representatives for each marker gene ranges from 65 to 1555 sequences, with an average of 1312 sequences per marker gene. See Section S2 for the list of marker genes and the empirical statistics of the reference alignments on these datasets.

3.1.3 Training MetaPhyler, MetaPhlAn and mOTU come pretrained, but their reference datasets depend on which version is used. NBC and PhymmBL require the user to manually train their software, and their reference datasets are dependent on when the genomes were downloaded for training. The versions of MetaPhyler, MetaPhlAn and mOTU used in our article are based on reference datasets downloaded from NCBI in 2010, 2011 and 2012, respectively. TIPP is trained on the same reference dataset as MetaPhyler, and thus is trained on sequences downloaded from NCBI in 2010. NBC and PhymmBL were trained on the same set of genomes downloaded from NCBI on May 2013. Thus, NBC and PhymmBL are based on training datasets from 2013, mOTU is based on a training dataset from 2012, MetaPhlAn is based on a training dataset from 2011, and MetaPhyler and TIPP are based on a training dataset from 2010. These differences favour methods (such as NBC and PhymmBL) that are trained on more recent datasets than those based on older datasets (such as MetaPhyler and TIPP). See Section S6 for additional information on how TIPP was trained.

3.1.4 Simulated abundance profiling datasets The datasets we used have different properties, including the average read length (short versus long), complexity of the profiles (uniform versus non-uniform), the rate of sequencing errors (low versus high), and whether the datasets contained 'novel' or 'known' genomes (defined shortly). The complexity of the profile is labeled as low complexity (LC; staggered distribution of species), high complexity (HC; uniform distribution of species) and medium complexity (MC; distribution of species in between LC and HC). A dataset is labeled as 'novel' if none of the marker-based methods have been trained on any of the genomes in that dataset (see below for more detail). A dataset is labeled as 'known' if it contains any genome that was previously used to train at least one of the marker-based methods. We provide a brief overview of the datasets, separated out into experimental conditions (Table 1); a more in-depth description of the individual datasets (including their Shannon Entropy) can be found in Section S3.

3.1.5 Experimental condition 1: easy datasets The first group of datasets contained query sequences from known genomes (all the genomes were present in the training sets of at least one of the marker-based methods) and had low rates of sequencing error (insertions, deletions and substitutions).

The easy datasets are separated into two different conditions: long read datasets (average length of 200 to 1000 nucleotides) and short read datasets (average length of 100 nucleotides or shorter). The long read datasets include the FACS HC simulated dataset (Stranneheim *et al*., 2010), the FAMeS LC, MC and HC datasets (Mavromatis *et al*., 2007), and the WebCarma HC-simulated dataset (Gerlach and Stoye, 2011). Both the WebCarma and FACS dataset contained simulated 454 reads and had an average indel rate of 3% per base. The FAMeS dataset contained reads taken from Sanger sequencing projects, and thus, is expected to have very low rates of sequencing error [typically <1% error per base (Shendure and Ji, 2008)]. The original FACS HC dataset contained viral, bacterial, and human sequences, and we removed the viral and human sequences so that profiles were estimated only on bacterial reads. The short read datasets include the MetaPhlAn HC and LC-simulated datasets (Segata *et al*., 2012), the TIPP FACS HC Illumina-simulated dataset, and the TIPP WebCarma HC Illumina-simulated dataset; the last two datasets were generated using MetaSim (Richter *et al*., 2008) to simulate Illumina reads from the metagenomes used in the original FACS HC and WebCarma HC datasets. The short read datasets contain simulated Illumina reads with at most 0.2% substitutions per base.

3.1.6 Experimental condition 2: datasets with high rates of indel errors The second group of datasets was modified versions of the easy datasets with additional indels to make analyses more difficult. The easy datasets contained reads with less than 0.2% substitution errors per base or less than 3% indel errors per base. We inserted additional indel sequencing errors to the query sequences in the easy datasets, using indel rates typical of PACBio reads (12% insertion and 2% deletion error rate per base (Carneiro *et al*., 2012)). Details can be found in Section S3.6.

