doi: 10.1093/nargab/lqab004

Interpretable detection of novel human viruses from genome sequencing data

Jakub M. Bartoszewicz[1,2,3,4,*], Anja Seidel[1,2] and Bernhard Y. Renard[1,3,4,*]

[1]Bioinformatics (MF1), Department of Methodology and Research Infrastructure, Robert Koch Institute, 13353 Berlin, Germany, [2]Department of Mathematics and Computer Science, Free University of Berlin, 14195 Berlin, Germany, [3]Data Analytics and Computational Statistics, Hasso Plattner Institute for Digital Engineering, 14482 Potsdam, Brandenburg, Germany and [4]Digital Engineering Faculty, University of Postdam, 14482 Potsdam, Brandenburg, Germany

Received October 12, 2020; Revised January 04, 2021; Editorial Decision January 09, 2021; Accepted January 15, 2021

ABSTRACT

Viruses evolve extremely quickly, so reliable methods for viral host prediction are necessary to safeguard biosecurity and biosafety alike. Novel human-infecting viruses are difficult to detect with standard bioinformatics workflows. Here, we predict whether a virus can infect humans directly from next-generation sequencing reads. We show that deep neural architectures significantly outperform both shallow machine learning and standard, homology-based algorithms, cutting the error rates in half and generalizing to taxonomic units distant from those presented during training. Further, we develop a suite of interpretability tools and show that it can be applied also to other models beyond the host prediction task. We propose a new approach for convolutional filter visualization to disentangle the information content of each nucleotide from its contribution to the final classification decision. Nucleotide-resolution maps of the learned associations between pathogen genomes and the infectious phenotype can be used to detect regions of interest in novel agents, for example, the SARS-CoV-2 coronavirus, unknown before it caused a COVID-19 pandemic in 2020. All methods presented here are implemented as easy-to-install packages not only enabling analysis of NGS datasets without requiring any deep learning skills, but also allowing advanced users to easily train and explain new models for genomics.

INTRODUCTION

Background

Within a globally interconnected and densely populated world, pathogens can spread more easily than they ever had before. As the recent outbreaks of Ebola and Zika viruses have shown, the risks posed even by these previously known agents remain unpredictable and their expansion hard to control (1). What is more, it is almost certain that more unknown pathogen species and strains are yet to be discovered, given their constant, extremely fast-paced evolution and unexplored biodiversity, as well as increasing human exposure (2,3). Some of those novel pathogens may cause epidemics (similar to the SARS and MERS coronavirus outbreaks in 2002 and 2012) or even pandemics (e.g. SARS-CoV-2 and the 'swine flu' H1N1/09 strain). Many have more than one host or vector, which makes assessing and predicting the risks even more difficult. For example, Ebola has its natural reservoir most likely in fruit bats (4), but causes deadly epidemics in both humans and chimpanzees. As the state-of-the art approach for the open-view detection of pathogens is genome sequencing (5,6), it is crucial to develop automated pipelines for characterizing the infectious potential of currently unidentifiable sequences. In practice, clinical samples are dominated by host reads and contaminants, with often less than a hundred reads of the pathogenic virus (7). Metagenomic assembly is challenging, especially in time-critical applications. This creates a need for read-based approaches complementing or substituting assembly where needed.

Screening against potentially dangerous subsequences before their synthesis may also be used as a way of ensuring responsible research in synthetic biology. While potentially useful in some applications, engineering of viral genomes could also pose a biosecurity and biosafety threat. Two controversial studies modified the influenza A/H5N1 ('bird flu') virus to be airborne transmissible in mammals

[*]To whom correspondence should be addressed. Tel: +49 331 5509 4973; Email: jakub.bartoszewicz@hpi.de
Correspondence may also be addressed to Bernhard Y. Renard. Tel: +49 331 5509 4960; Email: bernhard.renard@hpi.de
Present address: Anja Seidel, Central Research Institute of Ambulatory Health Care, 10587 Berlin, Germany.

(8,9). A possibility of modifying coronaviruses to enhance their virulence triggered calls for a moratorium on this kind of research (10). Synthesis of an infectious horsepox virus closely related to the smallpox-causing *Variola* virus (11) caused a public uproar and calls for intensified discussion on risk control in synthetic biology (12).

Current tools for host range prediction

Several computational, genome-based methods exist that allow to predict the host-range of a bacteriophage (a bacteria-infecting virus). A selection of composition-based and alignment-based approaches has been presented in an extensive review by Edwards *et al.* (13). Prediction of eukariotic host tropism (including humans) based on known protein sequences was shown for the influenza A virus (14). Support-vector machines based on word2vec representations were shown to outperform homology searches with BLAST and HMMs in the same task, but lost their advantage when applied to nucleic acid sequences directly (15). Two recent studies employ *k*-mer based, *k*-NN classifiers (16), and deep learning (17) to predict host range for a small set of three well-studied species directly from viral sequences. While those approaches are limited to those particular species and do not scale to viral host-range prediction in general, the Host Taxon Predictor (HTP) (18) uses logistic regression and supports vector machines to predict if a novel virus infects bacteria, plants, vertebrates or arthropods. Yet, the authors argue that it is not possible to use HTP in a read-based manner; it requires long sequences of at least 3000 nucleotides. This is incompatible with modern metagenomic next-generation sequencing (NGS) workflows, where the DNA reads obtained are at least 10–20 times shorter. Another study used gradient boosting machines to predict reservoir hosts and transmission via arthropod vectors for known human-infecting viruses (19).

Zhang *et al.* (20) designed several classifiers explicitly predicting whether a new virus can potentially infect humans. Their best model, a *k*-NN classifier, uses *k*-mer frequencies as features representing the query sequence and can yield predictions for sequences as short as 500 base pairs (bp). It worked also with 150 bp-long reads from real DNA sequencing runs, although in this case the reads originated also from the viruses present in the training set (and were therefore not ‘novel’).

Deep learning for DNA sequences

While DNA sequences mapped to a reference genome may be represented as images (21), a majority of studies uses a distributed orthographic representation, where each nucleotide $\{A, C, G, T\}$ in a sequence is represented by a one-hot encoded vector of length 4. An ‘unknown’ nucleotide (N) can be represented as an all-zero vector. Chaos game representation (CGR) and its extension, the frequency matrix CGR (FCGR) are promising alternatives able to encode an arbitrary sequence in an image-like format. FCGR has been used to encode genomic inputs for deep learning approaches, including full bacterial genomes (22) and coding sequences of HIV for the drug resistance prediction task (23). In this study, we use one-hot encoding with *N*s as zeroes, which was previously shown to perform well for raw NGS reads (24) and abstract phenotype labels.

Convolutional neural networks (CNNs) and long short-term memory architectures (LSTMs) have been successfully used for a variety of DNA-based prediction tasks. Early works focused mainly on regulation of gene expression in humans (25–29), which is still an area of active research (30–32). In the field of pathogen genomics, deep learning models trained directly on DNA sequences were developed to predict host ranges of three multi-host viral species (17) and to predict pathogenic potentials of novel bacteria (24). DeepVirFinder (33) and ViraMiner (34) can detect viral sequences in metagenomic samples, but they cannot predict the host and focus on previously known species. For a broader view on deep learning in genomics, we refer to a recent review by Eraslan *et al.* (35).

Interpretability and explainability of deep learning models for genomics is crucial for their wide-spread adoption, as it is necessary for delivering trustworthy and actionable results. Convolutional filters can be visualized by forward-passing multiple sequences through the network and extracting the most-activating subsequences (25) to create a position weight matrix (PWM) that can be visualized as a sequence logo (36,37). Direct optimization of input sequences is problematic, as it results in generating a dense matrix even though the input sequences are one-hot encoded (38,39). This problem can be alleviated with Integrated Gradients (40,41) or DeepLIFT, which propagates activation differences relative to a selected reference back to the input, reducing the computational overhead of obtaining accurate gradients (42). If the bias terms are zero and a reference of all-zeros is used, the method is analogous to Layer-wise Relevance Propagation (43). DeepLIFT is an additive feature attribution method, and may used to approximate Shapley values if the input features are independent (44). TF-MoDISco (45) uses DeepLIFT to discover consolidated, biologically meaningful DNA motifs (transcription factor binding sites).

Contributions

In this paper, we first improve the performance of read-based predictions of the viral host (human or nonhuman) from next-generation sequencing reads. We show that reverse-complement (RC) neural networks (24) significantly outperform both the previous state-of-the-art (20) and the traditional, alignment-based algorithm—BLAST (46,47), which constitutes a gold standard in homology-based bioinformatics analyses. We show that defining the negative (nonhuman) class is nontrivial and compare different ways of constructing the training set. Strikingly, a model trained to distinguish between viruses infecting humans and viruses infecting other chordates (a phylum of animals including vertebrates) generalizes well to evolutionarily distant nonhuman hosts, including even bacteria. This suggests that the host-related signal is strong and the learned decision boundary separates human viruses from other DNA sequences surprisingly well.

