AMERICAN SOCIETY FOR MICROBIOLOGY | mSystems

Check for updates

| Bacteriophages | Research Article

Benchmarking informatics approaches for virus discovery: caution is needed when combining *in silico* identification methods

Bridget Hegarty,[1] James Riddell V,[2] Eric Bastien,[3] Kathryn Langenfeld,[4] Morgan Lindback,[3] Jaspreet S. Saini,[5] Anthony Wing,[3] Jessica Zhang,[6] Melissa Duhaime[3]

AUTHOR AFFILIATIONS See affiliation list on p. 15.

ABSTRACT Understanding the ecological impacts of viruses on natural and engineered ecosystems relies on the accurate identification of viral sequences from community sequencing data. To maximize viral recovery from metagenomes, researchers frequently combine viral identification tools. However, the effectiveness of this strategy is unknown. Here, we benchmarked combinations of six widely used informatics tools for viral identification and analysis (VirSorter, VirSorter2, VIBRANT, DeepVirFinder, CheckV, and Kaiju), called "rulesets." Rulesets were tested against mock metagenomes composed of taxonomically diverse sequence types and diverse aquatic metagenomes to assess the effects of the degree of viral enrichment and habitat on tool performance. We found that six rulesets achieved equivalent accuracy [Matthews Correlation Coefficient (MCC) = 0.77, $P_{adj} \geq 0.05$]. Each contained VirSorter2, and five used our "tuning removal" rule designed to remove non-viral contamination. While DeepVirFinder, VIBRANT, and VirSorter were each found once in these high-accuracy rulesets, they were not found in combination with each other: combining tools does not lead to optimal performance. Our validation suggests that the MCC plateau at 0.77 is partly caused by inaccurate labeling within reference sequence databases. In aquatic metagenomes, our highest MCC ruleset identified more viral sequences in virus-enriched (44%–46%) than in cellular metagenomes (7%–19%). While improved algorithms may lead to more accurate viral identification tools, this should be done in tandem with careful curation of sequence databases. We recommend using the VirSorter2 ruleset and our empirically derived tuning removal rule. Our analysis provides insight into methods for *in silico* viral identification and will enable more robust viral identification from metagenomic data sets.

IMPORTANCE The identification of viruses from environmental metagenomes using informatics tools has offered critical insights in microbial ecology. However, it remains difficult for researchers to know which tools optimize viral recovery for their specific study. In an attempt to recover more viruses, studies are increasingly combining the outputs from multiple tools without validating this approach. After benchmarking combinations of six viral identification tools against mock metagenomes and environmental samples, we found that these tools should only be combined cautiously. Two to four tool combinations maximized viral recovery and minimized non-viral contamination compared with either the single-tool or the five- to six-tool ones. By providing a rigorous overview of the behavior of *in silico* viral identification strategies and a pipeline to replicate our process, our findings guide the use of existing viral identification tools and offer a blueprint for feature engineering of new tools that will lead to higher-confidence viral discovery in microbiome studies.

KEYWORDS bacteriophages, viral discovery, microbial ecology, metagenomics

Editor Ileana M. Cristea, Princeton University, USA

Address correspondence to Bridget Hegarty, beh53@case.edu, or Melissa Duhaime, duhaimem@umich.edu.

Bridget Hegarty and James Riddell V contributed equally to this article. Author order was determined based on increasing seniority.

The authors declare no conflict of interest.

See the funding table on p. 15.

Received 26 October 2023
Accepted 24 January 2024
Published 20 February 2024

Viruses are an essential component of microbial ecosystems: they influence nutrient cycling and microbial community dynamics (1), account for 20%–40% of microbial mortality per day (2), reprogram their hosts' metabolism (3, 4), and horizontally transfer genes between host populations (5, 6). The primary approach used to discover and describe viral diversity is culture-independent metagenomic sequencing. However, viral sequences remain challenging to differentiate from non-viral ones because viruses have no universal marker gene (7), high mutation rates (8, 9), and relatively small reference databases relative to the magnitude of their diversity (10). Additionally, current environmental sample collection and sequencing methods recover predominantly short contigs. Short sequences are challenging to classify correctly because they often do not contain enough information (e.g., too few genes) to leverage our knowledge of what makes viral sequences distinct (11, 12).

The challenge of identifying viral sequences in metagenomic data sets has driven the development of many viral identification tools over the past decade that aim to differentiate viral sequences from non-viral sequences (13). Tools differ in the types of viruses they identify, what sequence lengths they are optimized for, and the training data and algorithms underlying them. To be confidently applied to environmental data, viral identification tools must be trained on sequences representative of the microbiota being studied to ensure the tool has seen enough of the sequence space to correctly classify viral sequences. Sequence types commonly found in environmental metagenomes include bacteria, viruses, plasmids, archaea, protists, and fungi. Some tools, such as VirSorter2 (12) and VIBRANT (14), include these diverse sequence types, as well as representative diversity within each sequence type, expanding the classification accuracy of each tool. Other tools like DeepVirFinder and VirSorter do not include as diverse sequences: DeepVirFinder does not include non-prokaryotic references and VirSorter is only built on bacterial and archaeal virus references. Further, the performance of viral identification tools depends on the interaction between the tool algorithm and the sample type. In a comparison of the viruses recovered by different viral identification tools across 13 environmental samples, differences were found in the number of sequences called viral between environments (14).

While many viral identifications tools have comparable accuracy, their underlying algorithms differ and may capture different sets of viruses from the same sample. With so many tools available, it can be difficult to choose the most appropriate tool for a given study. Rather than choose one tool, a number of studies have combined the outputs of multiple tools to classify viral sequences to capture a greater portion of the viral signal (15–21). This approach assumes that combining multiple tools will improve the overall accuracy by discovering more viruses without greatly increasing non-viral contamination (non-viral sequences called viral by the approach), but this assumption has not been rigorously evaluated. In particular, it remains unknown whether or not these multi-tool strategies significantly increase contamination (e.g., by each tool returning non-overlapping false positive viral sequences).

Here, we benchmarked whether multi-tool approaches can distill a more complete and accurate set of viral sequences. From our analysis, we recommend pipelines specific to short and long-read sequences, as well as cellular metagenomes and virus-enriched samples. By returning more viral sequences with less non-viral contamination, these pipelines will enable new and more accurate insights into the ways viruses impact microbial ecosystem functions, with far-reaching implications for human and environmental health.

MATERIALS AND METHODS

Creation of sequence testing set

To create a testing set for benchmarking multi-tool pipelines, we downloaded viral, bacterial, fungal, plasmid, protist, and archaeal genome sequences from the NCBI reference sequence database (RefSeq), as well as a unique set of non-RefSeq virus

genomes compiled by the VirSorter2 tool developers, herein "VirSorter2 database" (22) (Fig. 1). These non-RefSeq virus genomes represent a comprehensive validated set of viruses. We originally created nine testing sets by randomly sampling sequences with replacement from these sources to create data sets that mimicked metagenomic environmental data. As the variability between testing sets plateaued at five data sets (Fig. S1), five data sets were used for subsequent analyses. The testing sets were designed to contain approximately 68% bacteria, 10% archaea, 10% virus (not proviruses), 5% plasmid, 5% protist, and 2% fungi sequences, totaling ~8k sequences (Fig. 1B). The proportion of sequences was chosen to be representative of cellular-enriched