Table 1. Properties of the simulated datasets

Dataset	Experiment	# Genomes	Complexity	# Reads	Length
MetaPhlAn HC	1&2	100	High	1000000	88 (s)
MetaPhlAn LC	1&2	25	Low	240000	88 (s)
FAMeS HC	1&2	113	High	116771	949 (l)
FAMeS MC	1&2	113	Medium	114457	969 (l)
FAMeS LC	1&2	113	Low	97495	951 (l)
FACS HC-454	1&2	19	High	26984	268 (l)
TIPP FACS HC-Illum	1&2	19	High	300000	100 (s)
WebCarma-454	1&2	25	High	25000	265 (l)
TIPP WebCarma-Illum	1&2	25	High	300000	100 (s)
TIPP HC novel-Illum	3	100	High	1000000	100 (s)
TIPP LC novel-Illum	3	100	Low	1000000	100 (s)
TIPP HC novel-454	3	100	High	1000000	269 (l)
TIPP LC novel-454	3	100	Low	1000000	269 (l)

Eleven basic datasets were used in this study, but each dataset has two versions – a low error model (with mostly substitutions and few or no indels) and a high error model (where the indel rate is high); see text for details about the error models for all datasets. Datasets labelled 'TIPP' were generated for this study; the rest are datasets previously studied in the literature. 'Experiment' specifies the experiments where each dataset is used. The low indel versions are studied in Experiments 1 and 3, and the high indel versions are studied in Experiments 2 and 3. The number of genomes in the dataset is given in the column labelled '# Genomes'. The complexity of the dataset is given in the column labelled 'Complexity'. The number of reads in the dataset is given in the column labelled '# Reads'. 'Length' refers to the average length of the reads for the low indel version of the dataset ('s' and 'l' refer to short and long).

3.1.7 Experimental condition 3: datasets with novel genomes We generated datasets that contained 'novel' genomes, which are genomes that were not in the training sets of the marker-based methods. As the marker-based methods use older versions of the NCBI genomes compared to the composition-based methods, some of the selected genomes were present in the training set of the composition-based methods. To make the genomes novel to all methods, we modified the training sets of the composition-based methods to exclude the selected genomes. This modification was only applied to Experiment 3, where we analyzed the novel genome datasets.

We downloaded all genomes from NCBI in July 2014 and selected any genome whose genus was found in the training sets of all methods, but the species was not found in any of the training sets of the marker-based methods. We then excluded the species of the selected genomes from the training sets of the composition-based methods. This resulted in a set of genomes that were novel to all methods at the species level, but not novel to any of the methods at the genus level. In total, 100 genomes met this criterion.

We used MetaSim to generate four datasets containing all 100 genomes, each dataset varying the complexity of the profile (LC versus HC) and the sequencing model used to generate the reads (short Illumina reads versus long 454 reads). We categorize these datasets as 'easy' novel genome datasets as they contained no indels (datasets with short Illumina reads) or low amounts of indels (5% indel error per base for the datasets with long 454 reads). In addition, we generated 'hard' novel genome datasets by modifying the 'easy' novel genome datasets to include additional indel errors (12% insertion and 2% deletion error rate per base).

3.1.8 Computing profiles MetaPhlAn, mOTU, MetaPhyler and TIPP output an abundance profile from a set of query sequences. NBC and PhymmBL output the classification of the query sequences; abundance profiles for NBC and PhymmBL were estimated using the relative abundance of the query sequence classification results.

3.1.9 Performance evaluation We compute the Hellinger distance (Rao, 1995) between the estimated abundance profile and the true abundance profile. Let C_l be the set of clades found in the true profile and the estimated profile for the taxonomic level l, R_x be the abundance of clade x for the true profile, and E_x be the abundance of clade x for the estimated profile. Then H_l (the Hellinger distance for taxonomic level l) is:

$$H_l = \frac{\sqrt{\sum_{x\in C_l}(\sqrt{R_x}-\sqrt{E_x})^2}}{\sqrt{2}} \qquad (7)$$

H_l ranges from 0 (if the profiles match exactly) to 1 (if the profiles share no taxa in common and the profiles have no unclassified clades). Note that if a method estimates a portion of the clade as unclassified, that portion is not included in the H_l calculation (i.e. $E_{\text{unclassified}}$ is omitted from the formula). Finally, the MetaPhlAn HC and LC datasets consist of two and eight replicates; for these datasets, we report the average H_l of the replicates.

3.1.10 Computational resources The majority of experiments were run on the homogeneous Stampede cluster at the Texas Advanced Computing Center (TACC). Each method was run on a dedicated node with 16 cores with 32 GB of memory and given 48 h to complete on TACC (maximum allotted time that a job can run on Stampede). Methods that could not complete within the 48-h limit were also run on a heterogeneous Condor computing cluster and homogeneous Lonestar cluster.