Next, we propose a new approach for convolutional filter visualization using partial Shapley values to differentiate between simple nucleotide information content and the contribution of each sequence position to the final classification score. To test the biological plausibility of our models, we generate genome-wide maps of 'infectious potential' and nucleotide contributions. We show that those maps can be used to visualize and detect virulence-related regions of interest (e.g. genes) in novel genomes.

As a proof of concept, we analyzed one of the viruses randomly assigned to the test set—the Taï Forest ebolavirus, which has a history of host-switching and can cause a serious disease. To show that the method can also be used for other biological problems, we investigated the networks trained by Bartoszewicz *et al*. (24) and their predictions on a genome of a pathogenic bacterium *Staphylococcus aureus*. The authors used this particular species to assess the performance of their method on real sequencing data. Finally, we studied the SARS-CoV-2 coronavirus, which emerged in December 2019, causing the COVID-19 pandemic (48).

MATERIALS AND METHODS

Data collection and preprocessing

VHDB dataset. We accessed the Virus-Host Database (49) on July 31, 2019 and downloaded all the available data. We note that all the reference genomes from NCBI Viral Genomes are present in VHDB, as well as their curated annotations from RefSeq. Additional, manually curated records in VHDB extend on metadata available in NCBI. More nonreference genomes are available, but considering multiple genomes per virus would skew the classifiers' performance toward the more frequently resequenced ones.

The original dataset contained 14380 records comprising RefSeq IDs for viral sequences and associated metadata. Some viruses are divided into discontiguous segments, which are represented as separate records in VHDB; in those cases, the segments were treated as contigs of a single genome in the further analysis. We removed records with unspecified host information and those confusing the highly pathogenic Variola virus with a similarly named genus of fish. Following Zhang *et al*. (20), we filtered out viroids and satellites, which are classified as subviral agents and not *bona fide* viruses (50,51). Note that even though they require helper viruses for replication, this step did not affect ubiquitous adeno-associated viruses and large virophages, which are well established within the viral taxonomy in the families *Parvoviridae* and *Lavidaviridae*, respectively. Human-infecting viruses were extracted by searching for records containing '*Homo sapiens*' in the 'host name' field. Note that VHDB contains information about multiple possible hosts for a given virus where appropriate. Any virus infecting humans was assigned to the positive class, also if other, nonhuman hosts exist. In total, the dataset contained 9496 viruses (grouped in 7503 species), including 1309 human viruses (393 species). We considered both DNA and RNA viruses; RNA sequences were encoded in the DNA alphabet, as in RefSeq.

Defining the negative class. While defining a human-infecting class is relatively straightforward, the reference negative class may be conceptualized in a variety of ways. The broadest definition takes all nonhuman viruses into account, including bacteriophages (bacterial viruses). This is especially important, as most of known bacteriophages are DNA viruses, while many important human (and animal) viruses are RNA viruses. One could expect that the multitude of available bacteriophage genomes dominating the negative class could lower the prediction performance on viruses similar to those infecting humans. This offers an open-view approach covering a wider part of the sequence space, but may lead to misclassification of potentially dangerous mammalian or avian viruses. As they are often involved in clinically relevant host-switching events, a stricter approach must also be considered. In this case, the negative class comprises only viruses infecting Chordata (a group containing vertebrates and closely related taxa). Two intermediate approaches consider all eukaryotic viruses (including plant and fungi viruses) or only animal-infecting viruses. This amounts to four nested host sets: 'All' (8187 nonhuman viruses, 7110 species), 'Eukaryota' (5114 viruses, 4275 species), 'Metazoa' (2942 viruses, 2351 species) and 'Chordata' (2078 viruses, 1530 species). Auxiliary sets containing only noneukaryotic viruses ('non-Eukaryota'), nonanimal eukaryotic viruses ('non-Metazoa Eukaryota'), etc., can be easily constructed by set subtraction.

For the positive class, we randomly generated a training set containing 80% of the genomes, and validation and test sets with 10% of the genomes each. Importantly, the nested structure was kept also during the training-validation-test split: for example, the species assigned to the smallest test set ('Chordata') were also present in all the bigger test sets. The same applied to other taxonomic levels, as well as the training and validation sets wherever applicable.

Read simulation. We simulated 250 bp long Illumina reads following a modification of a previously described protocol (24) and using the Mason read simulator (52). First, we only generated the reads from the genomes of human-infecting viruses. Then, the same steps were applied to each of the four negative class sets. Finally, we also generated a fifth set, 'Stratified', containing an equal number of reads drawn from genomes of the following disjunct host classes: 'Chordata' (25%), 'non-Chordata Metazoa' (25%), 'non-Metazoa Eukaryota' (25%) and 'non-Eukaryota' (25%).

In each of the evaluated settings, we used a total of 20 million (80%) reads for training, 2.5 million (10%) reads for validation and 2.5 million (10%) paired reads as the held-out test set. Read number per genome was proportional to genome length, keeping the coverage uniform on average. Viruses with longer genomes were therefore represented by more reads than shorter viruses. On the other hand, their sequence diversity was covered at a similar level. This length-balancing step was previously shown to work well for bacterial genomes of different lengths (24,53). While the original datasets are heavily imbalanced, we generated the same number of negative and positive data points (reads) regardless of the negative class definition used.

This protocol allowed us to test the impact of defining the negative class, while using the exactly same data as representatives of the positive class. We used three training and validation sets ('All', 'Stratified' and 'Chordata'), representing the fully open-view setting, a setting more balanced with regard to the host taxonomy, and a setting focused on cases most likely to be clinically relevant. In each setting, the validation set matched the composition of the training set. The evaluation was performed using all five test sets to gain a more detailed insight on the effects of negative class definition on the prediction performance.

Human blood virome dataset. Similarily to Zhang *et al.* (20), we used the human blood DNA virome dataset (54) to test the selected classifiers on real data. We obtained 14,242,329 reads of 150 bp and searched all of VHDB using blastn (with default parameters) to obtain high-quality reference labels. If a read's best hit was a human-infecting virus, we assigned it to a positive class; the negative class was assigned if this was not the case. This procedure yielded 14 012 665 'positive' and 229 664 'negative' reads.

Virus-level and species-level predictions. In this study, we focus on predicting labels for reads originating from novel viruses. What constitutes a 'novel' biological entity is an open question—a novel virus does not necessarily belong to a novel species (55). If a given viral isolate clusters with a known group of isolates, it is considered to be the same virus; if it does not, it may be assigned a distinct name and considered novel (55). This is separate from its putative taxonomical assignment. Assigning a novel virus to a novel or a previously established species is performed pursuing a wider set of criteria, and the criteria for delineating distinct species differ between viral families (50,51,55,56). In most cases, species are perceived as human constructs rather than biological entities and host range often is explicitly one of the defining features (55,57), rendering reasoning based on cross-species homology searches inherently difficult.

The most prominent example of this problem is the SARS-CoV-2 virus, which is a novel virus within a previously known species (*Severe acute respiratory syndrome–related coronavirus*). Other members of this species include the human-infecting SARS-CoV-1, but also multiple related bat SARSr-CoV viruses (e.g. SARSr-CoV RaTG13 or Bat SARS-like coronavirus WIV1). Importantly, SARS-CoV-2 is not a strain of SARS-CoV-1; those two viruses share a common ancestor (55). This echoes similar problems related to pathogenic potential prediction for novel bacterial pathogens. A novel bacterium may be defined as a novel strain or a novel species (24), and the classifiers must be trained according to the desired definition.

As the 2020 pandemic has shown, different viruses of the same species can differ wildly in their infectious potential and the broader impact on human societies. Therefore, threat assessment must be performed for novel viruses, not only novel taxa; different related viruses are nonredundant. At the same time, redundancy below this level (i.e. multiple instances of the same virus) must be eliminated from the dataset to ensure reliability of the trained classifier. VHDB tackles this problem by collecting and annotating reference genomes—each virus in the database is a separate entity with its own ID in NCBI Taxonomy. This virus-level approach was previously used by Zhang *et al.* (20). We show that homology-based algorithms underperform in this setting already, suggesting that machine learning is indeed required to accurately predict labels for novel viruses even if other members of the same species are present in the training database.

Nevertheless, a more difficult alternative—predictions for reads of viruses belonging to completely novel species—is a related and potentially equally important task. For bacterial datasets, species novelty can be modeled by selecting a single representative genome per species (24). As the SARS-CoV-2 example shows, this is often not possible for viruses. To assess our approach in this stricter setup, we re-divided the VHDB dataset into training, validation and test sets ensuring that all viruses of a given species were assigned to only one of those subsets. This effectively models a 'novel species' scenario while also reflecting within-species phenotype diversity. We recreated the species-wide versions of the 'All' and 'Chordata' datasets by assigning 80%, 10% and 10% of the species to the training, validation and test datasets, respectively. We resimulated the reads as outlined above and compared the performance of the machine learning and homology-based approaches achieving the highest accuracy in the simpler 'novel virus' setting (see 'Prediction performance' section).