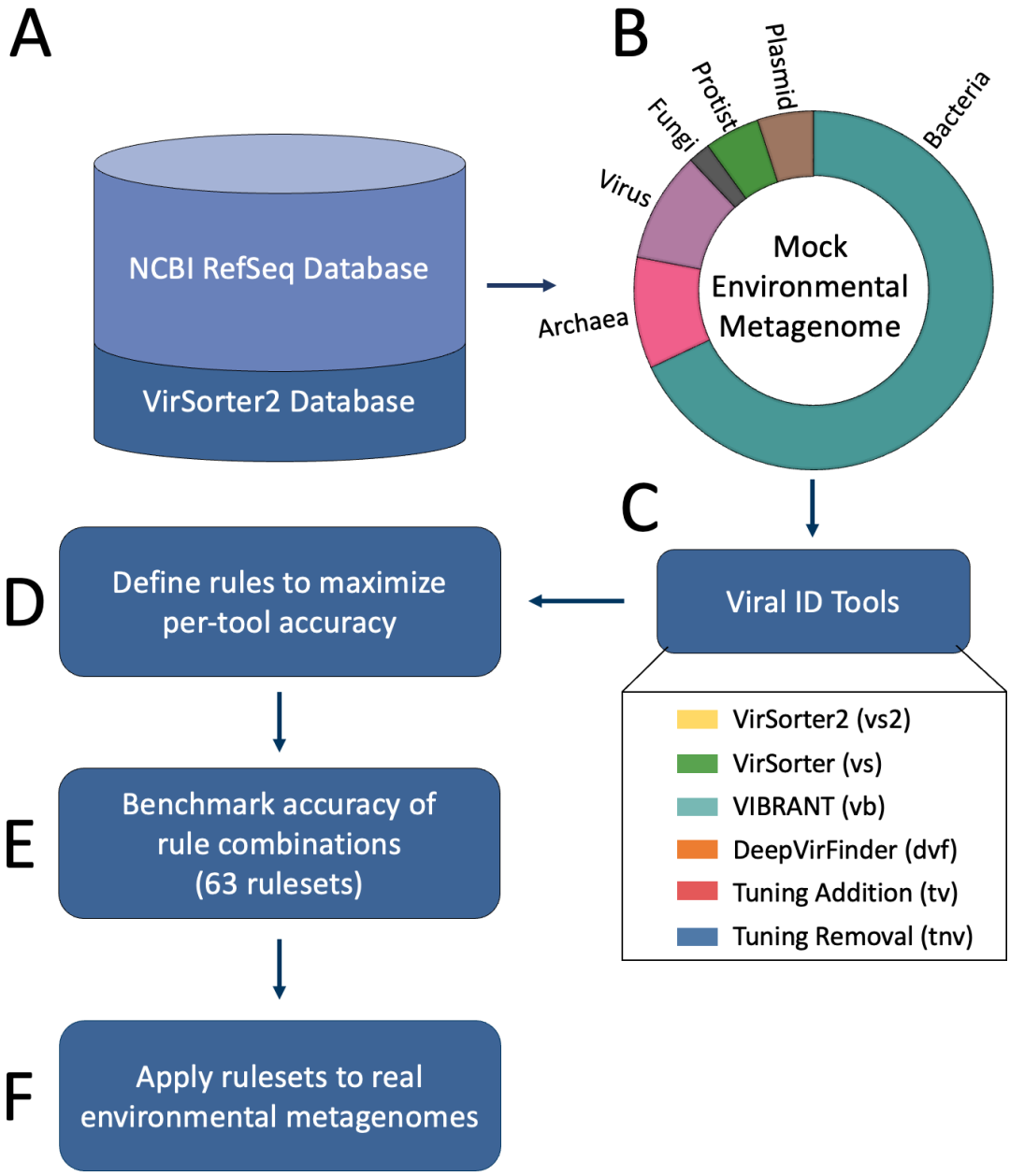

FIG 1 Overview of study workflow. (A) A set of sequences > 3 kb were randomly downloaded from NCBI and a curated Non-RefSeq viral genomes database ("VirSorter2 database") to (B) generate five mock environmental metagenomes, where the donut chart represents the proportion of each sequence type in each mock metagenome. (C) Mock metagenomes were run through six viral identification tools, (D) where score cutoffs were defined based on each tool's outputs to maximize their accuracy. (E) Accuracy was then assessed for each tool combination to guide the development of defined "rulesets." (F) Rulesets were then used to classify sequences from six real-world aquatic metagenomes: three cell-enriched metagenomes and three virus-enriched metagenomes.

metagenomic data, which are dominated by bacterial sequences and reportedly contain ~10% viral sequences (15, 23). The non-viral portion was randomly sampled from 5.3M bacteria, 55.1k archaea, 6.6k plasmid, 69.6k fungi, and 216.5k protist sequences from the NCBI database (accessed November 2019 for bacteria and archaea and April 2022 for others). The viral portion of the testing set was generated by random sampling from 13.8k viral sequences from the NCBI database and 370.153k viral sequences from the VirSorter2 database. As DeepVirFinder requires sequences to be less than 2,100 kb, a custom python script was written to trim the testing set sequences to meet this length cutoff. We did not expect trimming to impact results since the largest phage genome reported is smaller than this length cutoff (735 kb) (24). Further, only sequences longer than 3 kb were used because it has been previously shown that tool accuracy significantly decreases below 3 kb (12).

Selection of viral identification tools

Twenty-seven viral identification tools (12, 14, 25, 25–49) were found through literature search and assessed to determine their suitability for inclusion in this study (Table S1). Tools were included if they met the following criteria: (i) the tool identifies viruses that infect prokaryotes (i.e., bacteria and archaea), (ii) the tool is suitable for application to multiple environments (e.g., not only the human gut), (iii) the tool is designed to target viral sequences of lengths greater than 3 kb, (iv) the tool can classify millions of sequences within a few days on high-performance computing clusters (i.e., not only available on a web server), (v) developers actively respond to user issues, (vi) the tool performs well in previous comparative studies of viral identification tools (12, 14, 50), (vii) the tool is not specific to prophages, and (viii) the tool was published before June 2022.

Four of the 29 viral identification tools met the above criteria: DeepVirFinder (27), VIBRANT (14), VirSorter (46), and VirSorter2 (12). While not designed strictly as viral identification tools, Kaiju (51) and CheckV (26) were used to tune the viral predictions in our test sets. All six are referred to as "viral identification tools" or simply "tools" in this manuscript (Table 1; Fig. 1C).

Design of viral identification rules

Viral identification tools generate scores that indicate how confident they are that a given sequence is of viral origin, but users are often faced with the dilemma of setting their own score cutoff to decide which sequences to call "viral." To aid in the process of choosing rules and cutoffs for predicting viral sequences, we designed six rules (Fig. 1D and 2). Each rule includes at least two *subrules* that use outputs from the six selected tools. These subrules were designed through two processes:

1. *Evaluation of existing recommendations for tool cutoffs and application*: the recommended cutoffs for distinguishing viral and non-viral sequences in each tool's protocol were used as an initial set of rules (12, 15, 16, 52).
2. *Curation and evaluation of biological features:* some viral identification tools generate information describing biological features for each sequence, e.g., VirSorter2 reports the number of viral hallmark genes identified, CheckV reports the completeness of a sequence and relative percentage of viral versus non-viral genes, and VIBRANT identifies virus orthologous genes (VOGs). These biological features were used to create classification criteria (Fig. 2) to distinguish viral and non-viral sequences.

The developers' recommendations for calling a sequence viral served as a baseline for assigning a given sequence a "*viral score*" that captured the relative likelihood that a given sequence was viral. The cutoffs were then adjusted to maximize the number of true viral sequences being classified as viral (true positives) and minimize the number of non-viral sequences being classified as viral (false positives) in the mock environmental microbial communities. We first defined four single-tool rules (Fig. 2A through D) derived

TABLE 1 Overview of viral identification tools selected for inclusion in this study

Tool name (version)	Tool description	Algorithmic approach	Why we included the tool
CheckV (26)	CheckV is an automated pipeline that identifies closed viral genomes, estimates the completeness of genome fragments, and removes host regions from proviruses.	HMM virus and host marker genes, virus-host boundary prediction, and AAI-based estimation of genome completeness	Not a viral identification tool but provides benchmarking information for refining predictions from other tools
DeepVirFinder (27)	DeepVirFinder uses a multi-layered deep learning algorithm trained on a positive set of viral sequences from viral RefSeq data and a negative set of prokaryotic ones.	K-mer-based deep learning convolutional neural network	Recent and increasingly commonly used tool based on a neural network
VIBRANT (14)	VIBRANT is a hybrid tool that uses both machine learning and protein similarity to classify viruses as either high, medium, low quality, or non-viral.	Neural network of protein annotations of HMMs	Recent and commonly used tool and provides gene annotation information
VirSorter (46)	VirSorter uses probabilistic models with reference and non-reference dependencies, as well as detecting hallmark viral genes.	Probabilistic modeling using HMMs	Commonly used and high-quality predictions
VirSorter2 (12)	VirSorter2 uses a neural network classifier built on top of the existing VirSorter infrastructure of reference-based viral identification.	A multi-classifier combining a Random Forest model and expert knowledge of viral features	Recent and increasingly commonly used tool that is more inclusive of viral diversity than most other tools
Kaiju (51)	Kaiju is a taxonomic classifier that compares metagenomic sequences to NCBI reference databases at the protein level and assigns a near or exact taxon match if one is found.	A taxonomic classifier that uses protein-level classifications to assign reads to taxon from NCBI databases	Not a viral identification tool but extremely fast method of taxonomic identification based on NCBI releases

from four viral identification tools (i.e., VIBRANT, VirSorter, VirSorter2, and DeepVirFinder). Next, we defined two sets of tuning rules derived from outputs of different tools that indicated a strongly viral or non-viral signal (Fig. 2E and F). These included the following: (i) a "tuning removal" rule that decreases the *viral score* based on distinctly non-viral sequence features and (ii) a "tuning addition" rule that increases the *viral score* based on distinctly viral sequence features.

Each putative viral sequence is assigned a numerical value by each rule; these are combined to give a value that comprises the sequence's *viral score* (Fig. 2G). Sequences with a final *viral score* ≥ 1 were considered viral, and scores < 1 are non-viral (Fig. 2G).

Evaluation of viral identification rulesets

Ultimately, 63 combinations (rulesets) of these six rules were evaluated by comparing the viral score of each sequence to the classification assigned by the database (Fig. 1E). From these values, precision (the number of true viruses in our test set called viral divided by all contigs called viral), recall (the number of true viruses in our test set called viral divided by all true viruses), and MCC (considers the relative proportion of false positives, false negatives, true positives, and true negatives) (53) were calculated.

Application of rulesets to environmental metagenomes

All rulesets were used to identify viruses from five previously published environmental data sets representing different aquatic environments and size fractions (Fig. 1F; Table S2). Three environments (drinking water, global ocean water, and eutrophic lake water) contained metagenomic assemblies (>2 µm), and three environments (wastewater, eutrophic lake water, and oligotrophic lake water) contained virome assemblies (<0.2 and <0.45 µm), meaning samples were enriched for viruses by filtering through a small pore to remove most cellular organisms before DNA extraction.

The default tool settings were used except for the oligotrophic lake and wastewater virus-enriched samples, where the "-virome" flag was used for VIBRANT and VirSorter to