4 RESULTS

Experiment 1: Easy datasets. Our first experiment explored datasets with low error rates coming from known genomes. On the long read easy datasets, mOTU had generally poor performance (Supplementary Table S2), and also failed to run on some of these datasets (terminated with an error message, see Supplementary Section S1.6); we, therefore, omit mOTU from the long read results.

Figure 2a shows the average Hellinger distances on the easy datasets; figures and tables for individual datasets can be found in Supplementary Section S1.1. Results on the long read datasets (left subfigure) show that TIPP had the best overall accuracy, followed by NBC. MetaPhlAn was in third place in terms of overall performance (though Metaphyler was more accurate than MetaPhlAn at the class level). Finally, PhymmBL was the

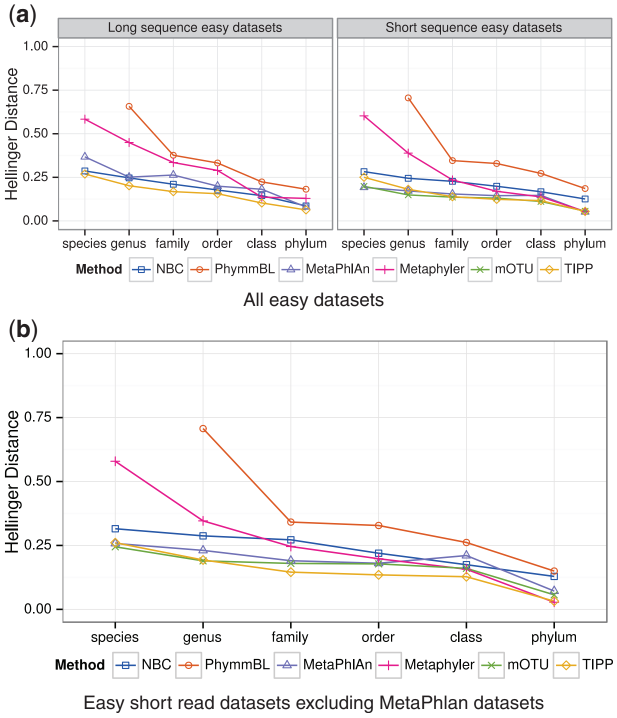

Fig. 2. Experiment 1: Errors in abundance profile estimates on easy datasets. We show average Hellinger distance for different methods on simulated metagenome datasets containing known genomes and query sequences with low rates of sequencing error (see Table 1 for list of datasets). We omit mOTU from the long read results because it failed to complete on the FACS HC and WebCarma datasets. Results are shown for (a) all datasets, and (b) easy short read datasets after excluding the MetaPhlan datasets (thus on WebCarma Illumina and FACS HC Illumina datasets). See Section S1.1 for results on individual datasets

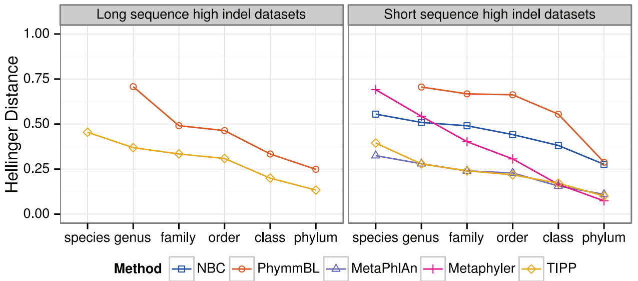

Fig. 3. Experiment 2: Error in abundance profiles for datasets with known genomes and high indel error rates. We show average Hellinger distance for different methods on simulated metagenome datasets with known genomes and simulated with indel errors. We omit mOTU from the results because it failed to complete on any of these datasets. We omit MetaPhlAn, MetaPhyler, and NBC from the long read datasets as they could not classify any reads from at least one of the following datasets: WebCarma (MetaPhlan), FACs HC (NBC and MetaPhyler), and FAMeS LC (MetaPhyler). See Section S1.2 for results on individual datasets

least accurate on these data. On the short read datasets (right subfigure), results were somewhat different. First, while mOTU had poor performance on the long read datasets, mOTU had the best overall accuracy of all methods on the short read datasets. At the species level, MetaPhlAn tied with mOTU for first place, followed by TIPP and then NBC. At the genus level, mOTU had the best accuracy, but TIPP and MetaPhlAn were very close seconds. Results at the family through class levels showed TIPP and mOTU tying for first, with MetaPhlAn in third place. Finally, at the phylum level, TIPP, MetaPhlAn, Metaphyler and mOTU tied for first, with much better accuracy than NBC or PhymmBL.

