Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

Subject Section

Evaluating performance of metagenomic characterization algorithms using *in silico* datasets generated with FASTQSim

Anna Shcherbina [1], Darrell O. Ricke [1,*] and Nelson Chiu [1]

[1] Bioengineering Systems and Technologies Group, MIT Lincoln Laboratory, Lexington, MA, 02420, USA

* To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: *In silico* bacterial, viral, and human truth datasets were generated to evaluate available metagenomics algorithms. Sequenced datasets include background organisms, creating ambiguity in the true source organism for each read. Bacterial and viral datasets were created with even and staggered coverage to evaluate organism identification, read mapping, and gene identification capabilities of available algorithms. These truth datasets are provided as a resource for the development and refinement of metagenomic algorithms. Algorithm performance on these truth datasets can inform decision makers on strengths and weaknesses of available algorithms and how the results may be best leveraged for bacterial and viral organism identification and characterization.

Results: Algorithms were evaluated on runtime, true positive organisms identified to the genus and species levels, false positive organisms identified to genus and species level, read mapping, relative abundance estimation, and gene calling. No algorithm out performed the others in all categories, and the algorithm or algorithms of choice strongly depends on analysis goals. MetaPhlAn excels for bacteria and One Codex, followed by LMAT, for viruses. The algorithms were ranked by overall performance using a normalized weighted sum of the above metrics, and MetaScope emerged as the overall winner, followed by One Codex, Kraken and LMAT. Simulated FASTQ datasets with well-characterized truth data about microbial community composition reveal numerous insights about the relative strengths and weaknesses of the metagenomics algorithms evaluated.

Availability: The simulated datasets are available to download from the Sequence Read Archive (SRP062063).

Contact: Darrell.Ricke@ll.mit.edu

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

1 Background

Continuing advances in sequencing technologies are increasing the feasibility of sequencing entire microbial communities rather than individual organisms. This has led to rapid developments in the field of metagenomics aimed at studying genomic material recovered directly from environmental and medical samples. Sequencing the metagenome enables the capture of greater genetic diversity than can be sampled with highly targeted approaches such as microarrays. Metagenomic sequencing has a number of applications for medical diagnostics (i.e. human gut microbiome analysis), environmental profiling (i.e. soil samples), and homeland defense(16)-(34). Metagenomic techniques also enable the study of communities of organisms simulated in vitro(20). Simultaneously, a number of bioinformatics tools have been developed to analyze metagenomic sample data. They employ a variety of techniques to achieve the opposing goals of high accuracy and low runtime. In this study,

* This work is sponsored by the Assistant Secretary of Defense for Research and Engineering under Air Force Contract #FA8721-05-C-0002. Opinions, interpretations, recommendations and conclusions are those of the authors and are not necessarily endorsed by the United States Government.

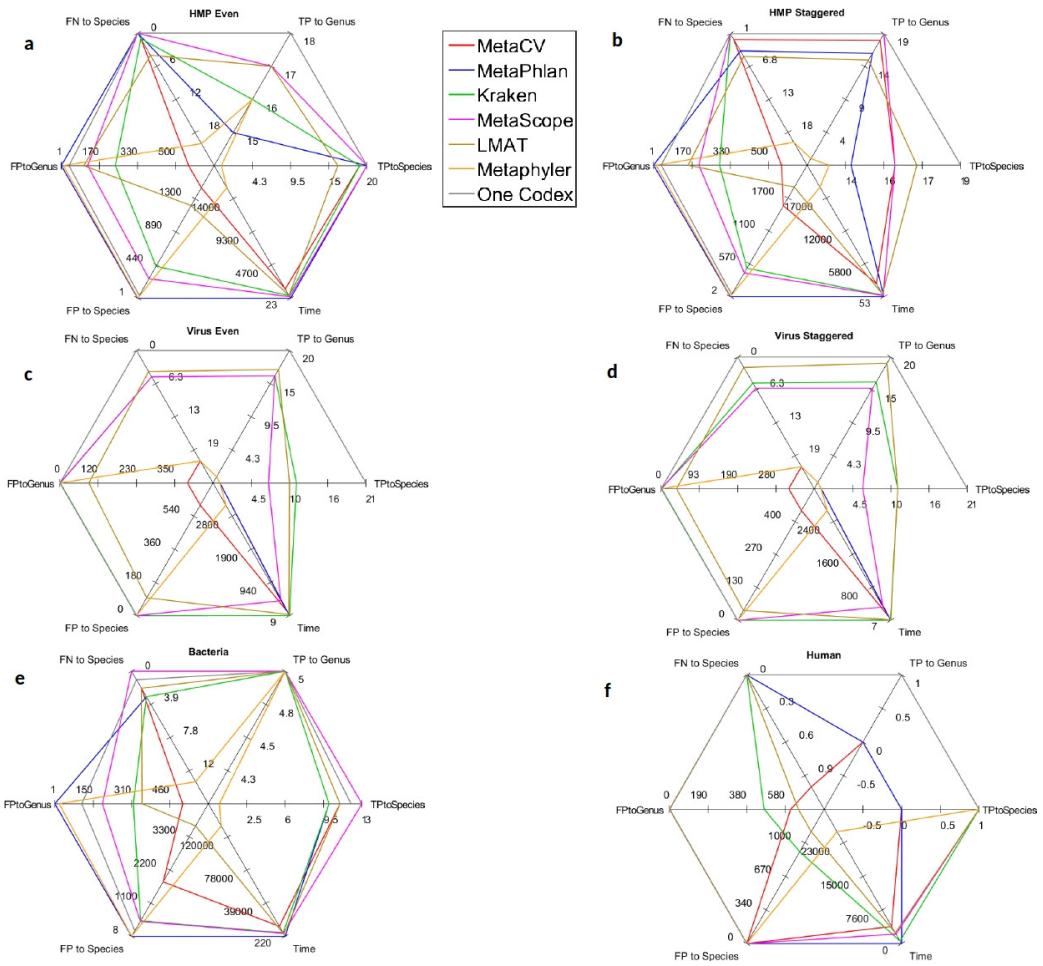

Fig. 1. Performance metrics for 6 metagenomic analysis algorithms across the 6 in silico evaluation datasets. Algorithms evaluated include MetaCV (red line), MetaPhlAn (blue line), Kraken (green line), MetaScope (pink line), LMAT (brown line), MetaPhyler (orange line), One Codex (gray line). Metrics evaluated include true positives (TP) to genus level, TP to species level, false positives (FP) to genus level, FP to species level, false negatives (FN) to species level, and runtime in seconds. Values indicative of high performance are at the periphery of the radar plot, values indicative of poor performance are at the center of the plot. a. HMP dataset with even coverage. b. HMP dataset with staggered coverage. c. Virus dataset with even coverage. d. Virus dataset with staggered coverage. e. Bacterial dataset. f. Human dataset. The MetaPhlAn algorithm failed to run on the human dataset.