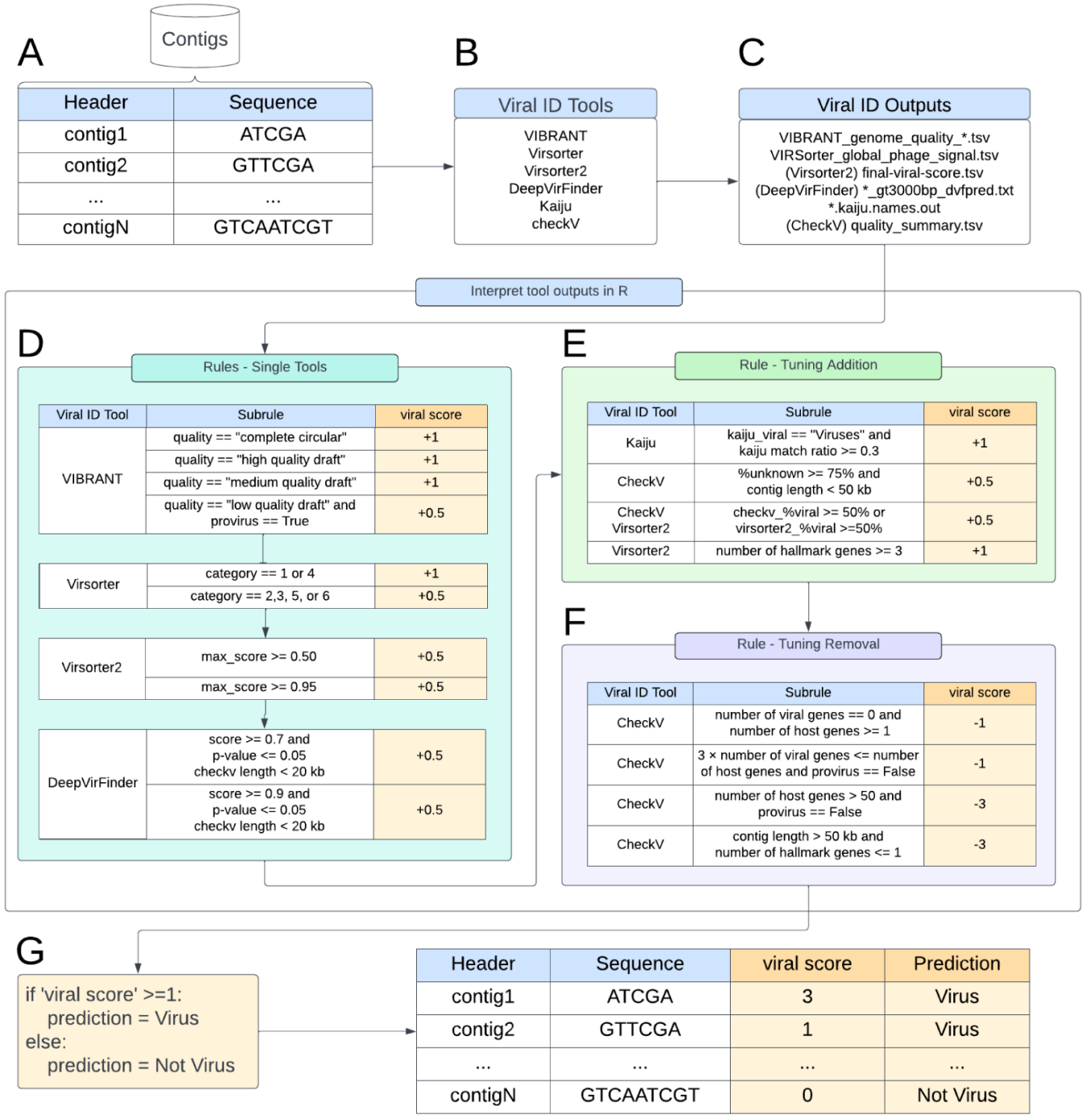

FIG 2 Diagram of approach details. (A) Sequences are first processed by each (B) viral identification tool. Next, (C) the tool outputs are programmatically post-processed to generate a *viral score* based on both (D) single-tool rules and the data-driven creation of (E) tuning addition and (F) tuning removal processes. (G) This combined post-processing generates a *viral score* that indicates whether each sequence input is predicted as "Virus" or "Not Virus." Subrules are scored based on the confidence of the prediction: low confidence = ±0.5, confident = ±1, and highly confident not viral = −3.

reduce sensitivity, as recommended by the developers of those tools given that a greater fraction of total sequences are expected to be viral in virus-enriched samples. The "-virome" flag was not used for the eutrophic lake water virome assemblies due to all eutrophic lake water assemblies being processed together, but the eutrophic lake water virome still was most similar to the other virus-enriched samples (Fig. 7). All tools were run with a 3-kb cutoff to remove small sequences.

RESULTS AND DISCUSSION

Viral identification tools use algorithms based on the knowledge of viral sequences and machine learning to separate viral and non-viral sequences. In this study, we test the hypothesis that combining viral identification tools with different underlying algorithms will improve accuracy. The performance of 63 combinations (rulesets) of the six rules was evaluated using five mock metagenomes of known composition (Fig. 3B; Fig. S1). The six rules are as follows: four single-tool rules derived from four viral identification tools (i.e., VIBRANT, VirSorter, VirSorter2, and DeepVirFinder; Fig. 2A through D) and two additional tuning rules: tuning removal (which removes predictions using Kaiju, CheckV, VirSorter2, VirSorter, and VIBRANT outputs) and tuning addition (which adds predictions using Kaiju, CheckV, and VirSorter2 outputs; Fig. 2E and F). In this section, we compare the performance of our rulesets, elaborate on their strengths and weaknesses, and provide recommendations for use.

More tools are better… to a point

Across the 63 rulesets, MCC, our metric for overall performance ("accuracy," herein), ranges from 0.05 (DeepVirFinder) to 0.77 (VirSorter2 + Tuning Removal). With the exception of VirSorter2 (MCC = 0.75), single-rule rulesets (i.e., viral identification tools run on their own) either missed most of the viruses in the benchmarking data set

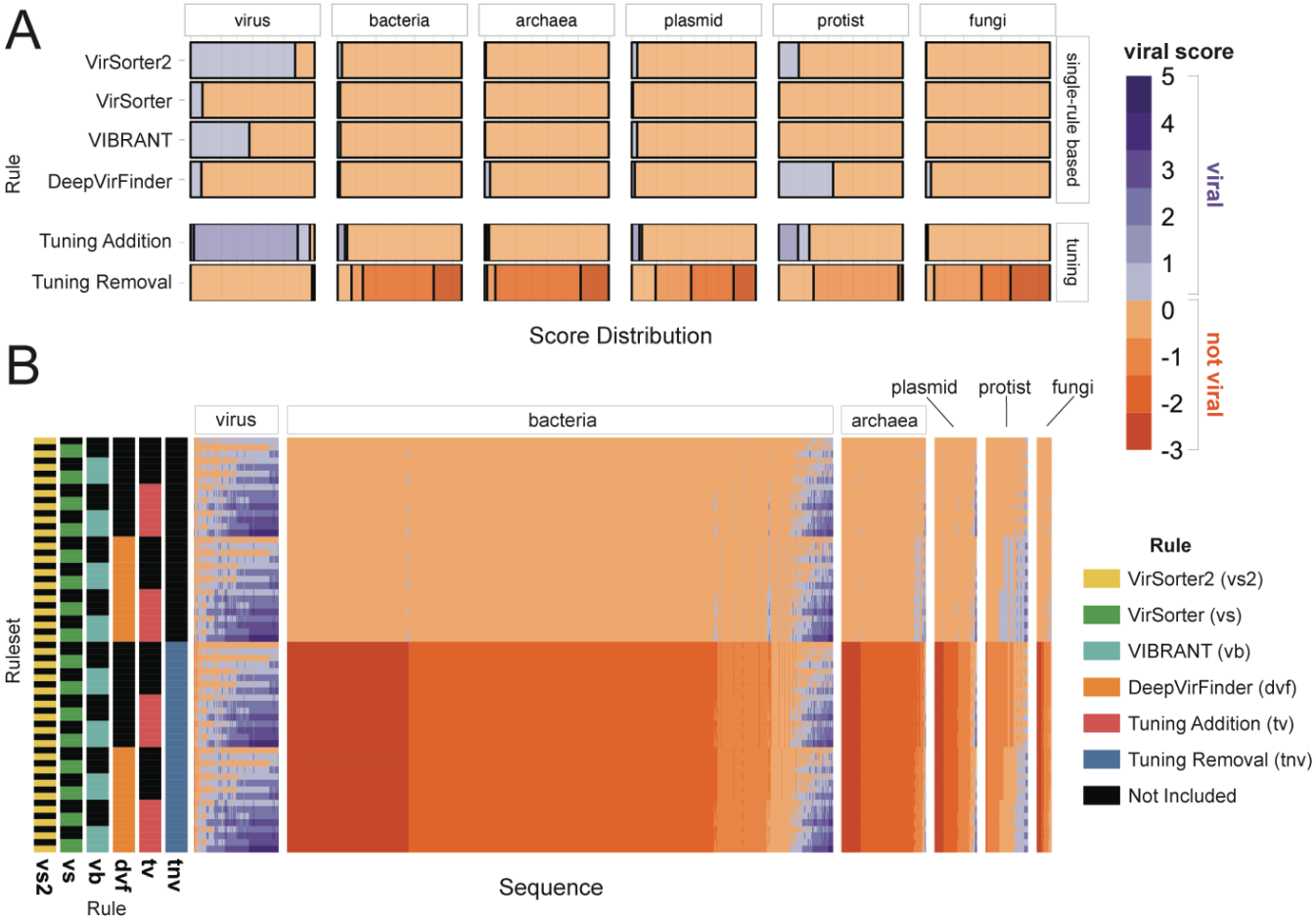

FIG 3 Comparison of different rulesets. (A) Distribution of *viral scores* assigned to mock metagenome sequences for our six rules: four single-tool and two tuning rules. (B) Viral scores of all sequences across six mock metagenomes classified by each ruleset. Ruleset rows are colored based on whether or not each rule was used to attain the *viral score* result for a given sequence. In both A and B, *viral scores* ≥ 1 are classified as viral and <1 as not viral. All sequences are grouped by their assigned taxonomy.

or misclassified such a large number of non-viruses that the viral signal was heavily contaminated (Fig. 3A). Of the single-rule rulesets, VIBRANT performs second best (MCC of 0.55), followed by VirSorter (MCC of 0.16) and DeepVirFinder (MCC of 0.05). Previous studies have reported higher accuracy for these tools, but those studies used a testing set composed of 50% or more viruses compared with our 10% viral sequences and/or did not include taxonomically diverse sequences compared with the taxonomically diverse training data used here (14, 27, 46, 52). Using testing data sets representative of environmental metagenomic data sets is important for accurate ecological interpretations. We found that our MCCs increased when using a testing set with a more similar composition to other studies (50% viral and 50% non-viral; max MCC = 0.91); and, in their validation of DeepVirFinder, Ren et al. (27) demonstrated that the relative proportion of viral to non-viral sequences in a data set can have a strong effect on AUROC (area under receiver operating characteristic; a performance metric). Given the observed taxonomic distribution of environmental cellular metagenomic sequences (15, 23), users likely need to be more conservative (higher classification score cutoffs) in viral calling and assume lower accuracy than previous studies have reported.

Combining rules generally increased the average MCC (Fig. 4). This is driven by a statistically significant increase in recall for multi-rule rulesets compared with single-rule rulesets and for three or more rule rulesets compared with two-rule rulesets (Table S3). While the average precision is constant as rules are combined (Fig. 4A; Table S4), the precision of higher-precision rules (precision > 0.7, five right-most points; Fig. S2A; the VIBRANT and VirSorter2 rules) decreases as they are combined with lower-precision rules (precision < 0.5; the DeepVirFinder and VirSorter rules). Accuracy is maximized by the VirSorter2 and tuning removal ruleset and is not improved by adding more rules (Fig. 4B).