Training

We used the DeePaC package (24) to investigate RC-CNN and RC-LSTM architectures, which guarantee identical predictions for both forward and reverse-complement orientations of any given nucleotide sequence, and have been previously shown to accurately predict bacterial pathogenicity. Here, we employ an RC-CNN with two convolutional layers with 512 filters of size 15 each, average pooling and 2 fully connected layers with 256 units each. The LSTM used has 384 units (Supplementary Figure S1). We use dropout regularization in both cases, together with aggressive input dropout at the rate of 0.2 or 0.25 (tuned for each model). Input dropout may be interpreted as a special case of noise injection, where a fraction of input nucleotides is turned to *N*s. Representations of forward- and reverse-complement strands are summed before the fully connected layers. As two mates in a read pair should originate from the same virus, predictions obtained for them can be averaged for a boost in performance. If a contig or genome is available, averaging predictions for constituting reads yields a prediction for the whole sequence. We used Tesla P100 and Tesla V100 GPUs for training and an RTX 2080 Ti for visualizations.

We wanted the networks to yield accurate predictions for both 250 bp (our data, modeling a sequencing run of an Illumina MiSeq device) and 150 bp long reads (as in the Human Blood Virome dataset). As shorter reads are padded with zeros, we expected the CNNs trained using average pooling to misclassify many of them. Therefore, we prepared a modified version of the datasets, in which the last 100 bp of each read were turned to zeros, mocking a shorter sequencing run while preserving the error model. Then, we retrained the CNN that had performed best on the original dataset. Since in principle, the Human Blood Virome dataset should not contain viruses infecting nonhuman Chordata, a 'Chordata'-trained classifier was not used in this setting.

Benchmarking

We compare our networks to the the *k*-NN classifier proposed by Zhang *et al*. (20), the only other approach explicitly tested on raw NGS reads and detecting human viruses in a fully open view setting (not focusing on a limited number of species). We use the real sequencing data that they used (54) for an unbiased comparison.

We trained the classifier on the 'All' dataset as described by the authors, i.e. using nonoverlapping, 500 bp-long contigs generated from the training genomes (retraining on simulated reads is computationally prohibitive). We also tested the performance of using BLAST to search against an indexed database of labeled genomes. We constructed the database from the 'All' training set and used discontiguous megablast to achieve high inter-species sensitivity. For NGS mappers (BWA-MEM (58) and Bowtie2 (59)), the indices were constructed analogously. Kraken (60) was previously shown to perform worse than both BLAST and machine learning when faced with read-based pathogenic potential prediction for novel bacterial species (53). Its major advantage—assigning reads to lowest common ancestor (LCA) nodes in ambiguous cases—turns into a problem in the infectivity prediction task, as transferring labels to LCAs is often impossible (53). Therefore, we focus on alignment-based approaches as the most accurate alternative to machine learning in this context.

Note that both alignment and *k*-NN can yield conflicting predictions for the individual mates in a read pair. What is more, BLAST and the mappers yield no prediction at all if no match is found. Therefore, similarly to Bartoszewicz *et al*. (24), we used the accept anything operator to integrate binary predictions for read pairs and genomes. At least one match is needed to predict a label, and conflicting predictions are treated as if no match was found at all. Missing predictions lower both true positive and true negative rates.

Filter visualization

Substring extraction. In order to visualize the learned convolutional filters, we downsample a matching test set to 125 000 reads and pass it through the network. This is modeled after the method presented by Alipanahi *et al*. (25). For each filter and each input sequence, the authors extracted a subsequence leading to the highest activation, and created sequence logos from the obtained sequence sets ('max-activation'). We used the DeepSHAP implementation (44) of DeepLIFT (42) to extract score-weighted subsequences with the highest contribution score ('max-contrib') or all score-weighted subsequences with nonzero contributions ('all-contrib'). Computing the latter was costly and did not yield better quality logos.

We use an all-zero reference. As reads from real sequencing runs are usually not equally long, shorter reads must be padded with *N*s; the 'unknown' nucleotide is also called whenever there is not enough evidence to assign any other to the raw sequencing signal. Therefore, *N*s are 'null' nucleotides and are a natural candidate for the reference input. We do not consider alternative solutions based on GC content or dinucleotide shuffling, as the input reads originate from multiple different species, and the sequence composition may itself be a strong marker of both virus and host taxonomy (13). We also avoid weight-normalization suggested for zero-references (42), as it implicitly models the expected GC content of all possible input sequences, and assumes no *N*s present in the data. Finally, we calculate average filter contributions to obtain a crude ranking of feature importance with regard to both the positive and negative class.

Partial Shapley values. Building sequence logos involves calculating information content (IC) of each nucleotide at each position in a prospective DNA motif. This can be then interpreted as a measure of evolutionary sequence conservation. However, high IC does not necessarily imply that a given nucleotide is relevant in terms of its contribution to the classifier's output. Some sub-motifs may be present in the sequences used to build the logo, even if they do not contribute to the final prediction (or even a given filter's activation).

To test this hypothesis, we introduce partial Shapley values. Intuitively speaking, we capture the contributions of a nucleotide to the network's output, but only in the context of a given intermediate neuron of the convolutional layer. More precisely, for any given feature x_i, intermediate neuron y_j and the output neuron z, we aim to measure how x_i contributes to z while regarding only the fraction of the total contribution of x_i that influences how y_j contributes to z. Although similarly named concepts were mentioned before as intermediate computation steps in a different context (61,62), we define and use partial Shapley values to visualize contribution flow through convolutional filters. This differs from recently introduced contribution weight matrices (32), where feature attributions are used as a representation of an identified transcription factor binding site irreducible to a given intermediate neuron.

Using the formalism of DeepLIFT's multipliers (42) and their reinterpretation in SHAP (44), we backpropagate the activation differences only along the paths 'passing through' y_j. In (Equation 1), we define partial multipliers $\mu^{(y_j)}_{x_i z}$ and express them in terms of Shapley values ϕ and activation differences with respect to the expected activation values (reference activation). Calculating partial multipliers is equivalent to zeroing out the multipliers $m_{y_k z}$ for all $k \neq j$ before backpropagating $m_{y_j z}$ further.

$$\mu^{(y_j)}_{x_i z} = m_{x_i y_j} m_{y_j z} = \frac{\phi_i(y_j, x)\phi_j(z, y)}{(x_i - E[x_i])(y_j - E[y_j])} \qquad (1)$$

We define partial Shapley values $\varphi_i^{(y_j)}(z, x)$ analogously to how Shapley values can be approximated by a product of multipliers and input differences with respect to the reference (Equation 2).

$$\varphi_i^{(y_j)}(z, x) = \mu^{(y_j)}_{x_i z}(x_i - E[x_i]) = \frac{\phi_i(y_j, x)\phi_j(z, y)}{y_j - E[y_j]} \qquad (2)$$

From the chain rule for multipliers (42), it follows that standard multipliers are a sum over all partial multipliers for a given layer y. Therefore, Shapley values as approximated by DeepLIFT are a sum of partial Shapley values for the layer y (Equation 3).

$$\phi_i(z, x) = m_{x_i z}(x_i - E[x_i]) = \sum_j \varphi_i^{(y_j)}(z, x) \qquad (3)$$

Once we calculate the contributions of convolutional filters for the first layer, $\varphi_i^{(y_j)}(z, x)$ for the first convolutional layer of a network with one-hot encoded inputs and an all-zero reference can be efficiently calculated using weight matrices and filter activation differences (Equations 4 and 5). First, in this case we do not traverse any nonlinearities and can directly use the linear rule (42) to calculate the contributions of x_i to y_j as a product of the weight w_i and the input x_i. Second, the input values may only be 0 or 1.

$$\phi_i(y_j, x) = w_i x_i = \begin{cases} w_i, & \text{if } x_i = 1 \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

$$\varphi_i^{(y_j)}(z, x) = \frac{w_i \phi_j(z, y)}{y_j - E[y_j]} \tag{5}$$

Resulting partial contributions can be visualized along the IC of each nucleotide of a convolutional kernel. To this end, we design extended sequence logos, where each nucleotide is colored according to its contribution. Positive contributions are shown in red, negative contributions are blue, and near-zero contributions are gray. Therefore, no information is lost compared to standard sequence logos, but the relevance of individual nucleotides and the filter as a whole can be easily seen. Color saturation is limited by the reciprocal of a user-defined gain parameter, here set to *nm*, where *n* equals the number of input features x_i (sequence length) and *m* equals the number of convolutional filters y_j in a given layer.

Genome-wide phenotype analysis

We create genome-wide phenotype analysis (GWPA) plots to analyze which parts of a viral genome are associated with the infectious phenotype. We scramble the genome into overlapping, 250 bp long subsequences (pseudo-reads) without adding any sequencing noise. For the highest resolution, we use a stride of one nucleotide. For *S. aureus*, we used a stride of 125 bp. We predict the infectious potential of each pseudo-read and average the obtained values at each position of the genome. Analogously, we calculate average contributions of each nucleotide to the final prediction of the convolutional network. Finally, we normalize raw infectious potentials into the [−0.5, 0.5] interval for a more intuitive graphical representation. We visualize the resulting nucleotide-resolution maps with IGV (63). For protein structures, we average the scores codon-wise to obtain contribution scores per amino acid and visualize them with PyMOL (64).