the performance of these varied approaches to metagenomic sequence classification was evaluated on a suite of *in silico* datasets with perfectly characterized composition. MetaScope, winner of the Defense Threat Reduction Agency's Grand Challenge for identifying organisms from a stream of DNA sequences (`https://www.innocentive.com/ar/challenge/9933138`) relies on sequence analysis using spaced seeds followed by an augmented least common ancestor algorithm to map reads and assign genes for input FASTQ samples(13),(7). Kraken(36) uses exact alignment of k-mers in combination with an optimized database and another version of the least common ancestors algorithm. MetaPhlAn(28) relies on unique clade-specific marker genes identified from 3000 reference genomes. The Livermore Metagenomic Analysis Toolkit (LMAT) exploits genetic relationships between different organisms by pre-computing the occurrence of each short sequence across the entire reference database and storing the evolutionarily conserved sequence patterns(2)-(33). MetaCV translates nucleotide sequences into six frame peptides, which are then decomposed into k-mers. The k-mer frequency is computed in a protein-reference database and used to assign k-mer weights(15). MetaPhyler uses a precomputed database of reference phylogenetic marker genes to build a sequence classifier. The classifier, based on BLAST, uses trained thresholds for various combinations of taxonomic ranks, sequence length, and reference genomes(14). Finally, One Codex(19) identifies microbial sequences using a k-mer based taxonomic classification algorithm through a web-based data platform, using a reference database of 40,000 bacterial, viral, fungal, and protozoan genomes. In this study, the performance of these varied approaches to metagenomic sequence classification was evaluated on a suite of *in silico* datasets with perfectly characterized composition.

Simulated *in silico* datasets are a valuable tool for metagenomic research and provide capabilities to evaluate algorithm performance as well as to test hypotheses that cannot be examined through empirical observation. For example, simulated data has revealed biases and heterogeneity in the estimation of diversity metrics from metagenomics samples(5). Additionally, multiple studies have demonstrated the usefulness of simulated metagenomics datasets for benchmarking sequence assembly and gene prediction pipelines(17)-(18). Simulated datasets are also an effective means of parameter optimization for improved algorithm performance and can be used to optimize study design. Sequence simulation can aid with answering questions about coverage requirements, necessary sequence length, and whether paired-end or single-end sequencing should be used. For example, the ART simulator was successfully used by the 1000 Genomes Project Consortium

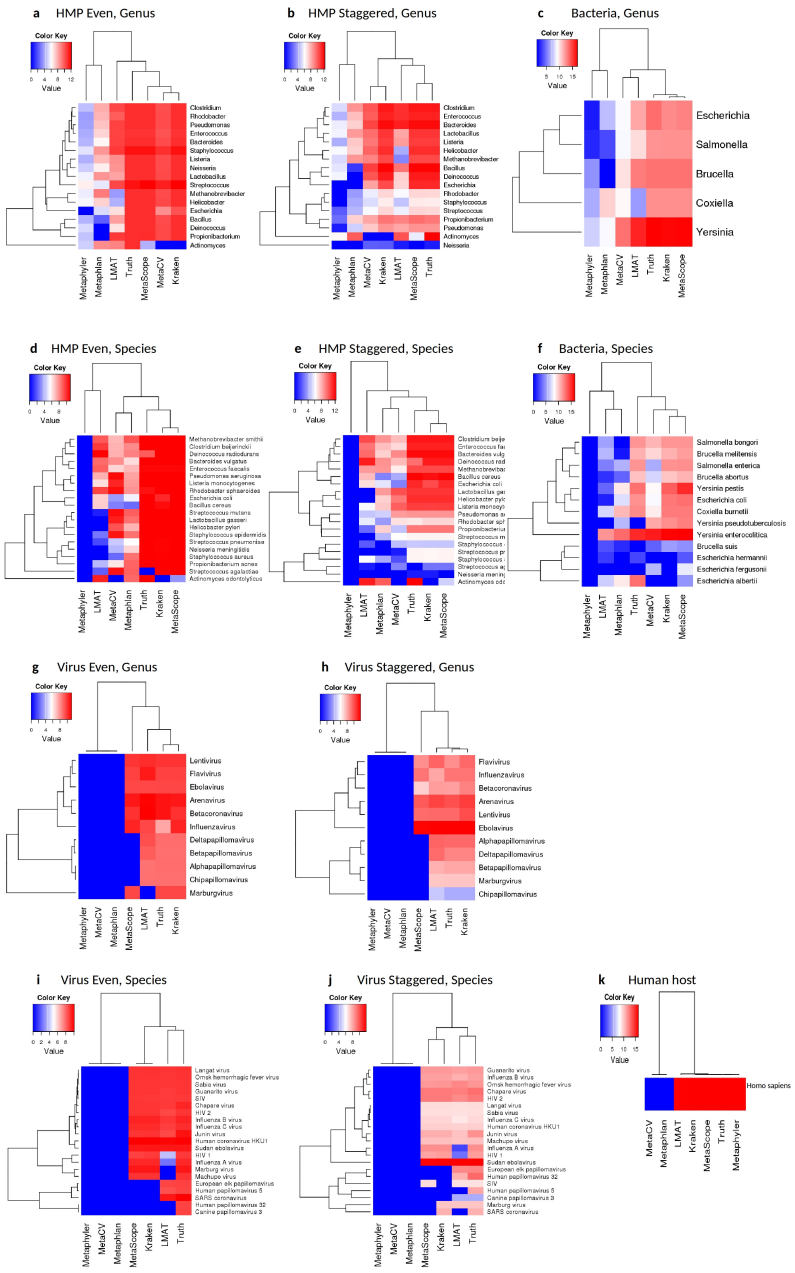

Fig. 2. Number of correctly assigned reads to each organism at the genus and species level. Heatmap color scales are log10 (number of correctly assigned reads). The "Truth" column indicates the number of reads spiked into the FASTQ input file for the specified genus or species. a.–e. Reads mapped correctly to the genus level for the HMP even, HMP staggered, bacteria, virus even, virus staggered datasets, respectively. f. – k. Reads mapped correctly to the species level for HMP even, HMP staggered, bacteria, virus even, virus staggered, and human datasets, respectively.