The VirSorter2-based and tuning removal rules are the most critical for accurate virus identification in our testing. When rulesets were ranked by increasing MCC, top-performing "high MCC" rulesets were identified as those that did not demonstrate a statistically significant decrease relative to the highest MCC ruleset ($P_{adj} > 0.05$; Fig. 4B). VirSorter2's rule ("vs2") was in all six of these high-MCC rulesets (Fig. 4B) and tuning removal ("tnv") was in five of them. For comparison, the other single-tool rules (i.e., VirSorter, DeepVirFinder, and VIBRANT) were each only in one of the high-MCC rulesets, where they always co-occurred with VirSorter2 and the tuning removal rule (Fig. 4B). None of the high-MCC rulesets have more than four rules. In the same way, "high precision" (three rulesets; Fig S2A) and "high recall" (four rulesets; Fig S2B) rulesets were defined.

There is a high degree of overlap in the viruses predicted by the different rulesets (Fig. 5). Of the comparisons between rulesets , 68% (Fig. 5, green and yellow cells) was more than 50% identical to each other and 30 rulesets were at least 90% identical to at least one other ruleset (Fig. 5, yellow cells), representing 4% of the total comparisons between rulesets. Rulesets with VirSorter, DeepVirFinder, and VIBRANT all have more sequences in common as the number of rules in the compared sets increases, but this trend is much less pronounced for VirSorter2 (Fig. S3). This suggests that our observed increase in recall through combining rules is being driven by a subset of rules that give more tool-specific viral predictions (i.e., VirSorter, DeepVirFinder, and VIBRANT), leading to the question "how confident should users be of their tool-specific predictions?"

One assumption made during *in silico* virus identification by previous studies is that sequences with low-confidence predictions by multiple tools are more likely to be viral (i.e., called viral by multiple + 0.5 subrules). For example, if one tool predicts a sequence as viral with low confidence, it may be disregarded. But if multiple tools each provide low-confidence predictions for a given sequence, many studies have presumed the sequence is more likely to be viral, thereby combining low-quality predictions to arrive at the set of predicted viruses (15, 16, 18). However, we found this was not a safe strategy as it did not increase the number of true positives . Rulesets using multiple low-quality prediction subrules in the single-tool rules did not significantly increase MCC ($P = 0.19$) or recall ($P = 0.18$) and, in fact, slightly decreased precision (0.54 vs 0.59, $P = 2.5*10^{-5}$) when

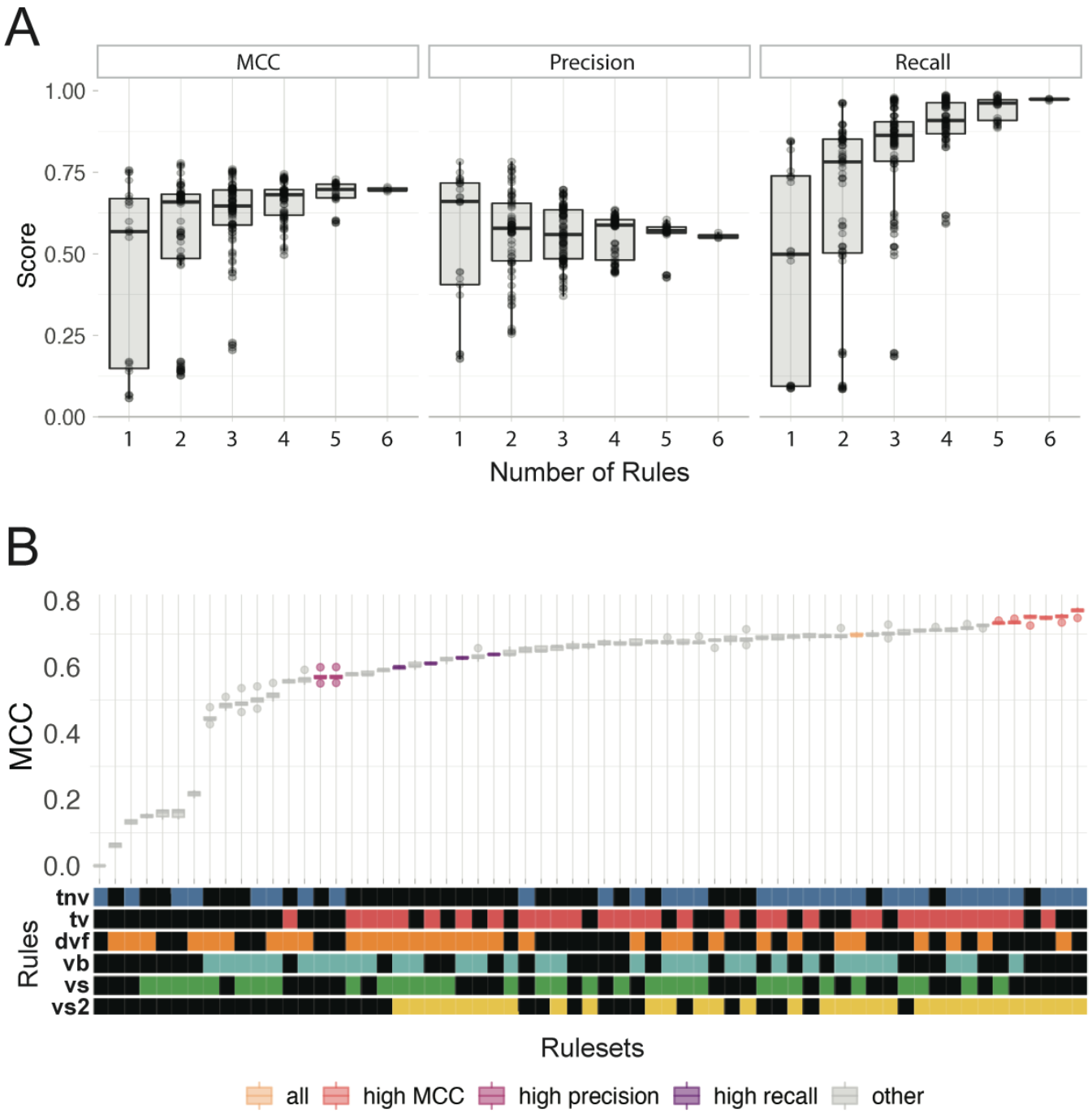

FIG 4 Performance of the 63 rulesets. (A) Box and whisker plots of the performance scores representing variation in MCC, precision, and recall of different rulesets based on the number of rules used for prediction. (B) Ruleset accuracy (MCC) ordered by increasing MCC and colored based on the ruleset's type according to statistically equivalent ($P_{adj} \geq 0.05$) rulesets. For A and B, the middle line represents the group mean; boxes above and below the middle line represent the top and bottom quartiles, respectively; whiskers above and below the boxes represent 1.5 times the interquartile range (roughly the 95% CI), outliers are represented by circles beyond the whiskers. The boxplots in A are overlaid with points that represent each testing set's MCC.

compared with rulesets that did not use low-quality prediction subrules (Fig. S4). This pattern is likely due to the rulesets being similarly uncertain about sequences that are unlikely to be viral, thus introducing a significant amount of non-viral contamination. The additive uncertainty did not create certainty. We recommend being cautious of sequences classified as viral by multiple low-quality predictions and manually inspecting them before viral assignment.