For well-annotated genomes, we compile a ranking of genes (or other genomic features) sorted by the average infectious potential within a given region. In addition to that, we scan the genome with the learned filters of the first convolutional layer to find genes enriched in subsequences yielding nonzero filter activations. We use Gene Ontology to connect the identified genes of interest with their molecular functions and biological processes they are engaged in.

RESULTS

Negative class definition

Choosing which viruses should constitute the negative class is application dependent and influences the performance of the trained models. Supplementary Table S1 summarizes the prediction accuracy for different combinations of the training and test set composition. The models trained only on human and Chordata-infecting viruses maintain similar, or even better performance when evaluated on viruses infecting a much broader host range, including bacteria. This suggests that the learned decision boundary separates human viruses from all the others surprisingly well. We hypothesize that the human host signal must be relatively strong and contained within the Chordata host signal. Dropout rate of 0.2 resulted in the highest validation accuracy for $\text{CNN}_{\text{Str-150}}$ and LSTM_{Str}. A rate of 0.25 was selected for the other models.

Adding more diversity to the negative class may still boost performance on more diverse test sets, as in the case of CNN trained on the 'All' dataset (CNN_{All}). This model performs a bit worse on viruses infecting hosts related to humans, but achieves higher accuracy than the 'Chordata'-trained models and the best recall overall. Rebalancing the negative class using the 'Stratified' dataset helps achieve higher performance on animal viruses while maintaining high overall accuracy. The LSTMs are outperformed by the CNNs, but they can be used for shorter reads without retraining (see 'Training and Prediction performance' sections).

Prediction performance

We selected LSTM_{All} and CNN_{All} for further evaluation. We used a single consumer-grade RTX 2080 Ti GPU to measure inference speed. The CNN classifies 5000 reads/s and the LSTM 1855 reads/s. Analyzing ten million reads takes only 33 min using the faster model; linear speedups are possible if more GPUs are available. Therefore, the trained models achieve high-throughputs necessary to analyze NGS datasets. Table 1 presents the results of a benchmark using the 'All' test set. Low performance of the *k*-NN classifier (20) is caused by frequent conflicting predictions for each read in a read pair. In a single-read setting it achieves 75.5% accuracy, while our best model achieves 87.8% (Supplementary Table S2). Although BLAST achieves high precision, it yields no predictions for over 10% of the samples. CNN_{All} is the most sensitive and accurate. As expected, standard mapping approaches (BWA-MEM and Bowtie2) struggle with analyzing novel pathogens—they are the most precise but the least sensitive. Our approach outperforms them by 15–30%.

Although we focus on the extreme case of read-based predictions, our method can also be used on assembled contigs and full genomes if they are available, as well as on read sets from pure, single-virus samples. We note that assembly itself does not yield any labels and a follow-up analysis (via alignment, machine learning or other approaches) is required to correctly classify metagenomic contigs in any case. We ran predictions on contigs without any size filtering with both *k*-NN and BLAST (Table 2). We present performance mea-

Table 1. Classification performance in the fully open-view setting (all virus hosts), read pairs. Acc. – accuracy, Prec. – precision, Rec. – recall, Spec. – specificity. Bowtie2, BWA-MEM and BLAST yield no predictions for over 35%, 19% and 10% of the samples, respectively. Best performance in bold

	Acc.	Prec.	Rec.	Spec.
CNN$_{\text{All}}$ (ours)	**89.9**	93.9	**85.4**	**94.4**
LSTM$_{\text{All}}$ (ours)	86.4	89.0	83.0	89.8
k-NN	57.1	57.8	52.1	62.0
Bowtie2	58.6	**99.2**	59.2	58.0
BWA-MEM	72.8	98.9	73.9	71.8
BLAST	80.6	98.4	79.1	82.2

Table 2. Classification performance, all hosts. Whole available genomes. Negative class is the majority class. BAcc. – balanced accuracy, Rec. – recall, Spec. – specificity. BLAST (reads) and our networks use read-wise majority vote or output averaging to aggregate predictions over all reads from a genome. *k*-NN (genome) and BLAST (genome) use contig-wise majority vote. *k*-NN (contigs) and BLAST (contigs) represent performance on individual contigs treated as separate entities. *k*-NN (reads) was not used, as high conflicting prediction rates made read-wise aggregation impracticable

	Bacc.	AUPR	Rec.	Spec.
CNN$_{\text{All}}$ (ours)	**91.7**	**91.2**	89.3	94.2
LSTM$_{\text{All}}$ (ours)	86.3	85.8	**96.2**	76.4
BLAST (reads)	90.3	n/a	85.5	**95.1**
k-NN (genome)	82.8	n/a	93.9	71.6
BLAST (genome)	90.5	n/a	86.3	94.6
k-NN (contigs)	83.0	n/a	94.3	71.6
BLAST (contigs)	88.4	n/a	87.1	89.7

Table 3. Classification performance on the human blood virome dataset. Positive class is the majority class. BAcc. – balanced accuracy, Rec. – recall, Spec. – specificity

	BAcc.	AUPR	Rec.	Spec.
CNN$_{\text{All-150}}$ (ours)	**96.8**	**>99.9**	**97.3**	**96.2**
LSTM$_{\text{All}}$ (ours)	91.8	**>99.9**	88.2	95.5
k-NN	83.1	99.5	80.9	85.4

sures for both individual contigs and whole genome predictions based on contig-wise majority vote. We compare them to BLAST with read-wise majority vote (53) and to read-wise average predictions of our networks, analogous to presented previously for bacteria (24). Our method outperforms BLAST by 1.2% and *k*-NN by 8.9%, even though they have access to the full biological context (full sequences of all contigs in a genome), while we simply average outputs for short reads originating from the contigs.

We benchmarked our models against the human blood virome dataset used by Zhang *et al.* (20). Our models outperform their *k*-NN classifier. As the positive class massively outnumbers the negative class, all models achieve over 99% precision. CNN$_{\text{All-150}}$ performs best (Table 3). However, the positive class is dominated by viruses that are not necessarily novel. The CNN was more accurate on training data, so we expected it to detect those viruses easily.

Finally, we repeated the analysis in the 'novel species' scenario. Classifying novel viral species when restricted to Chordata-infecting viruses is too challenging for practical purposes (Supplementary Table S3). Read-wise predictions are not much better than random guesses for both BLAST and CNNs. Low precision of BLAST shows that it often recovers wrong labels even when it does find a match—sequence similarity is not a reliable predictor of the infectious potential in this setting. Even if a whole genome is available, overall accuracy is low. This looks very differently in the fully open view scenario (Table 4). The CNN trained on the species-wise division of the 'All' dataset (CNN$_{\text{SP-All}}$) outperforms BLAST by a wide margin in the read-wise setting. Strikingly, CNN$_{\text{SP-All}}$ predictions based on a single read pair achieve higher accuracy than BLAST predictions using whole genomes, mainly due to their significantly higher recall. What is more, pooling predictions from all the reads originating from a given genome does not improve overall CNN$_{\text{SP-All}}$ accuracy any further. As CNN$_{\text{SP-All}}$ does not reliably outperform its Chordata-trained analog on the 'Chordata' dataset (CNN$_{\text{SP-Cho}}$, Supplementary Table S3), we suspect that its relatively high accuracy on the 'All' dataset is caused by its high sensitivity while maintaining good specificity on non-Chordata viruses. However, classification accuracy is still noticeably lower than for the virus-level classification scenario. The virus-level models are not optimized for entirely novel species, effectively treating them as out-of-distribution samples. This suggests that they might overfit to (potentially new) viruses of known species. Therefore, all predictions for novel agents, whether based on machine learning or sequence homology, must be handled with caution.