to examine the effects of read length and PE insert size on a read's ability to map to the human genome(12). In this study, six *in silico* datasets were simulated by the FASTQsim tool. Figure S1 illustrates the composition of each dataset. These datasets contained sequences from reference bacterial and viral genomes, as most human pathogens are members of these taxa. The HMP Even and HMP Staggered datasets were generated to include sequences from the 20 organisms from the Human Microbiome Project(23) (Supplementary Table S1). The HMP organisms were selected for inclusion after an attempt to benchmark the performance of MetaScope with the HMP dataset revealed potential contamination in the dataset. As the HMP benchmark dataset was generated by sequencing organisms cultured in vitro, there was no absolute truth for any background contaminant organisms in the dataset and it was not possible to determine whether the contamination was real or whether MetaScope was calling false positive organisms.

The bacterial dataset (Supplementary Table S2) was designed to test algorithm specificity. Four genera of pathogens were selected from the National Institute of Allergy and Infectious Diseases (NIAID) list of biodefense and emerging infectious disease agents (`https://www.niaid.nih.gov/topics/biodefenserelated/biodefense/pages/cata.aspx`) due to their relevance to disease diagnostics from metagenomics samples. These included *Yersinia*, *Coxiella*, *Brucella*, and *Salmonella*. Additionally, the *Escherichia* genus was added to the list due to the high abundance of representative sequences in GenBank(4).

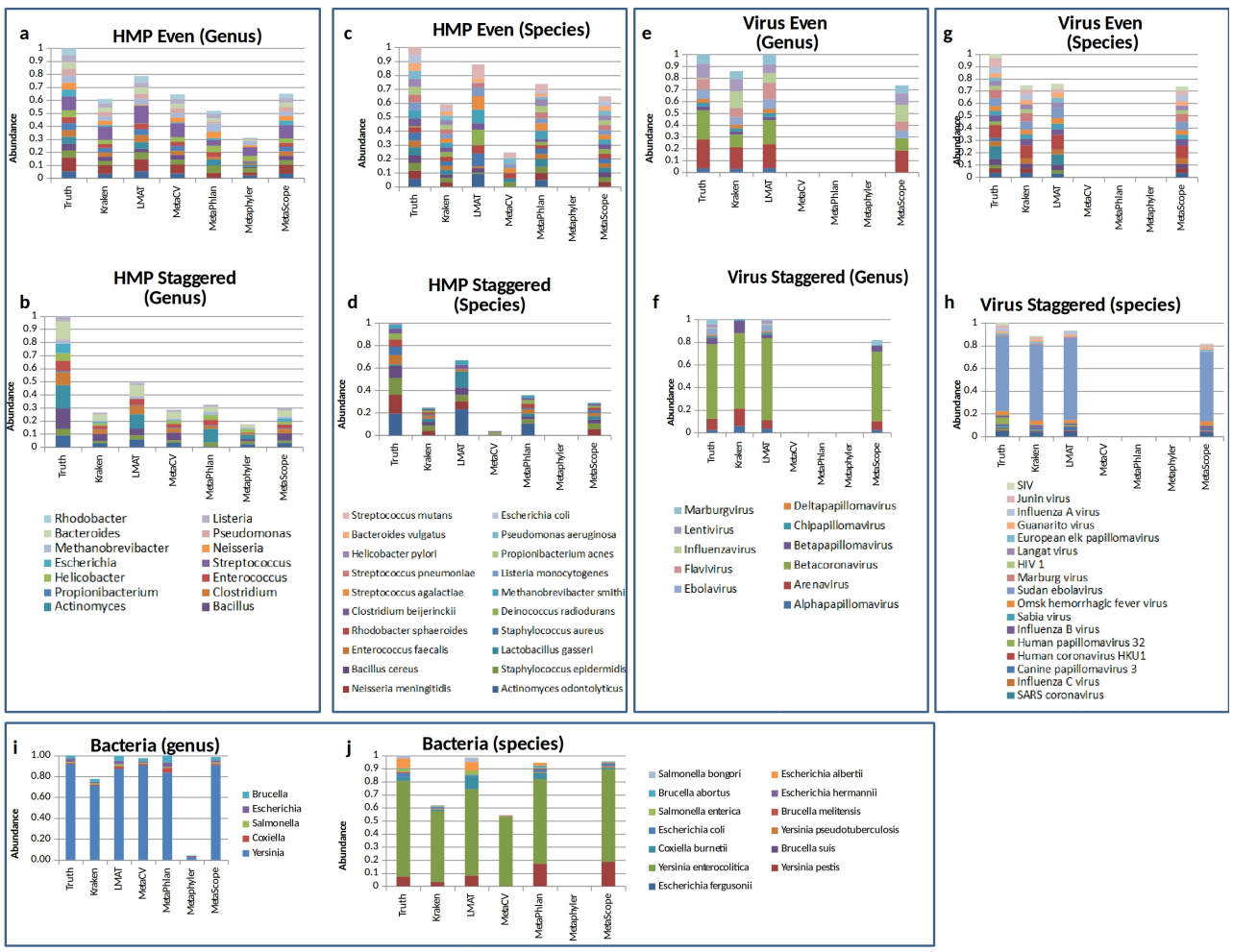

Fig. 3. Relative abundance of organisms to the species and genus level. "Truth" column indicates relative abundance of genera and species added to the in silico FASTQ input file.

For each of the five genera, several representative species were selected (i.e.,*Brucella abortus*, *Brucella melitensis*, *Brucella suis*). Next, several representative strains were selected for each species (i.e. *Brucella melitensis ATCC 23457*, *Brucella melitensis biovar abortus 2308*, *Brucella melitensis biovar 1 strain 16M*, and *Brucella melitensis M28*). Organisms were spiked into a FASTQ dataset with coverage levels ranging from 10x to 0.00002x (1 read).