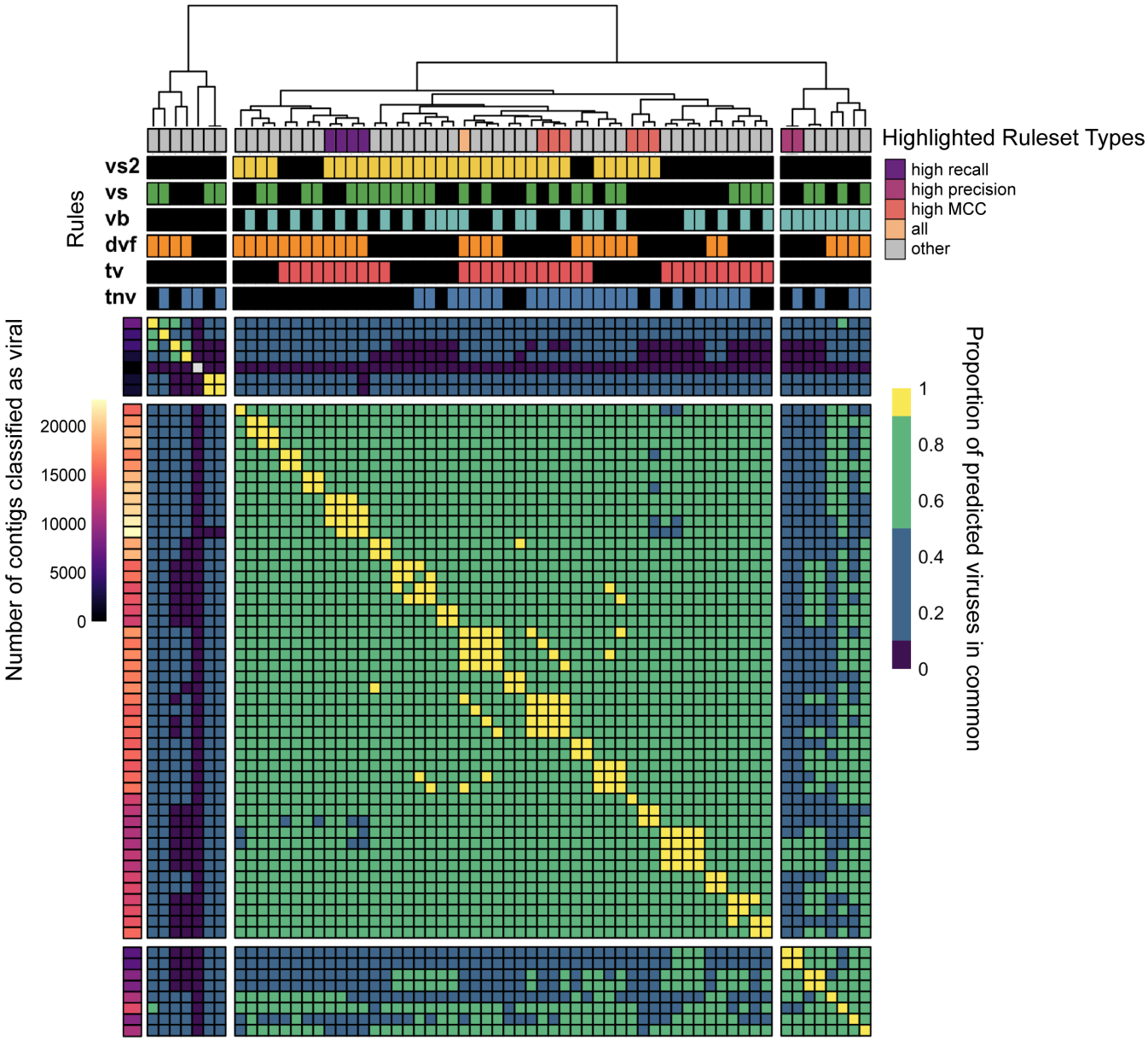

FIG 5 Proportion of viruses in common between rulesets. Heatmap values calculated by dividing the intersection (called viral by both rulesets) by the union (called viral by at least one) of the viruses found by both rulesets, which represents the proportion in common between the tools (scale bar on right: dark-purple: 0–0.1, blue: 0.1–0.5, green: 0.5–0.9, and yellow: 0.9–1). The bar to the left of the heatmap represents the total number of viruses identified by each tool combination (scalebar to its left). The bars above the heatmap indicate the tool(s) used in the rulesets, as well as the ruleset type.

As metagenomes frequently have a high proportion of short-sequence fragments and the correct identification of short fragments with only a few genes is particularly challenging (26), we tested the rulesets on 3–5-kb fragments from our original testing sets. Like the >3-kb testing sets, many rulesets performed similarly for the 3–5-kb fragments (Fig. S6). Unlike in the >3-kb testing sets where the VirSorter2 single-rule ruleset was in the high-MCC set (Fig. 4B), no single-rule ruleset was in the high-MCC rulesets for the short fragments (Fig. S5). Also in contrast with the >3-kb testing sets, the six-rule ruleset was identified as a high-MCC ruleset and DeepVirFinder is in 8 of the 12 high-MCC rulesets. In part due to our newly defined viral tuning rules, the accuracy of the viral predictions reported here for these short fragments is greater than previously published (12). This increase in accuracy suggests that data sets where short fragments

abound may particularly benefit from our tuning rules. Further, while researchers with only 3–5-kb fragments may consider using more tools, equally accurate predictions can still be achieved from the VirSorter2 and tuning removal ruleset.

Tuning rules increase confidence of viral predictions

To leverage expert knowledge of the differences between viral and non-viral sequences, we designed tuning addition and removal rules (Fig. 2E and F; see Fig. S6 for subrule performance). These rules were designed based on specific outputs from multiple tools that distinguish between viral and non-viral sequences, such as sequence length and the number of host genes. In general, tuning addition improved MCC and recall, while tuning removal improved MCC and precision (Fig. S7). Seven of the 10 highest-MCC rulesets have both tuning addition and removal rules (Fig. 4B), demonstrating the importance of the tuning rulesets for accurate classification. The tuning *removal* rule was able to identify 89% of the testing set's non-viral sequences and only (mis)identified 2% of testing set's viral sequences as non-viral (*viral score* < 0 when only the tuning removal rule was applied). The tuning *addition* rule accurately identified 74% of the viral sequences and only misidentified 4% of non-viral sequences in the testing set as viral (*viral score* ≥ 1 when only the tuning addition rule was applied). Overall, the tuning rules increased our prediction accuracy beyond that of the rulesets composed of the single-tool rules. As such, we demonstrated the value of automating the refinement of viral identification tool predictions, a task that, if done at all, is currently a laborious manual process.

Even with the tuning rules, we could not improve both precision and recall beyond 0.77 (Fig. S8). Building a more accurate classifier means overcoming barriers such as imperfect gene reference data sets, overlap between viral and host sequences, and underrepresentation of viral types. This is because to recover more viruses, it becomes necessary to rely more on genes of unknown origin. These may include non-viral genes, particularly of eukaryotes, which were not represented in the reference data sets of DeepVirFinder, VirSorter, or CheckV (Fig. S9 to S16). Further, many true viral features overlap with non-viral features (Fig. S9 to S16) due to our imperfect knowledge of what distinguishes viruses and non-viruses (and homologous sequences shared by both viruses and cellular organisms), leading non-viruses with virus-like features to be misclassified as viruses. This challenge is particularly acute when trying to accurately classify both short sequences (<5 kb) (12, 27) and viral types underrepresented in our testing data (e.g., the accuracy of the "high MCC" ruleset is the highest for dsDNAphages compared with other viral sequence types; Fig. S17).

Mislabeled sequences within databases hinder tool accuracy and validation

To improve upon the maximum MCC of 0.77, we looked for patterns in the types of sequences being misclassified that could aid future tool design. To our surprise, the "false positive" sequences labeled as bacteria by the NCBI database, but classified as viral by our high MCC ruleset, looked more "viral" than the viruses themselves. Specifically, the proportion of the sequence's genes identified as VOGs was higher in the misclassified bacteria than the known viruses ($P < 2.2*10^{-16}$; Fig. 6A). The plasmids misclassified as viral have a similar proportion of viral genes as the viruses ($P = 0.63$). For all three sequence types, the proportion of sequences represented by VOGs increases with the number of VOGs in that sequence (Fig. 6B). If these highly viral sequences are not actually viruses, viral identification tools can be improved by removing these genes from viral gene databases. On the contrary, if these sequences are actually viruses, viral identification tools can be improved by relabeling these sequences in sequence databases because tools rely on accurate database classification for training and testing.

Manual inspection of a subset of these misclassified sequences revealed them to be viral sequences (Fig. 6C; Fig. S18). These false positives represented two types of mislabeled sequences: (i) viruses (either extracellular virions or intracellular extrachromosomal

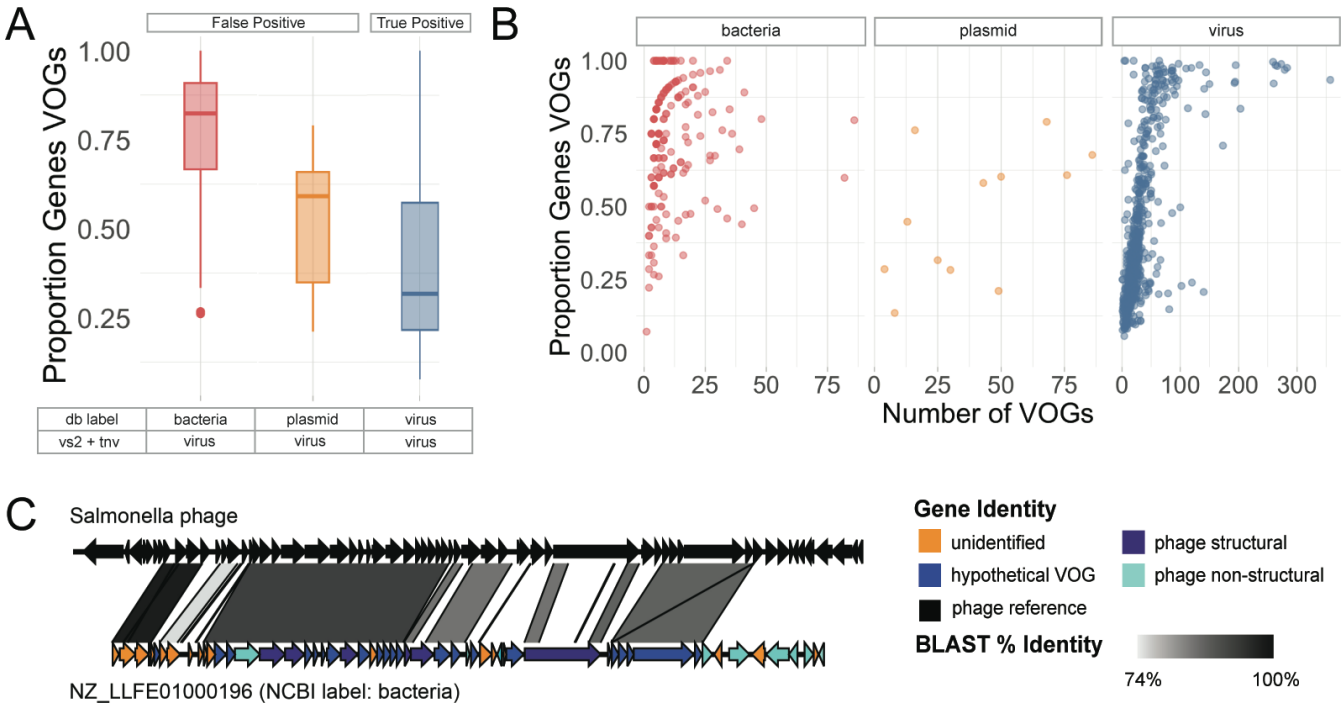

FIG 6 Mislabeled sequences. (A) Box and whisker plots of the proportion of genes on a sequence with a VOG annotation by VIBRANT broken down by sequence type (for the high-MCC rules). (B) Proportion of a sequence's genes with a VOG annotation versus the number of genes with a VOG annotation faceted by sequence type. Because VIBRANT is the only tool that provided VOG annotation, only sequences classified as viral by VIBRANT are included in panels A and B (which only included bacteria, viruses, and plasmids). (C) Sequence synteny plots indicating sequence similarity between a representative bacterial "false positive:" NZ_LLFE01000196 (NCBI label: bacteria) versus Salmonella phage SSU5. All genes of the testing set sequences are colored by their gene identity.

viral genomes) co-sequenced when a host isolate was sequenced (Fig. 6C; Fig. S18A) and (ii) prophages integrated into their hosts' genome (Fig. S18B). We also screened for ΦX contamination, as it is a known problem for database sequences using Illumina library preparation due to its use in Illumina libraries (54). Only 19 ΦX sequences were taxonomically identified by Kaiju and thus do not explain our high degree of false positives. These findings support the known problem of phage sequences not being removed before being deposited on NCBI (54) and lead to phages being misclassified as bacteria, archaea, plasmids, and cellular and satellite chromosomes. Mislabeled sequences in public databases make it difficult to produce accurate viral identification tools because developers rely on accurately labeled data sets to train and test their classifiers. We recommend that before uploading sequences to public databases where non-viral sequences are screened for viral contamination.