Table 4. Classification performance, novel species. Top: paired reads (see Table 1). BLAST yields predictions for only 64.3% of the pairs. Bottom: whole available genomes or contigs – negative class is the majority class (see Table 2). BAcc. – balanced accuracy (equal to accuracy for the balanced paired-read dataset), Rec. – recall, Spec. – specificity. BLAST (reads) and our networks use read-wise majority vote or output averaging to aggregate predictions over all reads from a genome. BLAST (genome) uses contig-wise majority vote. BLAST (contigs) represents performance on individual contigs treated as separate entities. Note that low precision is heavily affected by class imbalance

	BAcc.	Prec.	Rec.	Spec.
CNN$_{\text{SP-All}}$ (ours)	**74.6**	87.0	**57.9**	**91.4**
BLAST	47.1	**94.1**	17.8	76.4
CNN$_{\text{SP-All}}$ (ours)	**64.9**	31.0	**40.6**	89.1
BLAST (reads)	61.8	**46.8**	30.2	**93.5**
BLAST (genome)	64.0	44.9	36.5	91.5
BLAST (contigs)	57.9	37.9	33.6	82.1

Filter visualization

Over 84% of all contributing first-layer filters in CNN$_{\text{All}}$ have positive average contribution scores. We comment more on this fact in 'Nucleotide contribution logos' section. For CNN$_{\text{All}}$, the average information content of our motifs is strongly correlated nucleotide-wise with IC of DeepBind-like logos (Spearman's $\rho > 0.95$, $P < 10^{-15}$ for all contributing filter pairs except one). The difference in average IC is negligible (0.04 bit higher for 'max-contrib', Wilcoxon test, $P < 10^{-15}$). Therefore, our contribution logos represent analogous 'motifs', while extracting additional, nucleotide-level interpretations. For exactly one filter, 'max-contrib' and 'max-activation' scores are not correlated. A deeper analysis reveals that this particular filter is activated by stretches of 0s (*N*s)—it is the only filter with a positive bias, and almost all of its weights are negative (with one near-zero positive). Therefore, an overwhelming majority of its maximum activations are in fact padding artifacts. On the other hand, regions of unambiguous nucleotide sequences result in high positive contributions, since they correspond to a lack of filter activation, where an activation is present for the all-*N* reference. In fact, for over 99.9% of the reads, positive contributions occur at every single position. We suspect that the filter works as an 'ambiguity detector'. Since *N*s are modeled as all-zero vectors in the one-hot encoding scheme used here, the network represents 'meaningful' (i.e. unambiguous) regions of the input as a missing activation of the filter. This is supported by the fact that the filter lacks any further preference for the specific nonzero nucleotide type. Since sequence logos presented here ignore ambiguous (i.e. noninformative) nucleotides, their ICs for this filter are near-zero, preventing meaningful visualization. On the other hand, this ambiguity seems to play a role in the final classification decision, as contribution distributions are well-separated for both classes (Supplementary Figure S2). We speculate that this could be caused by lower quality of the nonpathogen reference genomes, but understanding how exactly this information is used would require further investigation, including feature interactions at all layers of the network. Importantly, only the contribution analysis reveals the relevance of the filter beyond simple activation and nucleotide over-representation. The choice of the reference input is crucial.

In the Figure 1 we present example filters, visualized as 'max-contrib' sequence logos based on mean partial Shapley values for each nucelotide at each position. All nucleotides of the filters with the second-highest (Figure 1A) and the lowest (Figure 1B) score have relatively strong contributions in accordance with the filters' own contributions. However, we observe that some nucleotides consistently appear in the activating subsequences, but the sign of their contributions is opposite to the filter's (low-IC nucleotides of a different color, Figure 1C). Those 'counter-contributions' may arise if a nucleotide with a negative weight forms a frequent motif with others with positive weights strong enough to activate the filter. We comment on this fact in 'Nucleotide contribution logos' section. Some filters seem to learn gapped motifs resembling a codon structure (Figure 1C). We extracted this filter from the original DeePaC network predicting bacterial pathogenicity (24) where the counter-contributions are common, but we find similar filters in our networks as well (Supplementary Figure S3). We scanned a genome of *S. aureus* subsp. *aureus* 21200 (RefSeq assembly accession: GCF_000221825.1) with this filter and discovered that the learned motif is indeed significantly enriched in coding sequences (Fisher exact test with Benjamini–Hochberg correction, $q < 10^{-15}$). It is also enriched in a number of specific genes. The one with the most hits (sraP, $q < 10^{-15}$) is a serine-rich adhesin involved in the pathogenesis of infective endocarditis and mediating binding to human platelets (65). The filter seems to detect serine and glycine repeats in this particular gene (Supplementary Figure S5), but a broader, cross-species, multi-gene analysis would be required to fully understand its activation patterns. An analogous analysis revealed that the second-highest contributing filter (Figure 1A) is overall enriched in coding sequences in both Taï Forest ebolavirus ($q < 10^{-15}$, RefSeq accession: NC_014372) and SARS-CoV-2 coronavirus ($q = 5.6 \times 10^{-5}$, RefSeq accession: NC_045512.2). The top hits are the nucleocapsid (N) protein gene of SARS-CoV-2 and the VP35 ebolavirus gene encoding a polymerase cofactor suppressing innate immune signaling ($q < 10^{-15}$).

Genome-wide phenotype analysis

We created a GWPA plot for the Taï Forest ebolavirus genome. Most genes (6 out of 7) can be detected with visual inspection by finding peaks of elevated infectious potential score predicted by at least one of the models (Figure 2A). Intergenic regions are characterized by lower mean scores. Noticeably, most nucleotide contributions are positive, and low non-negative contributions coincide with regions of negative predictions. Taken together with the surprisingly good generalization of Chordata-trained classifiers and a dominance of positive filters discussed above, this suggests that our networks work as positive class detectors, treating all other sequences as 'negative' by default. Indeed, the reference sequence of all *N*s is predicted to be 'nonpathogenic' with a score of 0.

We ran a similar analysis of *S. aureus* using the built-in DeePaC models (24) and our interpretation workflow. While a viral genome contains usually only a handful of genes, by compiling a ranking of 870 annotated genes of the analyzed *S. aureus* strain we could test if the high-ranking regions are indeed associated with pathogenicity (Supplementary Table S4). Indeed, out of three top-ranking genes with known biological names and Gene Ontology terms, sarR and sspB, are directly engaged in virulence, while hupB regulates expression of virulence-involved genes in many pathogens (66). In contrast to the viral models, both negative and positive contributions are present (Supplementary Figure S6), and the model's output for the all-*N* reference is slightly above the decision threshold (0.58). Even though the network architecture of the viral and the bacterial model are the same, the latter learns a 'two-sided' view of the data. We assume this must be a feature of the dataset itself.

Figure 2B presents a GWPA plot for the whole genome of the SARS-CoV-2 coronavirus, successfully predicted to infect humans, even though the data were collected at least 5 months before its emergence. Interestingly, its mean infectious potential (0.57 as scored by CNN_{All}) is relatively close to the decision threshold, while its closest known relative, a bat-infecting SARSr-CoV RaTG13, is actually falsely classified as a human virus with a slightly lower mean infectious potential (0.55). What is more, the gene encoding the spike protein, which plays a significant role in host entry (67), has a mean score slightly above the threshold for SARS-CoV-2 (0.52) and below the threshold for RaTG13 (0.49). As shown in the GWPA plots of both viruses (Figure 2B and Supplementary Figure S4), regions that the network has learned to associate with the infectious phenotype are distributed nonuniformly and tend to cluster together. This suggests that low-confidence mean prediction for those viruses is not a result of random guessing, but genuine ambiguity present in the data—and the misclassifica-

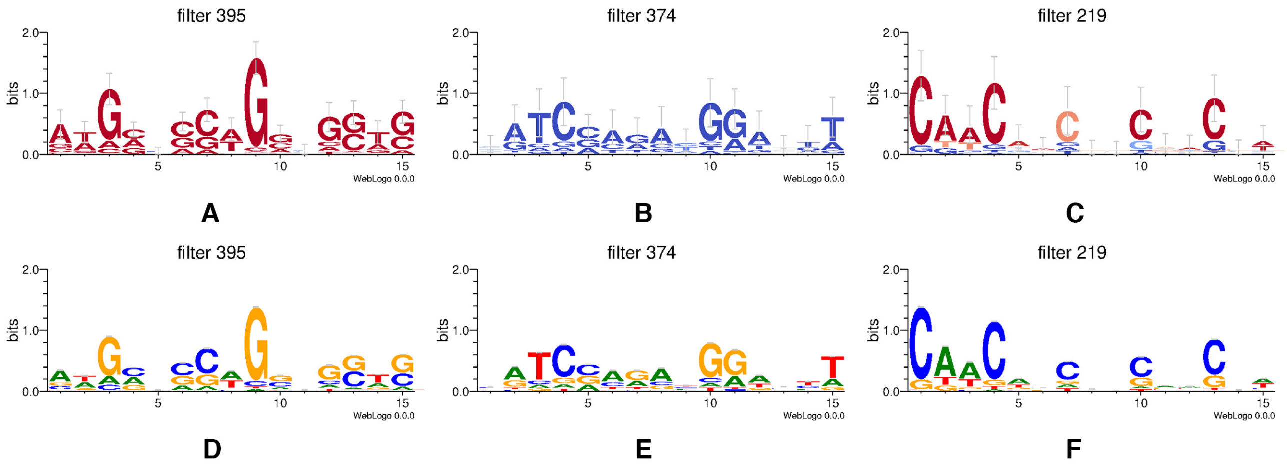

Figure 1. Nucleotide contribution logos of example filters. (**A**) Second-highest mean contribution score (CNN_{All}). Error bars correspond to Bayesian 95% confidence intervals. (**B**) Lowest mean contribution score (CNN_{All}). (**C**) Gaps resembling a codon structure, extracted from Bartoszewicz *et al.* (24). Consensus sequence: CAWCNNCNNCNNCNN. (**D–F**) Analogous logos created with the DeepBind-like 'max-activation' approach. Our 'max-contrib' logos visualize contributions of individual nucleotides, including counter-contributions.