Two virus datasets were generated with 21 species across 11 representative genera (Supplementary Table S3). As with the bacterial dataset, candidates were selected due to their inclusion on the NIAID list of emerging pathogens (*Marburg virus*, *Machupo virus*, *Sudan ebolavirus*, *Junin virus*, *Guanarito virus*, *Chapare virus*, *Omsk hemorrhagic fever virus*) as well as abundance of representative organisms in GenBank (*HIV1*, *HIV2*, *Influenza A virus*). For the Virus Even dataset, 10x coverage of each organism was simulated. For the Virus Staggered dataset, coverage varied from 100x for Sudan ebolavirus to 0.5x for the Human coronavirus HKU1. Finally, a dataset of human reads from build GRCh38 at 10x (22 million reads) coverage was generated to test host-filtering capabilities of each algorithm. This dataset was generated to measure how well algorithms can overcome the challenges posed by human sequence contamination in public reference databases(8). For example, endogenous retroviral remnants may be incorrectly classified as belonging to viral genomes in a sample(6)-(29).

Table 1. Radar plot area in normalized units across six evaluation datasets. Area with runtime excluded is indicated in parentheses.

Dataset	Human	Vir. Stag.	Virus Even	Bacteria	HMP Stag.	HMP Even	Area Sum
One Codex	(2.37)	(2.37)	(2.37)	(2.052)	(2.31)	(2.31)	(13.8059)
MetaScope	2.54(2.37)	1.9(1.38)	2.14(1.47)	2.23(1.99)	2.03(1.57)	2.32(1.847)	13.15(10.65)
Kraken	1.64(1.34)	2.36(1.63)	1.21(1.63)	1.75(1.48)	1.9(1.44)	1.76(1.32)	10.62(8.85)
LMAT	1.41(1.12)	2.45(1.69)	2.25(1.41)	1.38(1.10)	1.46(1.00)	1.59(1.16)	10.54(7.85)
MetaPhlAn		0.48(0.58)	0.99(0.58)	2.25(1.99)	1.88(1.49)	2.02(1.66)	7.62(7.85)
MetaCV	0.82(0.52)	0.09(0.34)	0.14(0.034)	1.43(1.17)	1.25(0.87)	1.08(0.62)	4.8(3.57)
MetaPhyler	1.88(2.37)	0.6(0.58)	0.09(0.58)	0.67(0.68)	0.57(0.62)	0.63(0.62)	4.44(5.47)

2 Results and Discussion

Runtime in seconds, true positive genus and species identification, false positive genus and species identification, and false negative species calls were determined for each of the metagenomic algorithms (Figure 1).

Among the algorithms evaluated, only MetaScope and One Codex mapped a small number of reads in our datasets to a taxon rank below species. Consequently, although the initial focus of the Bacterial dataset was to assess the ability of the algorithms to distinguish between different strains of the same species, it was decided to evaluate both true and false positives at species and genus level. To determine an overall rank of the algorithms across the datasets, the area occupied by each in the radar plot was computed (Table 1). The One Codex results were submitted by the algorithms' authors after the initial release of this manuscript to bioRxiv, and consequently runtime information/radar plot areas are not presented for this algorithm.

When the polygon area was calculated using the MATLAB polyarea function and summed across all datasets, MetaScope emerges as the winner, with the largest overall area. Kraken and LMAT are the runner-ups, and MetaPhyler performed the worst. If runtime is excluded from the polygon area calculation, One Codex emerges as the winner, followed by MetaScope. In addition to the algorithms' rank overall, several trends can be noted in the individual performance categories. The algorithms diverged in runtime by several orders of magnitude (Table 2). Overall, MetaPhlAn had the shortest runtime. The algorithm had the fastest time on the three bacterial datasets – 22.64 s for HMP Even, 53.3 s on HMP staggered, and 220 s. on Bacteria. The second fastest times for these three datasets were 5 to 10 times slower: 233 s (MetaPhlAn), 261 s (MetaScope), and 2,700 s (LMAT), respectively. MetaPhlAn is able to execute quickly partly because it does not perform a host-filtering step. MetaPhlAn came in second for the virus datasets, with a runtime of 11 seconds on both, compared to 9 and 7 seconds for Kraken. MetaPhlAn failed to run on the human dataset. Kraken, MetaScope, and LMAT exhibited similar runtimes on all datasets, averaging 353 s on HMP Even, 354 s on HMP staggered, and 3,595 s on Bacteria. On the other end of the spectrum, MetaPhyler was an outlier for high runtime, requiring 15,480 s on HMP Even, 19,231 s on HMP staggered, and 129,600 s on Bacteria. In addition to its high speed, MetaPhlAn also achieved the highest accuracy, defined as ratio of true positives to false positives, on the bacterial datasets. It identified all 20 species in the HMP even dataset with only a single false positive organism. On HMP staggered, it missed 4 species out of 20 but reported only 2 false positive species. MetaScope, the runner up, reported a single false negative species but 414 false positives. However, the MetaPhlAn reference database is customized for bacteria, and no support exists at the time of this writing for profiling viruses or eukaryotes. MetaScope achieved the second- highest ratio of true positives to false positives, reporting slightly more true positives and approximately half as many false positives as Kraken. LMAT was the least conservative and reported the highest number of false positive organisms. MetaPhyler made highly conservative calls – false positives were low, but so were true positives. Additionally, MetaPhyler, and MetaCV, as well as MetaPhlAn, did not report results for the viral datasets. Algorithm performance on the Human dataset (Figure 2k) illustrates the efficacy of the host-filtering step for each algorithm. The human reference genome is incomplete(1),(31) and misses regions specific to individual host subjects. These missed regions show up as false positives on the Human evaluation dataset – algorithms assign them to organisms other than the human host because these reads are not removed during the host filtering step. For example, One Codex reported two false positive organisms (but with only 1 read each), illustrating excellent host-filtering capabilities. MetaScope reports 152 organisms, with fewer than 100 reads assigned to each. Kraken has a similar false positive profile; it reports 1,266 species that account for <1% of the reads in the dataset. MetaCV reports 2,998 false organisms with low read count, and LMAT reports 1,118 species that account for less than 0.01% of the reads. MetaPhyler does not report results more specific than the Class taxonomy level for the Human dataset, in line with the conservative approach of this algorithm. MetaPhlAn crashes with a segmentation fault on the Human dataset, which most likely is an artifact of the non-host-filtering approach used by this algorithm.