Because MetaPhlAn was trained on its datasets, we removed MetaPhlAn's datasets and re-evaluated performance (Fig. 2b). On this reduced set, MetaPhlAn tied for first at the species level (slightly less accurate than mOTU, but equal to TIPP), but was much less accurate than TIPP at all other levels. TIPP had the best overall performance on these data, though it tied for first with mOTU at the genus level, and tied for first with PhymmBL at the phylum level.

Experiment 2: Datasets with known genomes and high rates of indel errors. Figure 3 shows the average performance of different methods on datasets with known genomes and simulated with high rates of indel errors. We omit results for mOTU because it terminated with error messages on all these datasets (see Supplementary Section S1.6). We also omit results for MetaPhlAn, MetaPhyler and NBC on the long read datasets as they estimated taxonomic profiles that were completely unclassified on some of the long read datasets.

On the long read datasets (left subfigure), only TIPP and PhymmBL successfully ran on all the datasets. Other methods either terminated early or could not classify fragments on some of the datasets. TIPP had the best accuracy of all methods at all taxonomic levels. PhymmBL was the next best method.

On the short read datasets (right subfigure), MetaPhlAn and TIPP had the best results, with MetaPhlAn more accurate than TIPP at the species level but the two methods generally having indistinguishable performance at the other levels. PhymmBL had the least accurate results of all methods, followed by NBC. MetaPhyler's performance was interesting: least accurate at the species level, but improving rapidly with the higher taxonomic groups, so that it matched the best at the class level, and was slightly more accurate than both TIPP and MetaPhlAn at the phylum level. Finally, when the MetaPhlAn datasets were removed from the experiment, TIPP had the best overall performance, followed closely by MetaPhlAn (see Supplementary Fig. S3); the relative performance of the remaining methods was largely similar to the average results on the high indel short read datasets.

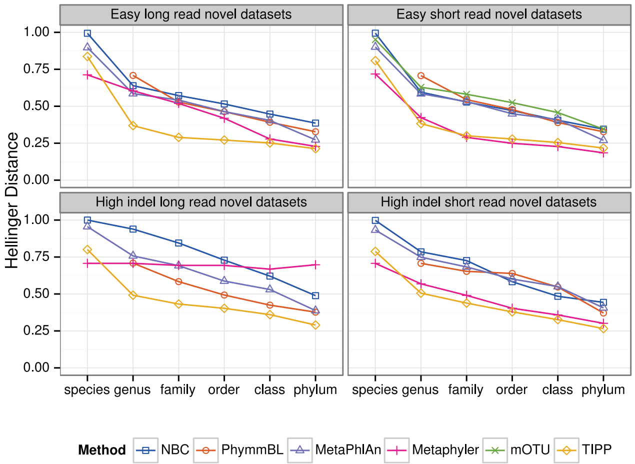

Fig. 4. Experiment 3: Error in abundance profiles on datasets with novel genomes. We show average Hellinger distance for different methods on simulated metagenome datasets containing novel genomes. The top row shows results for datasets with little to no indel errors, and the bottom row shows results for datasets with high rates of indel errors (12% insertion and 2% deletion errors per base). The left column shows results for datasets containing long reads, and the right column shows results for datasets containing short reads. See Section S1.3 for results on individual datasets

Experiment 3: Datasets containing novel genomes. Figure 4 shows the average Hellinger distance for the 'easy' and 'hard' novel metagenome datasets, with the long read datasets in the left column and the short read datasets in the right column (figures and tables for individual datasets can be found in Supplementary Section S1.3). Because all genomes are novel (none of the species are in the training datasets), the lowest error results at the species level would have 100% of the species unclassified, and such a profile would have a Hellinger distance of $\frac{1}{\sqrt{2}} \approx 0.71$.

On the easy long read novel genome datasets (top left), TIPP had the best accuracy at all but the species level (where MetaPhyler was the best). The next best method was MetaPhyler, and the remaining methods had comparable performances to each other (though NBC was slightly less accurate than the others at all levels). On the high indel long read novel genome datasets (bottom left), TIPP again had the best accuracy at all but the species level, where Metaphyler was best. But on these more difficult long read datasets, Metaphyler had very poor results at the remaining levels, largely because it characterized very few reads (see Supplementary Table S8). Finally, PhymmBL was clearly more accurate than MetaPhlAn, which was more accurate then NBC.