Sample preparation and viral identification tool choice affects viral sequence recovery

To compare our rulesets across environments, we evaluated the proportion of sequences classified as viral for five publicly available aquatic metagenomic assemblies (Table S2; Fig. S19). We found that viral sequence recovery varied greatly based on sample preparation (e.g., virus enriched or not) and viral identification tool(s) used (Fig. 7). The highest MCC ruleset identified a higher proportion of viral sequences in the virus-enriched samples (44%–46%) compared with the non-enriched metagenomes (7%–19%). The proportion of viruses recovered across rulesets for the environmental data sets mimics the behavior of the testing data: the "high recall" rulesets classified the most sequences as viral (with presumably the greatest non-viral contamination, given the results on our testing sets), while the "high precision" rulesets labeled the fewest sequences as viral (with presumably the least non-viral contamination, given the

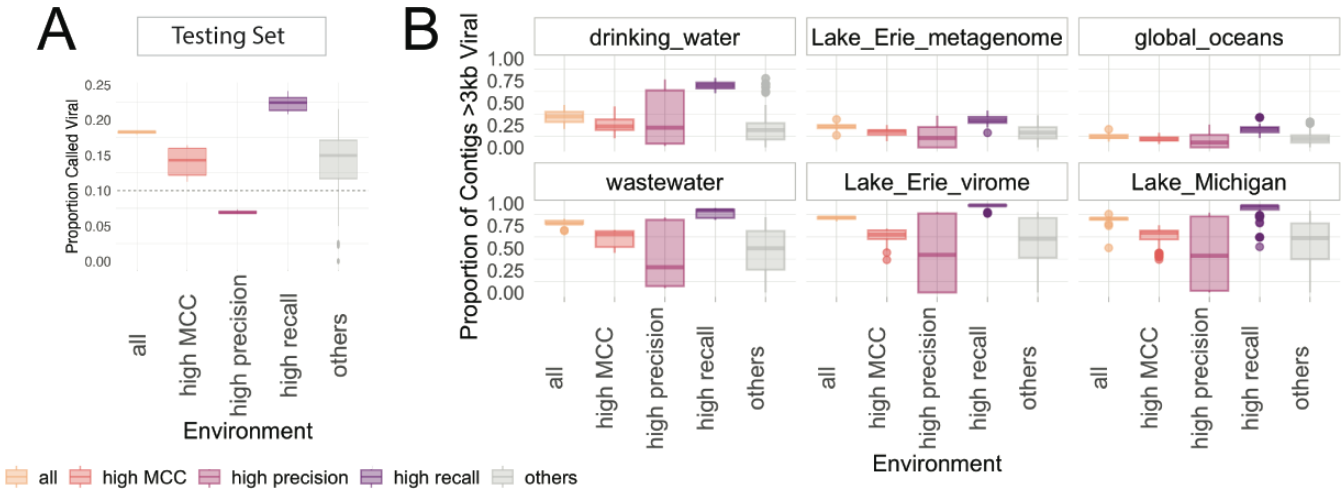

FIG 7 Proportion of viruses predicted by each tool combination across (A) our testing sets and (B) environmental data sets. Rulesets are grouped based on the accuracy type on the testing set shown in the highlighted rulesets in Fig. 5.

results on the testing sets) (Fig. 7; Table S2). Previous metagenomic studies of biomass collected on a 0.2-µm filter have found viral sequences to be less than 16% of the total metagenome (15, 23), which is in line with our results.

For the virus-enriched samples, nearly all sequences were called viral by the "high recall" rulesets (Fig. 7; Fig. S18). It is likely that some of these sequences are false positives because the tuning removal rule reduces the number of sequences called viral by 14.9% (1 SD = 2.1%) when comparing the "all" to the "highest recall" rulesets, and our benchmarking demonstrated that the tuning removal rule effectively removed the contaminating non-viral sequences without removing true viruses. Further, one of the few studies to report the proportion of viral sequences found that 30%–60% of the sequences with known taxonomy were similar to at least one viral sequence (55). Even if all unknown sequences were viruses in this study, the proportion of viruses would not exceed 92%, which is still lower than the proportions we found in two of our three virus-enriched samples and further suggests the importance of tuning removal even for viral-enriched metagenomes. We present this "high recall" example to caution readers against using the "high recall" rulesets on virus-enriched metagenomes. It may be tempting to assume virus enrichment removes nearly all non-viral sequence contamination, but even for virus-enriched fragments, we instead recommend using the tuning removal rule unless the number of sequences is small enough to be manually inspected.

Recommendations and future work

In silico prediction of viral sequences is a critical first step to any metagenomic study that aims to resolve viral ecology and virus-microbe interactions. As downstream analyses and conclusions are predicated on accurate viral prediction, it is paramount to choose the most suitable tool(s) [and know how to interpret its output(s)] among the rapidly increasing number of *in silico* viral prediction tools available. Through the above benchmarking, we demonstrated that specific two-rule rulesets provide the highest-precision viral identification with only minor sacrifices in recall, whereas the worse performance of combining all tools may lead to erroneous biological conclusions. For this reason, we urge caution when using recent automated pipelines for sequence identification that combine the output of multiple tools (17, 19–21).

Our recommendations based on this study vary depending on research question and experimental design Fig. 8. For a typical study investigating viral diversity and functional potential from a mixed metagenome, we recommend our "high MCC" ruleset (VirSorter2

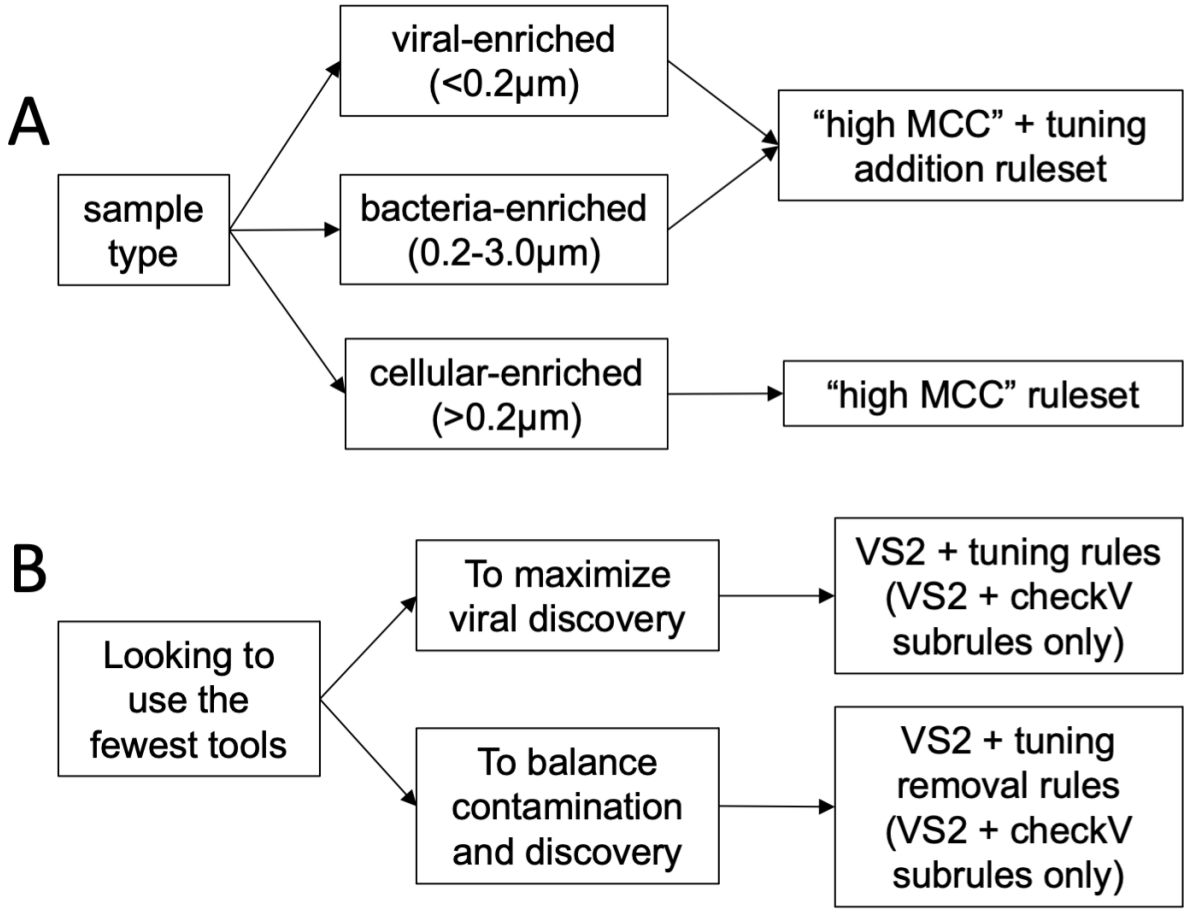

FIG 8 Recommendations. Flowcharts are based on (A) sample type and (B) study goals when looking to minimize the number of tools used.

with tuning removal). If the majority of eukaryotes were filtered out of the sample (e.g., <3 µm fraction was sequenced), the tuning additional rule may increase recall. For researchers seeking to minimize the number of tools they use, we recommend using VirSorter2 with our CheckV-based tuning subrules. VirSorter2 has a comparable MCC to multi-tool rules (Fig. 4), though its high recall comes at the cost of more false positives when used in isolation (Fig. S2). While VIBRANT's high precision with and without our tuning removal rule (Fig. 4; Fig. S2) and convenient information about the viruses (e.g., their metabolic potential) are attractive, we found that its recall was much worse than our other recommendations on the environmental data sets. In general, however, we do not recommend researchers to use any of the tools in isolation, based on the poor accuracy of the single-rule sets on the short sequences (3–5-kb testing set; Fig. S5).