The algorithms were evaluated based on their ability to correctly map reads and predict relative abundance of the organisms in the data (Figures 2,3). For the bacterial datasets, Kraken, One Codex, and MetaScope classified the highest number of reads correctly for both the genus and species level, and cluster closest to the truth in the dendrogram. However, for the viral datasets, LMAT performed best, classifying the most reads correctly, followed by One Codex. Although the *Actinomyces odontolyticus* (NZ_DS264586.1) organism had the highest coverage (11.3x, 217512 reads) in the HMP staggered dataset, the algorithms on the whole did not perform well on this organism. It was not identified by the Kraken, MetaCV, and MetaPhyler algorithms, and called at a low level by MetaScope (153 reads) (Figure 2g) MetaCV mapped the most reads correctly –108,211 (49.7%) and MetaPhlAn was second best, identifying 22,647 (10.4%) of the reads. None of the algorithms identified any of the 2,045 *A. odontolyticus* genes (Figure 4b). This poor performance likely results from the fact that *A. odontolyticus* genome annotation in GenBank is incomplete(27).

Table 2. Algorithm runtime in seconds across six evaluation datasets.

Dataset	Human	Vir. Stag.	Vir. Even	Bact.	HMP Stag.	HMP Even
MetaScope	2160	327	427	3686	261	233
Kraken	600	7	9	4400	300	400
LMAT	2428	20	39	2700	502	427
MetaPhlAn	Seg Fault	12	12	220	53	23
MetaCV	3873	120	150	11966	2337	1322
MetaPhyler	25200	2640	3100	129600	19231	15480

Conversely, at the species level, six of the seven algorithms mapped a high number of reads to *Streptococcus agalactiae* for both the HMP even and HMP staggered datasets (Figure 2f, 2g), but only a small number of reads for this organism were present in the truth data. The relative abundance of *Streptococcus mutans* is lower in the algorithm calls as compared to truth, while the relative abundance of *Streptococcus agalactiae* is higher, suggesting that a number of the reads called for *S. agalactiae* are actually from *S. mutans* (Figure 3b, 3d). This implies difficulty distinguishing between closely related species. Similarly, a high number of reads are assigned correctly to the *Yersinia* and *Escherichia* genera by Kraken and MetaScope (Figure 2c.) However, the algorithms under-assign reads for *Escherichia albertii* and over-assign reads for *Yersinia pseudotuberculosis*, which indicates difficulty in distinguishing between these species (Figure 2h). Overall, algorithms were equally as able to identify organisms in the staggered datasets as in the even datasets, suggesting that accurate read mapping depends more on the database supplied to the algorithm rather than the abundance of the organism in the dataset. Additionally, for the bacterial datasets, Kraken, MetaScope, LMAT, and MetaPhlAn generally agreed on read mapping assignments. However, for the viral datasets, the algorithms missed different sets of organisms – i.e., in Figure 3i, LMAT failed to map reads for *HIV1*, *Influenza A virus*, *Marburg virus*, and *Machupo virus*, whereas MetaScope and Kraken correctly mapped reads for these organisms. However, MetaScope and Kraken both failed to map reads for *Human papillomavirus 5*, *SARS coronavirus*, *Human papillomavirus 32*, and *Canine papillomavirus 3*, while LMAT succeeded in mapping reads for these organisms. This suggests that for viral datasets, it might be worthwhile to execute both LMAT and one of Kraken or MetaScope, and calculate the union of the results.

The algorithms were also evaluated based on false positive hits (Figure S2). MetaCV and LMAT have diverse error profiles – small numbers of reads are mapped to a high number of false positive organisms. Our past experiences with the MetaScope algorithm suggest that this false positive profile indicates an algorithm has difficulty classifying organisms that are not present in the reference database. Ideally, when an algorithm encounters a novel organism, it should regress up the taxonomic tree until a nearest neighbor for the unknown organism can be established. However, the algorithm may instead report all reference organisms that match the unknown sample to a certain threshold. In contrast, Kraken has a highly concentrated error profiles; fewer than 20 false positive organisms are reported, but several thousand reads are mapped to each of