On the short read novel genome datasets (right column), TIPP and MetaPhyler were distinctly more accurate than all other methods for both easy datasets and high indel datasets (with TIPP slightly more accurate in the presence of high indels, and MetaPhyler slightly more accurate on the easy datasets). The performance of mOTU was interesting: on the easy short read novel genome datasets, mOTU was very close to PhymmBL but slightly less accurate; however, mOTU failed to complete on the high indel short read datasets (terminated with an error message, see Supplementary Section S1.6). The other methods were largely indistinguishable from each other, but much less accurate than both TIPP and MetaPhyler.

4.1 Running time

We generated five replicates from the easy TIPP HC datasets, varying the total number of fragments from 500,000 to 2,000,000 fragments. PhymmBL and NBC failed to complete within 48 h on any of these datasets (see Supplementary Section S1.4), but the other methods completed all analyses, most within an hour. The fastest method was mOTU, which completed in <7 min on 2,000,000 fragments for both short and long reads. TIPP finished in under an hour on both short and long fragments. MetaPhlAn required <25 min on short fragments, but used 90 min on long fragments. MetaPhyler completed in <13 min for short fragments, but used 29 h for long fragments.

5 DISCUSSION

The study compared four marker-based methods (mOTU, TIPP, MetaPhyler and MetaPhlAn) and two composition-based methods (PhymmBL and NBC) on a collection of datasets. On the easy datasets with only known genomes (Experiment 1), many methods produced highly accurate abundance profiles, although which method was the best depended on the read length. On the easy short read datasets the most accurate methods were mOTU, MetaPhlAn and TIPP, while the most accurate methods on the easy long read datasets were TIPP and NBC, with MetaPhlAn a close third.

However, when datasets had novel genomes or high indel error rates, the performance of most methods degraded significantly. For example, while NBC had excellent accuracy on the easy long read datasets, it had poor accuracy in the presence of high indel rates. Similarly, although mOTU had excellent accuracy on easy short read datasets with known genomes, it had poor performance on short read datasets with novel genomes and terminated with an error message on the datasets with high indel rates. We also saw that MetaPhyler had difficulties on datasets with long sequences and high indels (Experiments 2 and 3).

TIPP did well on the easy datasets (known genomes with low sequencing error rates), where it tied for first with other methods. However, TIPP was the only method that was robust to all the tested model conditions (short versus long reads, novel versus known genomes, low indel versus high indel error rates), so that it was either first or tied for first under even very difficult conditions (novel genomes with high indel rates). Consequently, TIPP dominated the other methods in terms of overall performance.

Thus, one of the main features of TIPP is its relative robustness to sequencing errors (both substitutions and indels) and its ability to perform well on novel genomes. We conjecture that the HMM Family technique within TIPP provides this robustness, which uses HMMs in a divide-and-conquer framework, and also benefits from the use of local alignment techniques.

Therefore, the high error rates commonly encountered in single-molecule sequencing technologies [up to 14% indel rate for Pacific Biosciences technologies (Carneiro *et al*., 2012)] that are likely to be increasingly used in the context of metagenomic data may not be particularly problematic for TIPP, even when the data contain novel genomes. Instead, our results indicate that TIPP may well continue to have good accuracy with new sequencing technologies that result in reduced data quality.

One of the interesting observations in this study is that the marker-based methods (TIPP, MetaPhlAn, mOTU and MetaPhyler) often gave more accurate abundance profiles than the composition-based methods (NBC and PhymmBL), even though marker-based methods estimate abundance profiles by characterizing only those reads that map to their selected marker genes. These results show that accurate profiles can be obtained by taxonomic characterization of only a fraction of the query sequences. However, the choice of marker genes, and the technique used to bin the reads to the markers, has an impact on the resultant abundance profile. Thus, future research should investigate whether improved performance can be obtained using a different set of marker genes and different techniques to map reads to the marker genes. Also, our abundance profile estimates were based on combining all reads for all markers into one set, and using the distribution estimated for that set; more sophisticated techniques could be used to combine distributions estimated for each marker.

ACKNOWLEDGEMENTS

The authors thank Sean Eddy for his help with the derivations of the probability that a query sequence is generated by one HMM in a set of HMMs. The authors also thank the Huttenhower lab for sharing the synthetic data with us for the taxonomic profiling experiment.

Funding: This research was partially supported by a Guggenheim Foundation Fellowship to T.W.; National Science Foundation grants DBI-1062335, DBI-1461364 and DEB 0733029 to T.W.; an HHMI International Predoctoral Fellowship to S.M.; the iPlant Collaborative U.S. National Science Foundation grant DBI-1265383 (via TACC) to N.N.; and National Institutes of Health grant R01-AI-100947 to M.P. Some of this work was performed while T.W. was a program director working for the US National Science Foundation, and supported by the IR/D program.