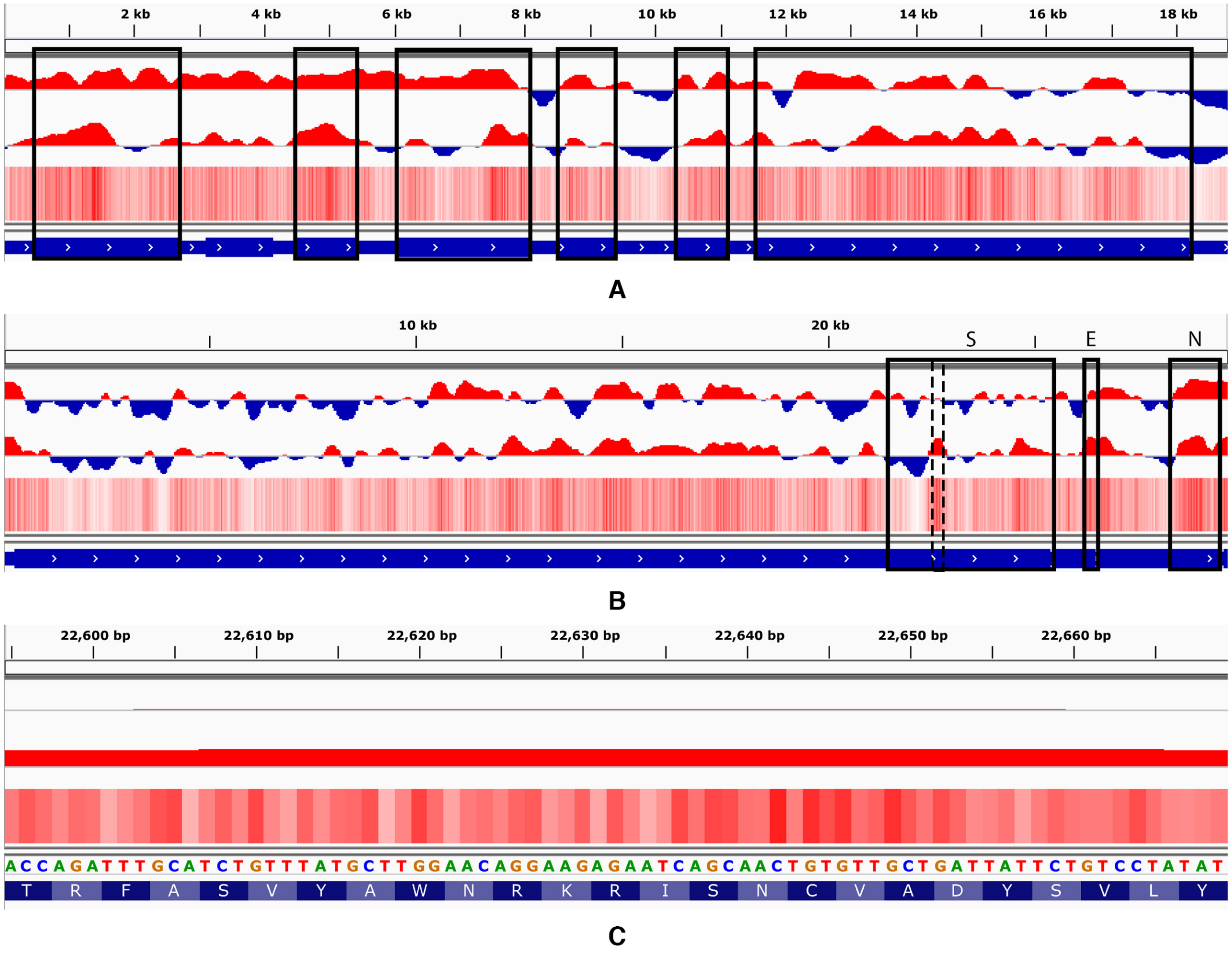

Figure 2. Taï Forest ebolavirus and SARS-CoV-2 coronavirus genomes. Top: score predicted by LSTM_{All}. Middle: score predicted by CNN_{All}. Heatmap: nucleotide contributions of CNN_{All}. Bottom, in blue: reference sequence. (**A**) Taï Forest ebolavirus. Genes that can be detected by at least one model are highlighted in black. (**B**) Whole genome and sequences encoding the spike protein (S), envelope protein (E) and nucleocapsid protein (N). (**C**) Spike protein gene, a small peak (positions 22 595–22 669, dashed line in Figure 2B) within the receptor-binding domain (predicted by CD-search, positions 22 517–23 185). Binding to the receptor is crucial for entry to the host cell. Local host adaptation could help switch hosts between the animal reservoir and humans.

tion of RaTG13 could be indicative of a general zoonotic potential of SARS-related coronaviruses. In the Figure 2B, we highlighted the score peaks aligning the spike protein gene (S), as well as the E and N genes, which were scored the highest (apart from an unconfirmed ORF10 of just 38aa downstream of N) by the CNN and the LSTM, respectively. Correlation between the CNN and LSTM outputs is significant, but species-dependent and moderate (0.28 for Ebola, 0.48 for SARS-CoV-2), which suggests they capture complementary signals.

Figure 2C shows the nucleotide-level contributions in a small peak within the receptor-binding domain (RBD) of the S protein, crucial for recognizing the host cell. The domain location was predicted with CD-search (68) using the default parameters. The maximum score of this peak is noticeably higher for SARS-CoV-2 (0.87) than for its analog in RaTG13 (0.67). Figure 3 presents the RBD in the structural context of the whole S protein (PDB ID: 6VSB, (69)), as well as in complex with a SARS-neutralizing antibody CR3022 (PDB ID: 6W41, (70)). The high score peak roughly corresponds to one of the regions associated with reduced expression of the RBD (71), located in the core-RBD subdomain. It covers over 71% of the CR3022 epitope, as well as the neighboring site of the N343 glycan. The latter is present in the epitope of another core-RBD targeting antibody, S309 (72). All the per-residue average contributions in the region are positive (Supplementary Figure S7), even in the regions of lower pathogenicity score, in accordance with the results presented in Figure 2C.

DISCUSSION

Accurate predictions from short DNA reads

Compared to the previous state-of-the-art in viral host prediction directly from next-generation sequencing reads (20), our models drastically reduce the error rates. This holds also for novel viruses not present in the training set. Generalization of virus-level Chordata models to other host groups is a sign of a strong, ‘human’ signal. We suspect our classifiers detect the positive class treating all other regions of the sequence space as ‘negative’ by default, exhibiting traits of a one-class classifier even without being explicitly trained to do so. We find further support for this hypothesis: the networks learn many more ‘positive’ than ‘negative’ filters and regions of near-zero nucleotide contributions (including the null reference sample) result in negative predictions. As this effect does not occur for bacteria, we expect it do be task- and data dependent. While we ignore the simulated quality information here, investigating the role of sequencing noise will be an interesting follow-up study. Although the data setup is crucial in general, the modeling step is also important, as shown by our comparison to the baseline k-NN model. The RC-nets are relatively simple, but they are invariant to reverse-complementarity and perform better than random forests, naïve Bayes classifiers and standard NN architectures in another NGS task (24).

In the paired read scenario, the previously described k-NN approach fails, and standard, alignment-based homology testing algorithms cannot find any matches in >10% of the cases, resulting in relatively low accuracy. On a real human virome sample, where a main source of negative class reads is most likely contamination (54), our method filters out nonhuman viruses with high specificity. In this scenario, the BLAST-derived ground-truth labels were mined using the complete database (as opposed to just a training set). In all cases, our results are only as good as the training data used; high quality labels and sequences are needed to develop trustworthy models. Ideally, sources of error should be investigated with an in-depth analysis of a model’s performance on multiple genomes covering a wide selection of taxonomic units. This is especially important as the method assumes no mechanistic link between an input sequence and the phenotype of interest, and the input sequence constitutes only a small fraction of the target genome without a wider biological context. Still, it is possible to predict a label even from those small, local fragments. A similar effect was also observed for image classification with CNNs (73). Virulence arises as a complex interplay between the host and the virus, so the predictions reflect only an estimated potential of the infectious phenotype. This mirrors the caveats of bacterial pathogenic potential prediction (24), including the considerations of balancing computational cost, reliability of error estimates, size and composition of the reference database. Even though deep learning outperforms the standard homology-based methods, it is still an open question whether it captures ‘functional’ signals, or just a more flexible sequence similarity function. By the very nature of machine learning and sequence comparison in general, we expect similar viruses to yield similar predictions; in principle this could be used to asses a risk of a host-switching event. The interpretability suite presented here aims at shedding some light on this question, but more research is needed.