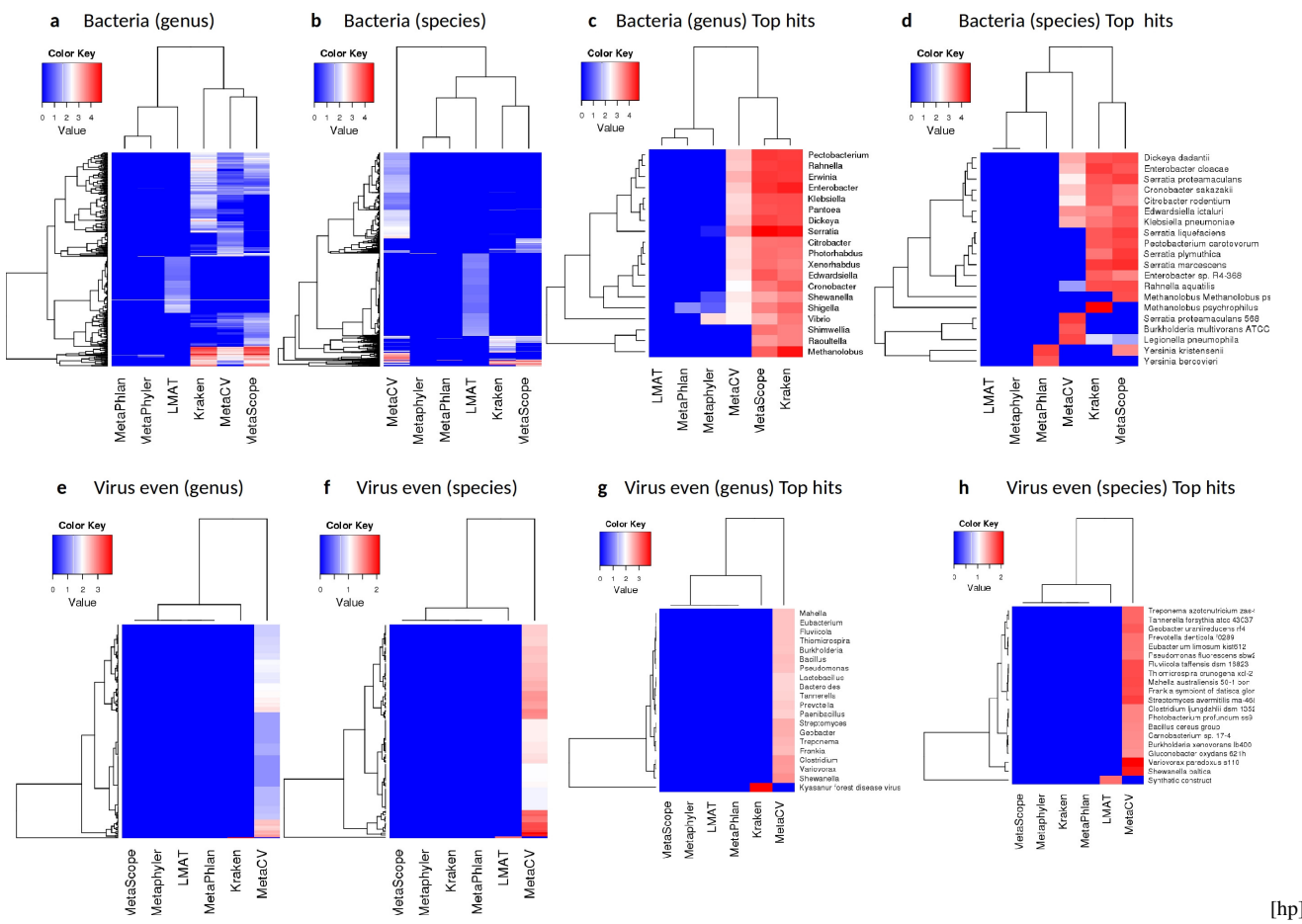

Fig. 4. Number of genes correctly identified to the species level across the 5 evaluation datasets (the 6th evaluation dataset consisting of human host reads is not shown). âŁœTruthâŁž column indicates the number of genes with non-zero read coverage in the dataset. MetaScope, MetaCV, and LMAT algorithms provide gene assignment capabilities; Kraken, MetaPhyler, and MetaPhlAn do not call genes and were not included in this evaluation.

them, suggesting high confidence calls. Figure S2c and S2d summarizes the top 20 organisms in terms number of mapped reads, indicating high agreement between Kraken and MetaScope. On the list of false positive genera are several members of the Enterobacteriaceae family, including *Shigella*, *Klebsiella*, and *Enterobacter*. The true positive genera *Salmonella*, *Escherichia*, and *Yersinia* are members of this family as well. More difficult to explain is the presence of the *Methanolobus genus*, which is a member of the kingdom Archaea and is distantly related to the bacteria in the truth data. For the viral datasets, MetaCV returned a high number of false positives and exhibited poor performance. LMAT, One Codex, and MetaScope did not report any false positive organisms for either viral dataset.

Finally, the gene calling capabilities of the algorithms were evaluated (Figure 4). Only MetaScope, LMAT, and MetaCV call genes, so these three were included for analysis. For the HMP Even/Staggered, Bacteria, and Virus Staggered datasets, MetaScope identified the most genes correctly out of the three algorithms. LMAT identified more correct genes on the Virus Even dataset (101, compared to 93 for MetaScope).

3 Conclusions

In summary, *in silico* datasets with known truth data for read and gene distribution across different taxons serve as a valuable tool for evaluating algorithm performance. The HMP Even/Staggered, Bacteria, Virus Even/Staggered, and Human datasets generated with FASTQsim elucidate multiple patterns in performance for leading metagenomics algorithms. No algorithm out performed the others in all categories, and the algorithm of choice strongly depends on analysis goals. For bacterial datasets, MetaPhlAn is a clear winner, achieving the lowest runtime, highest ratio of true positives to false positives, and the most precise read mapping. However, MetaPhlAn does not assign genes and does not work on taxons other than Bacteria. LMAT and One Codex were winners on the viral datasets in terms of accuracy, and LMAT also provides gene calling functionality. LMAT and One Codex algorithms most closely matched the relative abundance profile of the truth genera and species across all datasets. However, LMAT also reported the highest rate of false positive genera and species calls on the bacterial datasets. Kraken and MetaScope were the runners up in terms of runtime, ratio of true positives to false positives, and read mapping. MetaScope also performed best for gene mapping, which Kraken does not do. These algorithms performed solidly across all categories evaluated and can be applied most universally across versatile metagenomic applications. MetaPhyler and MetaCV came in last for runtime, ratio of true positives to false positives, and read mapping. They also do not provide results out of the box for viral datasets. Although viral, bacterial, and human datasets were simulated for this study, the techniques described here can be extended to evaluate metagenomic algorithm performance for other taxa. For example, fungal contamination incidents at medical facilities such as the 2012 incident at

the New England Compounding Center(35) can be contained more quickly and effectively with the aid of metagenomic sequencing. Other potential applications include rapid diagnosis of parasite infections(22).

4 Methods

4.1 Improvements to FASTQsim

The FASTQsim toolkit was augmented to annotate gene information for simulated reads(30). The "FASTQmapGenes" functionality was added, allowing users to specify NCBI accession ids to use for annotating gene information in simulated reads. The FASTQsim toolkit uses the Entrez and SeqIO libraries from BioPython(9) to download the specified files from GenBank in .gb format. The GenbankParser (`https://github.com/doricke/BioTools/tree/master/GenBankParser`) Java application is then used to parse the .gb files in order to extract all information encoded in the CDS and Gene tags. These gene and CDS annotations are appended to the headers within the simulated FASTQ files generated by FASTQsim, such that all reads that fall within a CDS or gene region are annotated with the corresponding CDS and gene information.

4.2 *In silico* data generation

The FASTQsim toolkit was used to generate six in silico datasets. All were generated with the Illumina error and read length profile included with FASTQsim version 2.0, with no host background added. Specifically, read length of 150 bases was used, with single base mutation, insertion, and deletion rates as specified in the FASTQsim v. 2.0 documentation `http://sourceforge.net/p/fastqsim/code/ci/master/tree/params/illumina/`. NCBI identifiers for all input data are listed in Supplementary Tables S1-3. The commands used to generate the datasets are listed in Supplementary File 1. The Krona toolkit(24) was used to visualize evaluation dataset composition. Two in silico datasets were generated – "HMP Even" and "HMP Staggered" (Supplementary Table 1). For the HMP even dataset, FASTQsim was executed to provide equal number of reads for each species of organism (approximately 60,000 reads per species), with one exception – 559 reads for *Streptococcus agalactiae* were added to simulate a low-level contaminant organism. Version 2.0 of the FASTQsim algorithm probabilistically simulated read counts and error distributions based on a provided model. Due to the probabilistic nature of the algorithm, coverage levels deviated slightly from the specified 60,000 reads, with the largest deviation observed for the *E. faecalis* organism (52,290 reads). For the HMP Staggered dataset, coverage levels varied from 11.3x (217,512 reads) for *Actinomyces odontolyticus* to 0.001x (2 reads) for *Neisseria meningitidis*. The goal of the staggered dataset was to evaluate the ability of metagenomic algorithms to detect organisms present at very low concentrations, i.e. less than 5 reads.