REFERENCES

Abe,T. *et al.* (2006) A novel bioinformatics tool for phylogenetic classification of genomic sequence fragments derived from mixed genomes of uncultured environmental microbes. *Polar Biosci.*, **20**, 103–112.

Altschul,S., Gish,W. and Miller,W. (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.

Brady,A. and Salzberg,S.L. (2009) Phymm and PhymmBL: metagenomic phylogenetic classification with interpolated Markov models. *Nat. Methods*, **6**, 673–676.

Brady,A. and Salzberg,S.L. (2011) PhymmBL expanded: confidence scores, custom databases, parallelization and more. *Nat. Methods*, **8**, 367.

Carneiro,M.O. *et al.* (2012) Pacific biosciences sequencing technology for genotyping and variation discovery in human data. *BMC Genomics*, **13**, 375.

Eddy,S.R. (1998) Profile hidden Markov models. *Bioinformatics*, **14**, 755–763.

Ewing,B. and Green,P. (1998) Base-calling of automated sequencer traces using phred. II. Error probabilities. *Genome Res.*, **8**, 186–194.

Gerlach,W. and Stoye,J. (2011) Taxonomic classification of metagenomic shotgun sequences with CARMA3. *Nucleic Acids Res.*, **39**, e91.

Huson,D.H. *et al.* (2007) MEGAN analysis of metagenomic data. *Genome Res.*, **17**, 377–386.

Koski,L.B. and Golding,G.B. (2001) The closest BLAST hit is often not the nearest neighbor. *J. Mol. Evol.*, **52**, 540–542.

Liu,B. *et al.* (2010) Metaphyler: Taxonomic profiling for metagenomic sequences. In: *2010 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. pp. 95–100.

Liu,B. *et al.* (2011) Accurate and fast estimation of taxonomic profiles from metagenomic shotgun sequences. *BMC Genomics*, **12**, S4.

Liu,K. *et al.* (2009) Rapid and accurate large-scale coestimation of sequence alignments and phylogenetic trees. *Science*, **324**, 1561–1564.

Liu,K. *et al.* (2012) SATé-II: very fast and accurate simultaneous estimation of multiple sequence alignments and phylogenetic trees. *Syst. Biol.*, **61**, 90–106.

Matsen,F.A., Kodner,R.B. and Armbrust,E.V. (2010) pplacer: linear time maximum-likelihood and Bayesian phylogenetic placement of sequences onto a fixed reference tree. *BMC Bioinformatics*, **11**, 538.

Mavromatis,K., Ivanova,N. and Barry,K. (2007) Use of simulated data sets to evaluate the fidelity of metagenomic processing methods. *Nat. Methods*, **4**, 495–500.

McHardy,A.C. *et al.* (2007) Accurate phylogenetic classification of variable-length DNA fragments. *Nat. Methods*, **4**, 63–72.

Mende,D.R. *et al.* (2013) Accurate and universal delineation of prokaryotic species. *Nat. Methods*, **10**, 881–884.

Patil,K.R. *et al.* (2011) Taxonomic metagenome sequence assignment with structured output models. *Nat. Methods*, **8**, 191–192.

Rao,C. (1995) A review of canonical coordinates and an alternative to correspondence analysis using Hellinger distance. *Questiio*, **19**, 23–63.

Richter,D.C. *et al.* (2008) MetaSim: a sequencing simulator for genomics and metagenomics. *PloS One*, **3**, e3373.

Rosen,G.L., Reichenberger,E.R. and Rosenfeld,A.M. (2011) NBC: the Naive Bayes Classification tool webserver for taxonomic classification of metagenomic reads. *Bioinformatics*, **27**, 127–129.

Segata,N. *et al.* (2012) Efficient metagenomic microbial community profiling using unique clade-specific marker genes. *Nat. Methods*, **9**, 811–814.

Shendure,J. and Ji,H. (2008) Next-generation DNA sequencing. *Nat. Biotechnol.*, **26**, 1135–1145.

Stamatakis,A. (2006) RAxML-VI-HPC: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models. *Bioinformatics*, **22**, 2688–2690.

Stranneheim,H. *et al.* (2010) Classification of DNA sequences using Bloom filters. *Bioinformatics*, **26**, 1595–1600.

Sunagawa,S. *et al.* (2013) Metagenomic species profiling using universal phylogenetic marker genes. *Nat. Methods*, **10**, 1196–1199.