One limitation of this work is that the available testing data included sequences that were part of the tools' original training and testing data: all tools included in this study were trained in part using NCBI sequences that overlap with our testing data, and some were trained using the VirSorter2 non-RefSeq genome set. These data sets also happen to harbor a substantial number of mislabeled sequences, making it difficult to assess the accuracy of the benchmarking results. As additional curated viral sets are published (56), other researchers can test our rules against the new data sets providing further information about the limitations and scope of our rules.

We focused on tools that were developed primarily for bacteriophage identification, as these are most commonly used by microbial ecologists and microbiome researchers. We did not evaluate tools that were specifically for prophages, human viral pathogens, eukaryotic viruses, or archaeal viruses more broadly (Table S1). Future integration of new

tools for plasmid and eukaryotic sequence identification (54, 57) is likely to improve viral identification tool pipelines.

Conclusions

With the rapid development of new viral identification tools, this paper offers a blueprint for intentional, data-driven validation of tool combinations. We found that the highest accuracy resulted from rulesets with four or fewer rules. For most applications, we recommend a combination of VirSorter2 and tuning rules based on features of viral and non-viral sequences and caution against simply combining viral identification tools expecting higher quality virus sets. By increasing the proportion of high-confidence viruses identified from mixed metagenomic data sets through intentional, data-driven combination of tools, this study enables more accurate ecological analyses by decreasing contamination of the viral signal, particularly from eukaryotic sequences.

ACKNOWLEDGMENTS

This work was supported by the Blue Sky Initiative of the University of Michigan College of Engineering (B.H. and M.B.D.), National Science Foundation award #2055455 (M.B.D., E.B., and M.L.), the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE-134012 (J.R.) and DGE-1256260 (M.L.), and the National Oceanic and Atmospheric Administration Great Lakes Omics program distributed through the UM Cooperative Institute for Great Lakes Research NA17OAR4320152 (M.B.D. and A.W.).

B.H. and J.R. contributed equally to this study. They wrote the main manuscript and prepared the figures and tables. B.H. and M.B.D. are co-corresponding authors and conceived the idea. M.B.D. provided feedback, longitudinal support, and funding for this work. B.H., J.R., K.L., M.L., and A.W. contributed environmental data sets. All authors contributed to the initial screening of tools, offered feedback throughout the process of analysis, and provided substantial comments to the manuscript.

AUTHOR AFFILIATIONS

[1]Department of Civil and Environmental Engineering, Case Western Reserve University, Cleveland, Ohio, USA
[2]Department of Microbiology, The Ohio State University, Columbus, Ohio, USA
[3]Department of Ecology and Evolutionary Biology, University of Michigan, Ann Arbor, Michigan, USA
[4]Department of Civil and Environmental Engineering, Stanford University, Palo Alto, California, USA
[5]Laboratory for Environmental Biotechnology, Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland
[6]Department of Civil and Environmental Engineering, University of Michigan, Ann Arbor, Michigan, USA

AUTHOR ORCIDs

Bridget Hegarty http://orcid.org/0000-0002-3291-4451
Melissa Duhaime http://orcid.org/0000-0001-7884-5087

FUNDING

Funder	Grant(s)	Author(s)
National Science Foundation (NSF)	DGE1256260, 2055455, DGE134012	James Riddell V Eric Bastien Morgan Lindback Melissa Duhaime

Funder	Grant(s)	Author(s)
DOC | National Oceanic and Atmospheric Administration (NOAA)	NA17OAR4320152	Anthony Wing Melissa Duhaime
College of Engineering, University of Michigan	Blue Sky Initiative	Bridget Hegarty Melissa Duhaime

AUTHOR CONTRIBUTIONS

Bridget Hegarty, Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Project administration, Supervision, Validation, Visualization, Writing – original draft, Writing – review and editing | James Riddell V, Data curation, Formal analysis, Investigation, Methodology, Validation, Visualization, Writing – original draft, Writing – review and editing | Eric Bastien, Investigation, Methodology, Writing – review and editing | Kathryn Langenfeld, Data curation, Methodology, Writing – review and editing | Morgan Lindback, Data curation, Methodology, Writing – review and editing | Jaspreet S. Saini, Data curation, Methodology, Writing – review and editing | Anthony Wing, Data curation, Methodology, Writing – review and editing | Jessica Zhang, Methodology, Writing – review and editing | Melissa Duhaime, Conceptualization, Investigation, Methodology, Project administration, Resources, Supervision, Visualization, Writing – review and editing

DATA AVAILABILITY

The scripts and data used here, including those to identify viral contigs from these tool outputs, example runs, and classifier outputs on the environmental metagenomic samples, are freely available at https://github.com/DuhaimeLab/VSTE. The environmental metagenomes are available on NCBI (accession numbers in Table S2).

ADDITIONAL FILES

The following material is available online.

Supplemental Material

Supplemental Figures (mSystems01105-23-S0001.pdf). Figures S1 to S19.
Supplemental Tables (mSystems01105-23-S0002.xlsx). Tables S1 to S4.

REFERENCES

1. Guidi L, Chaffron S, Bittner L, Eveillard D, Larhlimi A, Roux S, Darzi Y, Audic S, Berline L, Brum J, et al. 2016. Plankton networks driving carbon export in the oligotrophic ocean. Nature 532:465–470. https://doi.org/10.1038/nature16942
2. Wilhelm SW, Suttle CA. 1999. Viruses and nutrient cycles in the sea: viruses play critical roles in the structure and function of aquatic food webs. BioScience 49:781–788. https://doi.org/10.2307/1313569.
3. Howard-Varona C, Lindback MM, Bastien GE, Solonenko N, Zayed AA, Jang H, Andreopoulos B, Brewer HM, Glavina Del Rio T, Adkins JN, Paul S, Sullivan MB, Duhaime MB. 2020. Phage-specific metabolic reprogramming of virocells. ISME J 14:881–895. https://doi.org/10.1038/s41396-019-0580-z
4. Hurwitz BL, U'Ren JM. 2016. Viral metabolic reprogramming in marine ecosystems. Curr Opin Microbiol 31:161–168. https://doi.org/10.1016/j.mib.2016.04.002
5. Beumer A, Robinson JB. 2005. A broad-host-range, generalized transducing phage (SN-T) acquires 16S RRNA genes from different genera of bacteria. Appl Environ Microbiol 71:8301–8304. https://doi.org/10.1128/AEM.71.12.8301-8304.2005
6. Göller PC, Elsener T, Lorgé D, Radulovic N, Bernardi V, Naumann A, Amri N, Khatchatourova E, Coutinho FH, Loessner MJ, Gómez-Sanz E. 2021. Multi-species host range of staphylococcal phages isolated from waste water. Nat Commun 12:6965. https://doi.org/10.1038/s41467-021-27037-6
7. Sullivan MB. 2015. Not gene markers, for studying double-stranded DNA virus communities. J Virol 89:2459–2461. https://doi.org/10.1128/JVI.03289-14
8. Drake JW. 1999. The distribution of rates of spontaneous mutation over viruses, prokaryotes, and eukaryotes. Ann N Y Acad Sci 870:100–107. https://doi.org/10.1111/j.1749-6632.1999.tb08870.x
9. Peck KM, Lauring AS, Sullivan CS. 2018. Complexities of viral mutation rates. J Virol 92:e01031-17. https://doi.org/10.1128/JVI.01031-17
10. O'Leary NA, Wright MW, Brister JR, Ciufo S, Haddad D, McVeigh R, Rajput B, Robbertse B, Smith-White B, Ako-Adjei D, et al. 2016. Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation. Nucleic Acids Res 44:D733–D745. https://doi.org/10.1093/nar/gkv1189
11. Ponsero AJ, Hurwitz BL. 2019. The promises and pitfalls of machine learning for detecting viruses in aquatic metagenomes. Front Microbiol 10:806. https://doi.org/10.3389/fmicb.2019.00806
12. Guo J, Bolduc B, Zayed AA, Varsani A, Dominguez-Huerta G, Delmont TO, Pratama AA, Gazitúa MC, Vik D, Sullivan MB, Roux S. 2021. VirSorter2: a multi-classifier, expert-guided approach to detect diverse DNA and RNA viruses. Microbiome 9:37. https://doi.org/10.1186/s40168-020-00990-y