The bacterial dataset included reads from the genear Yersinia, Coxiella, Brucella, Salmonella, and Escherichia. For each of the five genera, several representative species were selected (i.e., *Brucella abortus*, *Brucella melitensis*, *Brucella suis*). Next, several representative strains were selected for each species (i.e. *Brucella melitensis ATCC 23457*, *Brucella melitensis biovar abortus 2308*,*Brucella melitensis biovar 1 strain 16M*, and *Brucella melitensis M28*). Organisms were spiked into a FASTQ dataset with coverage levels ranging from 10x to 0.00002x (1 read).

For the Virus Even dataset, 10x coverage of each organism was simulated. For the Virus Staggered dataset, coverage varied from 100x for *Sudan ebolavirus* to 0.5x for the *Human coronavirus HKU1*.

4.3 Metagenomic algorithm execution

Six metagenomic algorithms were selected for execution on the evaluation datasets. These included:

- MetaScope – winner of the Defense Threat Reduction Agency's Grand Challenge(7) (version 2.0)
- MetaPhlAn(28) (version 1.7.8, `https://bitbucket.org/nsegata/MetaPhlAn/src/`),
- MetaCV(15) (version 2.3.0, `http://sourceforge.net/projects/metacv/files/`),
- MetaPhyler(14) (version 1.13, `http://MetaPhyler.cbcb.umd.edu/#download`),
- Kraken(36) (v0.10.5, `https://ccb.jhu.edu/software/kraken/`),
- LMAT(2)-(33) (v1.2.5, `http://sourceforge.net/projects/lmat/`).

All algorithms were executed on each of the evaluation datasets using a machine with 512 GB of RAM, 64 cores, 1 TB hard drive, running the Fedora 17 operating system. All algorithms were executed with the default set of databases described in their respective documentation, downloaded on March 1, 2015. The commands used to execute all algorithms are listed in Supplementary File 2. Algorithms were evaluated using 60 of the 64 available cores. Attempts were also made to install and run the SURPI (v1.0, `https://github.com/chiulab/surpi`)(21) and compressed BLAST (v0.9, `http://cast.csail.mit.edu/`)(10) algorithms, but these were unsuccessful.

4.4 Algorithm performance evaluation

Runtime in seconds, true positive genus and species calls, false positive genus and species calls, read mapping, and relative abundance results at the species level were computed for all algorithm results. Additionally, correct gene calls were calculated for the set of algorithms that provided gene calling results (MetaScope, MetaCV, LMAT). The Gene ID Conversion function in the DAVID Bioinformatics Database(11) was used to convert across gene representation formats utilized by the three algorithms. Genes were marked as true positives if they matched the gene id, official gene symbol, locus tag, protein id, or specific product name of the truth data.

5 Availability of supporting data

The FASTQsim toolkit can be downloaded from SourceForge: `http://sourceforge.net/projects/fastqsim/`

In silico evaluation datasets can be downloaded from the Sequence Read Archive: SRP062063

- SRR2146185 – Virus Staggered dataset
- SRR2146184 – Virus Even dataset
- SRR2146183 – Bacterial dataset
- SRR2146181 – HMP Staggered dataset
- SRR2146182 – HMP Even dataset

6 List of abbreviations

- GB - gigabyte
- RAM – random-access memory
- s – seconds
- TB – terabyte
- x – fold coverage

7 Acknowledgements

The authors would like to thank Nick Greenfield and Dr. Sam Minot for contributing the One Codex results and feedback on the in silico datasets and supplemental tables.

8 Competing interests

The authors declare that they have no competing interests.

9 Ethics Committee Approval

Ethics approval was not required for this study because all data was generated in silico using references available in GenBank, as indicated in Supplementary Tables 1-2.

10 Authors' contributions

AS implemented FASTQSim updates and generated in silico datasets. AS and NC benchmarked algorithm performance on evaluation datasets. AS and DR wrote the manuscript. DR conceived of the study. All authors read and approved the final manuscript.

References

[1]Alkan C, Sajjadian S, Eichler E (2010) Limitations of next-generation genome sequence assembly, *Nature Methods* **8(1)**, 61-65.

[2]Ames S, Allen JE, Hysom DA, Lloyd GS, Gokhale MB (2014) Design and Optimization of a Metagenomics Analysis Workflow for NVRAM, *In: Parallel & Distributed Processing Symposium Workshops (IPDPSW)*, IEEE International: 19-23 May 2014 2014, 556-565.

[3]Ames SK, Hysom DA, Gardner SN, Lloyd GS, Gokhale MB, Allen JE (2013) Scalable metagenomic taxonomy classification using a reference genome database, *Bioinformatics*, **29(18)**, 2253-2260.

[4]Benson D, Cavanaugh M, Clark K, Karsch-Mizrachi I, Lipman D, Ostell J, Sayers E (2013) GenBank, *Nucleic Acids Research*, **41**, D36-42. DOI: 10.1093/nar/gks1195.

[5]Bonilla-Rosso G: Lessons learned from simulated metagenomic datasets. Encyclopedia of Metagenomics 2014:1-8.

[6]Bork P, Bairoch A (1996) Go hunting in sequence databases but watch out for the traps, *Trends Genet*, **12**,425-427.

[7]Buchfink B, Xie C, Huson DH: MetaScope. Fast and accurate identification of microbes in metagenomic sequencing data (2015) *arXiv.org* submitted.

[8]Chen Y, Lin C, Wang C, Wu H, Hwant P (2007) An optimized procedure greatly improves EST vector contamination removal, *BMC Genomics*, **8(416)**, DOI:10.1186/1471-2164-8-416.

[9]Cock P, Antao T, Chang J, Chapman B, Cox C, Dalke A, Friedberg I, Hamelryck T, Kauff F, Wilczynski B et al (2009) Biopython: freely available Python tools for computational molecular biology and bioinformatics, *Bioinformatics*, **25(11)**, 1422-1423, DOI:10.1093/bioinformatics/btp163.