Dual-use research and biosecurity

While we focused on the NGS-based prediction scenario, our models could in principle be used to screen DNA synthesis orders for potentially dangerous sequences the context of cyberbiosecurity in synthetic biology. Since standard, homology-based approaches like BLAST are not enough to guarantee accurate screening at a reasonable cost (74–76), machine learning methods are a promising solution. This has been suggested before for the bacterial DeePaC models (24), and is applicable to the viral networks presented here as well.

However, this line of research can raise questions about possible dual-use. O’Brien and Nelson (77) suggested that while the intended purpose of pathogenicity potential prediction is to mitigate biosecurity threats, it could actually enable designing new pathogens to cause maximal harm. The importance of this concern is difficult to overstate and it must be addressed. If an ML-guided, genome-wide phenotype optimization tool existed, it would indeed be a classical dual-use technology not unlike more established computer-aided design approaches for synthetic biology—potentially dangerous, but offering tremendous benefits (e.g. in agriculture, medicine or manufacturing) as well. However, the models presented here do not allow biologically sensible optimization of target sequences. For example, we find meaningless, low-complexity sequences of mononucleotide repeats corresponding to global maxima (infectious potential of 1.0). These artifacts highlight the fact that only some

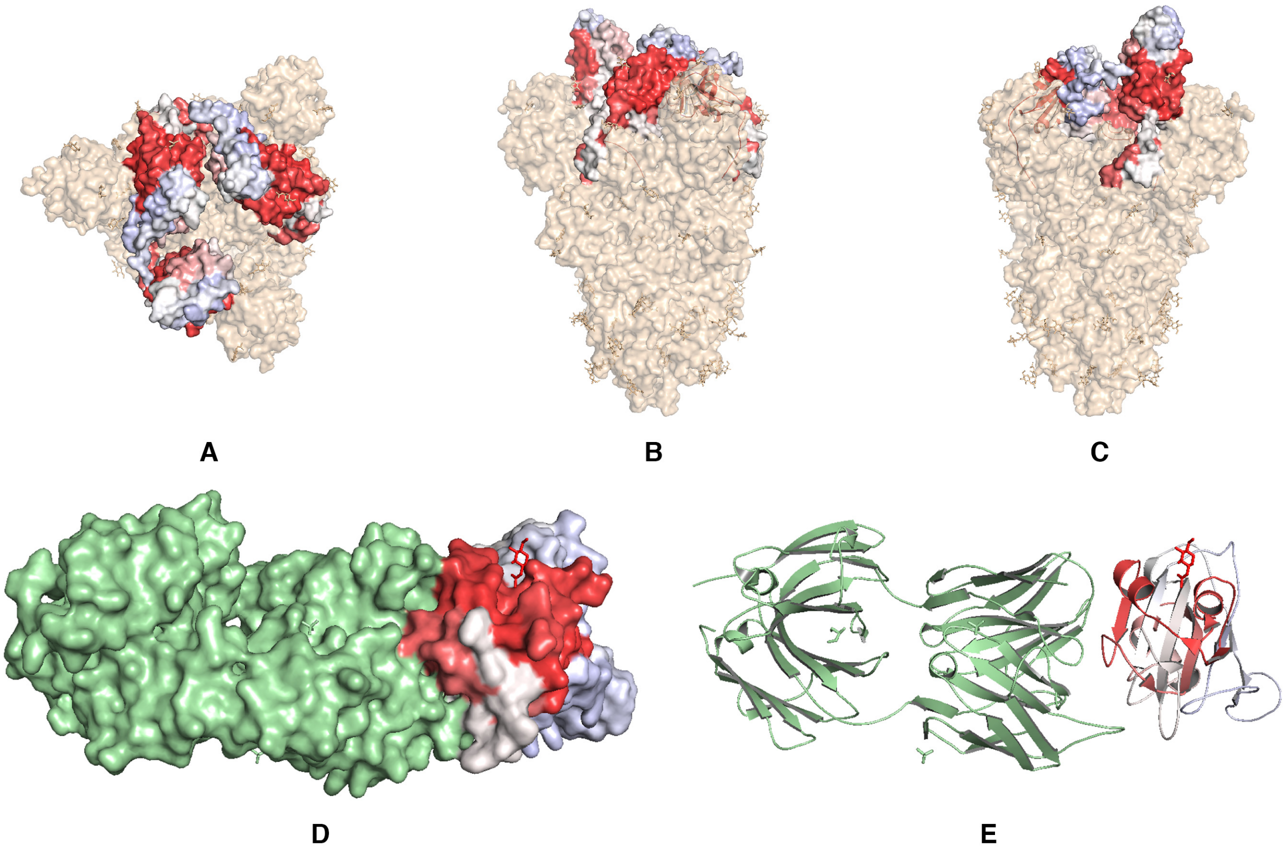

Figure 3. Predicted infectious potentials plotted over the SARS-CoV-2 spike glycoprotein receptor-binding domain. (**A**–**C**) Top and side view of the spike protein. Three receptor-binding domains (RBDs) are colored in blue, white and red according to the predicted infectious potential of the corresponding genomic sequence. One of the domains is in the 'up' conformation. Red regions corresponding to the peak in Figure 2C are located in the core-RBD subdomain. (**D**) RBD in complex with a SARS-neutralizing antibody CR3022 (green). The red region covers over 71% of the CR3022 epitope, but spans also to the neighboring fragments, including the site of the N343 glycan (carbohydrate in red stick representation). This is a part of the epitope of another neutralizing antibody, S309. (**E**) Cartoon representation of Figure 3D. The red region is centered on two exposed α-helices surrounding the core β-sheet (lower score, white).

generally undefined regions of the theoretically possible sequence space are biologically relevant. What is more, we operate on short sequences constituting minuscule fractions of the whole genome with all its complexity. Although successful deep learning approaches for both protein (78–80) and regulatory sequence design (81–84) do exist, moving from read-based classification to genome-wide phenotype optimization would require considerable research effort, if possible at all. This would entail capturing a wealth of biological contexts well beyond the capabilities of even the best classification models currently available.

Nucleotide contribution logos

Visualizing convolutional filters may help to identify more complex filter structures and disentangle the contributions of individual nucleotides from their 'conservation' in contributing sequences. Counter-contributions suggest that the information content and the contribution of a nucleotide are not necessarily correlated. Visualizing learned motifs by aligning the activating sequences (25) would not fully describe how the filter reacts to presented data. It seems that the assumption of nucleotide independence—which is crucial for treating DeepLIFT as a method of estimating Shapley values for input nucleotides (44)—does not hold in full. Indeed, *k*-mer distribution profiles are frequently used features for modeling DNA sequences (as shown also by the dimer-shuffling method of generating reference sequences proposed by Shrikumar *et al*. (42)). However, DeepLIFT's multiple successful applications in genomics indicate that the assumption probably holds approximately. We see information content and DeepLIFT's contribution values as two complementary channels that can be jointly visualized for better interpretability and explainability of CNNs in genomics. Filter enrichment analysis enables even deeper insight in the inner workings of the networks. We generate activation data for hundreds to thousands of species, genes and filters. Yet, aggregation and interpretation of those results beyond case studies is nontrivial, and a promising avenue for further research.

Genome-scale interpretability

Mapping predictions back to a target genome can be used both as a way of investigating a given model's performance and as a method of genome analysis. GWPA plots of well-annotated genomes highlight the sequences with erroneous and correct phenotype predictions at both genome and gene level, and nucleotide-resolution contribution maps help track those regions down to individual amino acids. On the other hand, once a trusted model is developed, it can be used on newly emerging pathogens, as the SARS-CoV-2 virus briefly analyzed in this work. Therefore, we see GPWA applications in both probing the behavior of artificial neural networks in pathogen genomics and finding regions of interest in weakly annotated genomes. What is more, the approach could be easily co-opted to genome-wide activation analyses of any arbitrary, intermediate neuron. The methods presented here may also be applied to other biological problems, and extending them to other hosts and pathogen groups, multi-class classification or gene identification is possible. However, experimental work and traditional sequence analysis are required to truly understand the biology behind host adaptation and distinguish true hits from false positives.

Conclusion

We presented a new approach for predicting a host of a novel virus based on a single DNA read or a read pair, cutting the error rates in half compared to the previous state-of-the-art. For convolutional filters, we jointly visualize nucleotide contributions and information content. Finally, we use GWPA plots to gain insights into the models' behavior and analyze a recently emerged SARS-CoV-2 virus. The approach presented here is implemented as a python package (see Data Availability) and a command line tool easily installable with Bioconda (85).

DATA AVAILABILITY

The datasets of simulated reads with associated metadata are hosted at https://doi.org/10.5281/zenodo.4312525. The tool can be installed with Bioconda (conda install deepacvir, requires setting up Bioconda), Docker (docker pull dacshpi/deepac) or pip (pip install deepacvir). Detailed installation instructions, user guide and the main codebase (including the interpretability workflows presented here) are available at https://gitlab.com/dacs-hpi/DeePaC. Source code of the plugin shipping the trained models, config files describing the architectures used and the models themselves are available at https://gitlab.com/dacs-hpi/DeePaC-vir.