13. Andrade-Martínez JS, Camelo Valera LC, Chica Cárdenas LA, Forero-Junco L, López-Leal G, Moreno-Gallego JL, Rangel-Pineros G, Reyes A. 2022. Computational tools for the analysis of uncultivated phage genomes. Microbiol Mol Biol Rev 86:e0000421. https://doi.org/10.1128/mmbr.00004-21
14. Kieft K, Zhou Z, Anantharaman K. 2020. VIBRANT: automated recovery, annotation and Curation of microbial viruses, and evaluation of viral community function from genomic sequences. Microbiome 8:90. https://doi.org/10.1186/s40168-020-00867-0
15. Hegarty B, Dai Z, Raskin L, Pinto A, Wigginton K, Duhaime M. 2022. A snapshot of the global drinking water virome: diversity and metabolic potential vary with residual disinfectant use. Water Res 218:118484. https://doi.org/10.1016/j.watres.2022.118484
16. Gregory AC, Zayed AA, Conceição-Neto N, Temperton B, Bolduc B, Alberti A, Ardyna M, Arkhipova K, Carmichael M, Cruaud C, et al. 2019. Marine DNA viral macro- and microdiversity from pole to pole. Cell 177:1109–1123. https://doi.org/10.1016/j.cell.2019.03.040
17. Rocha UN da, Kasmanas JC, Kallies R, Saraiva JP, Brizola Toscan R, Štefanič P, Bicalho MF, Correa FB, Baştürk MN, Fousekis E, Barbosa LMV, Plewka J, Probst A, Baldrian P, Stadler P, CLUE-TERRA consortium. 2022. MuDoGer: multi-domain genome recovery from metagenomes made easy. bioRxiv. https://doi.org/10.1101/2022.06.21.496983
18. Vik D, Gazitúa MC, Sun CL, Zayed AA, Aldunate M, Mulholland MR, Ulloa O, Sullivan MB. 2021. Genome-resolved viral ecology in a marine oxygen minimum zone. Environ Microbiol 23:2858–2874. https://doi.org/10.1111/1462-2920.15313
19. Zhou Z, Martin C, Kosmopoulos JC, Anantharaman K. 2023. ViWrap: a modular pipeline to identify, bin, classify, and predict viral–host relationships for viruses from metagenomes. iMeta 2:e118. https://doi.org/10.1002/imt2.118
20. Pandolfo M, Telatin A, Lazzari G, Adriaenssens EM, Vitulo N. 2022 MetaPhage: an automated pipeline for analyzing, annotating, and classifying bacteriophages in metagenomics sequencing data. mSystems 7:e00741–22. https://doi.org/10.1128/msystems.00741-22
21. Ru J, Khan Mirzaei M, Xue J, Peng X, Deng L. 2023. ViroProfiler: a containerized bioinformatics pipeline for viral metagenomic data analysis. Gut Microbes 15:2192522. https://doi.org/10.1080/19490976.2023.2192522
22. Guo J. 2022. VirSorter 2 Database. Available from: https://doi.org/10.5281/zenodo.4297575
23. DeLong EF, Preston CM, Mincer T, Rich V, Hallam SJ, Frigaard N-U, Martinez A, Sullivan MB, Edwards R, Brito BR, Chisholm SW, Karl DM. 2006. Community genomics among stratified microbial assemblages in the ocean's interior. Science 311:496–503. https://doi.org/10.1126/science.1120250
24. Al-Shayeb B, Sachdeva R, Chen L-X, Ward F, Munk P, Devoto A, Castelle CJ, Olm MR, Bouma-Gregson K, Amano Y, et al. 2020. Clades of huge phages from across earth's ecosystem. Nature 578:425–431. https://doi.org/10.1038/s41586-020-2007-4
25. Tisza MJ, Belford AK, Domínguez-Huerta G, Bolduc B, Buck CB. 2021. Cenote-taker 2 democratizes virus discovery and sequence annotation. Virus Evol 7:veaa100. https://doi.org/10.1093/ve/veaa100
26. Nayfach S, Camargo AP, Schulz F, Eloe-Fadrosh E, Roux S, Kyrpides NC. 2021. Checkv assesses the quality and completeness of metagenome-assembled viral genomes. Nat Biotechnol 39:578–585. https://doi.org/10.1038/s41587-020-00774-7
27. Ren J, Song K, Deng C, Ahlgren NA, Fuhrman JA, Li Y, Xie X, Poplin R, Sun F. 2020. Identifying viruses from metagenomic data using deep learning. Quant Biol 8:64–77. https://doi.org/10.1007/s40484-019-0187-4
28. Czeczko P, Greenway SC, de Koning APJ, Birol I. 2017. Ezmap: a simple pipeline for reproducible analysis of the human virome. Bioinforma Oxf Engl 33:2573–2574. https://doi.org/10.1093/bioinformatics/btx202
29. Amgarten D, Braga LPP, da Silva AM, Setubal JC. 2018. MARVEL, a tool for prediction of bacteriophage sequences in metagenomic bins. Front Genet 9:304. https://doi.org/10.3389/fgene.2018.00304
30. Antipov D, Raiko M, Lapidus A, Pevzner PA. 2020. Metaviralspades: assembly of viruses from metagenomic data. Bioinformatics 36:4126–4129. https://doi.org/10.1093/bioinformatics/btaa490
31. Jurtz VI, Villarroel J, Lund O, Voldby Larsen M, Nielsen M. 2016. Metaphinder—identifying bacteriophage sequences in metagenomic data SETS. PLOS ONE 11:e0163111. https://doi.org/10.1371/journal.pone.0163111
32. Deaton J, Yu FB, Quake SR. 2019. Mini-metagenomics and nucleotide composition aid the identification and host association of novel bacteriophage sequences. Adv Biosyst 3:e1900108. https://doi.org/10.1002/adbi.201900108
33. Arndt D, Grant JR, Marcu A, Sajed T, Pon A, Liang Y, Wishart DS. 2016. PHASTER: a better, faster version of the PHAST phage search tool. Nucleic Acids Res 44:W16–W21. https://doi.org/10.1093/nar/gkw387
34. Starikova EV, Tikhonova PO, Prianichnikov NA, Rands CM, Zdobnov EM, Ilina EN, Govorun VM. 2020. Phigaro: high-throughput prophage sequence annotation. Bioinformatics 36:3882–3884. https://doi.org/10.1093/bioinformatics/btaa250
35. Fang Z, Tan J, Wu S, Li M, Xu C, Xie Z, Zhu H. 2019. PPR-meta: a tool for identifying phages and plasmids from metagenomic fragments using deep learning. GigaScience 8:giz066. https://doi.org/10.1093/gigascience/giz066
36. Liu F, Miao Y, Liu Y, Hou T. 2022. RNN-VirSeeker: a deep learning method for identification of short viral sequences from metagenomes. IEEE/ACM Trans Comput Biol Bioinform 19:1840–1849. https://doi.org/10.1109/TCBB.2020.3044575
37. Auslander N, Gussow AB, Benler S, Wolf YI, Koonin EV. 2020. Seeker: alignment-free identification of bacteriophage genomes by deep learning. Nucleic Acids Res 48:e121. https://doi.org/10.1093/nar/gkaa856
38. Li Y, Wang H, Nie K, Zhang C, Zhang Y, Wang J, Niu P, Ma X. 2016. VIP: an integrated pipeline for metagenomics of virus identification and discovery. Sci Rep 6:23774. https://doi.org/10.1038/srep23774
39. Tampuu A, Bzhalava Z, Dillner J, Vicente R. 2019. ViraMiner: deep learning on raw DNA sequences for identifying viral genomes in human samples. PLOS ONE 14:e0222271. https://doi.org/10.1371/journal.pone.0222271
40. Ren J, Ahlgren NA, Lu YY, Fuhrman JA, Sun F. 2017. VirFinder: a novel k-Mer based tool for identifying viral sequences from assembled metagenomic data. Microbiome 5:69. https://doi.org/10.1186/s40168-017-0283-5
41. Garretto A, Hatzopoulos T, Putonti C. 2019. VirMine: automated detection of viral sequences from complex metagenomic samples. PeerJ 7:e6695. https://doi.org/10.7717/peerj.6695
42. Zheng T, Li J, Ni Y, Kang K, Misiakou M-A, Imamovic L, Chow BKC, Rode AA, Bytzer P, Sommer M, Panagiotou GM. 2019. Analyzing, and integrating viral signals from metagenomic data. Microbiome 7:42. https://doi.org/10.1186/s40168-019-0657-y
43. Abdelkareem AO, Khalil MI, Elaraby M, Abbas H, Elbehery AHA. 2018. "VirNet: deep attention model for viral reads identification. 13th International Conference on Computer Engineering and Systems (ICCES), p 623–626. https://doi.org/10.1109/ICCES.2018.8639400
44. Wommack KE, Bhavsar J, Polson SW, Chen J, Dumas M, Srinivasiah S, Furman M, Jamindar S, Nasko DJ. 2012. VIROME: a standard operating procedure for analysis of viral metagenome sequences. Stand Genomic Sci 6:427–439. https://doi.org/10.4056/sigs.2945050
45. Rampelli S, Soverini M, Turroni S, Quercia S, Biagi E, Brigidi P, Candela M. 2016. ViromeScan: a new tool for metagenomic viral community profiling. BMC Genomics 17:165. https://doi.org/10.1186/s12864-016-2446-3
46. Roux S, Enault F, Hurwitz BL, Sullivan MB. 2015. VirSorter: mining viral signal from microbial genomic data. PeerJ 3:e985. https://doi.org/10.7717/peerj.985
47. Miao Y, Liu F, Hou T, Liu Y. 2022. Virtifier: a deep learning-based identifier for viral sequences from metagenomes. Bioinformatics 38:1216–1222. https://doi.org/10.1093/bioinformatics/btab845
48. Zhao G, Wu G, Lim ES, Droit L, Krishnamurthy S, Barouch DH, Virgin HW, Wang D. 2017. VirusSeeker, a computational pipeline for virus discovery and virome composition analysis. Virology 503:21–30. https://doi.org/10.1016/j.virol.2017.01.005
49. Glickman C, Hendrix J, Strong M. 2021. Simulation study and comparative evaluation of viral contiguous sequence identification tools. BMC Bioinformatics 22:329. https://doi.org/10.1186/s12859-021-04242-0
50. Ho SFS, Wheeler NE, Millard AD, van Schaik W. 2023. Gauge your phage: benchmarking of bacteriophage identification tools in metagenomic

sequencing data. Microbiome 11:84. https://doi.org/10.1186/s40168-023-01533-x
51. Menzel P, Ng KL, Krogh A. 2016. Fast and sensitive taxonomic classification for metagenomics with Kaiju. Nat Commun 7:11257. https://doi.org/10.1038/ncomms11257
52. Guo J, Vik D, Adjie Pratama A, Roux S, Sullivan M. 2021. Viral sequence identification SOP with VirSorter2. https://doi.org/10.17504/protocols.io.bwm5pc86
53. Chicco D, Jurman G. 2020. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. BMC Genomics 21:6. https://doi.org/10.1186/s12864-019-6413-7
54. Camargo AP, Roux S, Schulz F, Babinski M, Xu Y, Hu B, Chain PSG, Nayfach S, Kyrpides NC. 2023. You can move, but you can't hide: Identification of mobile genetic elements with GeNomad. bioRxiv. https://doi.org/10.1101/2023.03.05.531206
55. Roux S, Enault F, Robin A, Ravet V, Personnic S, Theil S, Colombet J, Sime-Ngando T, Debroas D. 2012. Assessing the diversity and specificity of two freshwater viral communities through metagenomics. PLoS One 7:e33641. https://doi.org/10.1371/journal.pone.0033641
56. Elbehery AHA, Deng L. 2022. Insights into the global freshwater virome. Front Microbiol 13:953500. https://doi.org/10.3389/fmicb.2022.953500
57. Gabrielli M, Dai Z, Delafont V, Timmers PHA, van der Wielen PWJJ, Antonelli M, Pinto AJ. 2023. Identifying eukaryotes and factors influencing their biogeography in drinking water metagenomes. Environ Sci Technol 57:3645–3660. https://doi.org/10.1021/acs.est.2c09010