[10]Daniels N, Gallant A, Peng J, Cowen L, Baym M, Berger B (2013) Compressive genomics for protein databases, *Bioinformatics*, **29(13)**, i283-i290.

[11]Huang D, Sherman B, Lempicki (2009) R: Systematic and integrative analysis of large gene lists using DAVID Bioinformatics Resources, *Nature Protocols*, **4(1)**, 44-57.

[12]Huang W, Li L, Myers J, Marth G (2011) ART: a next-generation sequencing read simulator, *Bioinformatics*, **28(4)**, 593-594. DOI: 10.1093/bioinformatics/btr708.

[13]Ilie L, Ilie S (2007) Multiple spaced seeds for homology search. *Bioinformatics*, **23(22)**,2969-2977.

[14]Liu B, Gibbons T, Ghodsi M, Trengen T, Pop M (2010) Accurate and fast estimation of taxonomic profiles from metagenomic shotgun sequences, *BMC Genomics*, **12(suppl 2)**.

[15]Liu J, Wang H, Yang H, Zhang Y, Wang J, Zhao F, Qi J (2013) Composition-based classification of short metagenomic sequences elucidates the landscapes of taxonomic and functional enrichment of microorganisms, *Nucleic Acids Research*, **41(1)**, DOI: 10.1093/nar/gks828.

[16]Martin HG, Ivanova N, Kunin V, Warnecke F, Barry KW, McHardy AC, Yeates C, He S, Salamov AA, Szeto E et al (2006) Metagenomic analysis of two enhanced biological phosphorus removal (EBPR) sludge communities, *Nat Biotech*,**24(10)**, 1263-1269.

[17]Mavromatis K, Ivanova N, Barry K, Shapiro H, Goltsman E, McHardy A, Rigoutsos I, Salamov A, Korzeniewski F, Land M et al(2007) Use of simulated data sets to evaluate the fidelity of metagenomic processing methods, *Nature Methods*, **4**, 495-500. DOI:10.1038/nmeth1043.

[18]Mende D, Waller A, Sunagawa S, Jarvelin A, Chan M, Arumugam M, Raes J, Bork P (2012) Assessment of metagenomic assembly using simulated next generation sequencing data. *PLOS one*, DOI: 10.1371/journal.pone.0031386.

[19]Minot S, Krumm N, Greenfield N.(2015) One Codex: A sensitive and accurate data platform for genomic microbial identification. *bioRxiv* DOI: 10.1101/027607.

[20]Morgan JL, Darling AE, Eisen JA (2010) Metagenomic Sequencing of an In Vitro-Simulated Microbial Community, *PLoS ONE*,**5(4):e10209**, DOI: 10.1371/journal.pone.0010209.

[21]Naccache S, Federman S, Veeraraghavan N, Zaharia M, Lee D, Samayoa E, Bouquet J, Greninger A, Luk K, Enge B et al (2014) A cloud-compatible bioinformatics pipeline for ultrarapid pathogen identification from next-generation sequencing of clinical samples, *Genome Research*, **24(7)**, 1180-1192. DOI: 10.1101/gr.171934.113.

[22]Ndao M (2009) Diagnosis of parasitic diseases: old and new approaches, *Interdisciplinary Perspectives on Infectious Diseases*.

[23]NIH HMP Working Group: The NIH Human Microbiome Project. Genome Research 2009, 19(12):2317-2323.

[24]Ondov B, Bergman N, Phillippy A (2011) Interactive metagenomic visualization in a Web browser, *BMC Bioinformatics*, **12(385)**, DOI:10.1186/1471-2105-12-385.

[25]Pignatelli M, Moya A (2011) Evaluating the fidelity of de novo short read metagenomic assembly using simulated data, *PLOS One*, DOI:10.1371/journal.pone.0031386.

[26]Quality control in databanks for molecular biology (2000) *Bioessays*, **22(11)**,1024-1034. DOI:10.1002/1521-1878(200011) 22:11<1024:AID-BIES9>3.0.CO;2-W.

[27]Sarkonen N (2007) Oral Actinomyces Species in Health and Disease: Identification, Occurence and Importance of Early Colonization, *National Public Health Institute*.

[28]Segata N, Waldron L, Ballarini A, Narasimhan V, Jousson O, Huttenhower C (2012) Metagenomic microbial community profiling using unique clade-specific marker genes, *Nat Meth*, **9(8)**,811-814.

[29]Seluja G, Farmer A, McLeod M, Harger C, Schad P (1999) Establishing a method of vector contamination identification in database sequenes, *Bioinformatics*, **15(2)**, 106-110.

[30]Shcherbina A (2014) FASTQSim: platform-independent data characterization and in silico read generation for NGS datasets, *BMC Research Notes*, **7(1)**, 533.

[31]Smith T, Porter S (2014) Development and role of the human reference sequence in personal genomics, *Wiley Online Library*, DOI: 10.1002/9780470015902.a0025334.

[32]Tyson GW, Chapman J, Hugenholtz P, Allen EE, Ram RJ, Richardson PM, Solovyev VV, Rubin EM, Rokhsar DS, Banfield (2004) JF: Community structure and metabolism through reconstruction of microbial genomes from the environment *Nature*,**428(6978)**,37-43.

[33]Van Essen B, Hsieh H, Ames S, Gokhale M (2012) DI-MMAP: A High Performance Memory-Map Runtime for Data-Intensive Applications. In: High Performance Computing, Networking, Storage and Analysis (SCC), 731-735.

[34]Venter JC, Remington K, Heidelberg JF, Halpern AL, Rusch D, Eisen JA, Wu D, Paulsen I, Nelson KE, Nelson W et al (2004) Environmental Genome Shotgun Sequencing of the Sargasso Sea, *Science*, **304(5667)**,66-74.

[35]Vijayakumar R, Sandle T, Manoharan C (2012) Review of fungal contamination in pharmaceutical products and phenotypic identification of contaminants by conventional methods, *European Journal of Parenteral and Pharmaceutical Sciences*, **17(1)**, 4-19.

[36]Wood DE, Salzberg SL (2014) Kraken: ultrafast metagenomic sequence classification using exact alignments, *Genome Biology* **15(R46)**, DOI:10.1186/gb-2014-15-3-r46.