SUPPLEMENTARY DATA

Supplementary Data are available at NARGAB Online.

ACKNOWLEDGEMENTS

We gratefully acknowledge Yong-Zhen Zhang and the scientists at the Shanghai Public Health Clinical Center and School of Public Health, Fudan University, who shared the sequence of the SARS-CoV-2 virus ahead of publication. We thank Melania Nowicka (Max Plank Institute for Molecular Genetics) for inspiring discussions on efficient calculations of partial Shapley values, Vitor C. Piro (Hasso Plattner Institute) for discussions on traversing taxonomy graphs, Lothar H. Wieler (Robert Koch Institute) for useful comments on the first draft of the manuscript, and the Anonymous Reviewers for their suggestions and feedback.

FUNDING

German Academic Scholarship Foundation (to J.M.B.); Federal Ministry of Education and Research (BMBF) Computational Life Sciences Initiative (project DeepPath, to B.Y.R.); the BMBF-funded de.NBI Cloud within the German Network for Bioinformatics Infrastructure (de.NBI) [031A537B, 031A533A, 031A538A, 031A533B, 031A535A, 031A537C, 031A534A, 031A532B].

Conflict of interest statement. None declared.

REFERENCES

1. Calvignac-Spencer,S., Schulze,J.M., Zickmann,F. and Renard,B.Y. (2014) Clock rooting further demonstrates that Guinea 2014 EBOV is a member of the Zaïre lineage. *PLoS Curr.*, **6**, doi:10.1371/currents.outbreaks.c0e035c86d721668a6ad7353f7f6fe86.
2. Vouga,M. and Greub,G. (2016) Emerging bacterial pathogens: the past and beyond. *Clin. Microbiol. Infec.*, **22**, 12–21.
3. Trappe,K., Marschall,T. and Renard,B.Y. (2016) Detecting horizontal gene transfer by mapping sequencing reads across species boundaries. *Bioinformatics*, **32**, i595–i604.
4. Leendertz,S. A.J., Gogarten,J.F., Düx,A., Calvignac-Spencer,S. and Leendertz,F.H. (2016) Assessing the evidence supporting fruit bats as the primary reservoirs for ebola viruses. *EcoHealth*, **13**, 18–25.
5. Lecuit,M. and Eloit,M. (2014) The diagnosis of infectious diseases by whole genome next generation sequencing: a new era is opening. *Front. Cell. Infect. Mi.*, **4**, 25.
6. Calistri,A. and Palù,G. (2015) Editorial commentary: Unbiased next-generation sequencing and new pathogen discovery: undeniable advantages and still-existing drawbacks. *Clini. Infect. Dis*, **60**, 889–891.
7. Andrusch,A., Dabrowski,P.W., Klenner,J., Tausch,S.H., Kohl,C., Osman,A.A., Renard,B.Y. and Nitsche,A. (2018) PAIPline: pathogen identification in metagenomic and clinical next generation sequencing samples. *Bioinformatics*, **34**, i715–i721.
8. Herfst,S., Schrauwen,E.J.A., Linster,M., Chutinimitkul,S., Wit,E.d., Munster,V.J., Sorrell,E.M., Bestebroer,T.M., Burke,D.F., Smith,D.J. *et al.* (2012) Airborne transmission of influenza A/H5N1 virus between ferrets. *Science*, **336**, 1534–1541.
9. Imai,M., Watanabe,T., Hatta,M., Das,S.C., Ozawa,M., Shinya,K., Zhong,G., Hanson,A., Katsura,H., Watanabe,S. *et al.* (2012) Experimental adaptation of an influenza H5 HA confers respiratory droplet transmission to a reassortant H5 HA/H1N1 virus in ferrets. *Nature*, **486**, 420–428.
10. Lipsitch,M. and Inglesby,T.V. (2014) Moratorium on research intended to create novel potential pandemic pathogens. *mBio*, **5**, e02366-14.
11. Noyce,R.S., Lederman,S. and Evans,D.H. (2018) Construction of an infectious horsepox virus vaccine from chemically synthesized DNA fragments. *PLOS ONE*, **13**, e0188453.
12. Thiel,V. (2018) Synthetic viruses-Anything new? *PLOS Pathog.*, **14**, e1007019.
13. Edwards,R.A., McNair,K., Faust,K., Raes,J. and Dutilh,B.E. (2016) Computational approaches to predict bacteriophage-host relationships. *FEMS Microbiol. Rev.*, **40**, 258–272.
14. Eng,C.L., Tong,J.C. and Tan,T.W. (2014) Predicting host tropism of influenza A virus proteins using random forest. *BMC Med. Genomics*, **7**, S1.
15. Xu,B., Tan,Z., Li,K., Jiang,T. and Peng,Y. (2017) Predicting the host of influenza viruses based on the word vector. *PeerJ*, **5**, e3579.
16. Li,H. and Sun,F. (2018) Comparative studies of alignment, alignment-free and SVM based approaches for predicting the hosts of viruses based on viral sequences. *Sci. Rep.*, **8**, 10032.
17. Mock,F., Viehweger,A., Barth,E. and Marz,M. (2020) VIDHOP, viral host prediction with Deep Learning. *Bioinformatics*, btaa705.
18. Gałan,W., Bąk,M. and Jakubowska,M. (2019) Host taxon Predictor - A tool for predicting taxon of the host of a newly discovered virus. *Sci. Rep.*, **9**, 3436.
19. Babayan,S.A., Orton,R.J. and Streicker,D.G. (2018) Predicting reservoir hosts and arthropod vectors from evolutionary signatures in RNA virus genomes. *Science*, **362**, 577–580.

20. Zhang,Z., Cai,Z., Tan,Z., Lu,C., Jiang,T., Zhang,G. and Peng,Y. (2019) Rapid identification of human-infecting viruses. *Transbound. Emerg. Dis.*, **66**, 2517–2522.
21. Poplin,R., Chang,P.-C., Alexander,D., Schwartz,S., Colthurst,T., Ku,A., Newburger,D., Dijamco,J., Nguyen,N., Afshar,P.T. *et al.* (2018) A universal SNP and small-indel variant caller using deep neural networks. *Nat. Biotechnol.*, **36**, 983–987.
22. Rizzo,R., Fiannaca,A., La Rosa,M. and Urso,A. (2016) Classification Experiments of DNA Sequences by Using a Deep Neural Network and Chaos Game Representation. In: *Proceedings of the 17th International Conference on Computer Systems and Technologies 2016 New York*. Association for Computing Machinery CompSysTech '16, NY, pp. 222–228.
23. Löchel,H.F., Eger,D., Sperlea,T. and Heider,D. (2020) Deep learning on chaos game representation for proteins. *Bioinformatics*, **36**, 272–279.
24. Bartoszewicz,J.M., Seidel,A., Rentzsch,R. and Renard,B.Y. (2019) DeePaC: predicting pathogenic potential of novel DNA with reverse-complement neural networks. *Bioinformatics*, **36**, 81–89.
25. Alipanahi,B., Delong,A., Weirauch,M.T. and Frey,B.J. (2015) Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. *Nat. Biotechnol.*, **33**, 831–838.
26. Zhou,J. and Troyanskaya,O.G. (2015) Predicting effects of noncoding variants with deep learning–based sequence model. *Nat. Methods*, **12**, 931–934.
27. Zeng,H., Edwards,M.D., Liu,G. and Gifford,D.K. (2016) Convolutional neural network architectures for predicting DNA–protein binding. *Bioinformatics*, **32**, i121–i127.
28. Quang,D. and Xie,X. (2016) DanQ: a hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. *Nucleic Acids Res.*, **44**, e107.
29. Kelley,D.R., Snoek,J. and Rinn,J.L. (2016) Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome Res.*, **26**, 990–999.
30. Greenside,P., Shimko,T., Fordyce,P. and Kundaje,A. (2018) Discovering epistatic feature interactions from neural network models of regulatory DNA sequences. *Bioinformatics*, **34**, i629–i637.
31. Nair,S., Kim,D.S., Perricone,J. and Kundaje,A. (2019) Integrating regulatory DNA sequence and gene expression to predict genome-wide chromatin accessibility across cellular contexts. *Bioinformatics*, **35**, i108–i116.
32. Avsec,Ž., Weilert,M., Shrikumar,A., Alexandari,A., Krueger,S., Dalal,K., Fropf,R., McAnany,C., Gagneur,J., Kundaje,A. *et al.* (2019) Deep learning at base-resolution reveals motif syntax of the cis-regulatory code. bioRxiv doi: https://doi.org/10.1101/737981, 21 August 2019, preprint: not peer reviewed.
33. Ren,J., Song,K., Deng,C., Ahlgren,N.A., Fuhrman,J.A., Li,Y., Xie,X., Poplin,R. and Sun,F. (2020) Identifying viruses from metagenomic data by deep learning. *Quantitative Biology*, **8**, 64–77.
34. Tampuu,A., Bzhalava,Z., Dillner,J. and Vicente,R. (2019) ViraMiner: Deep learning on raw DNA sequences for identifying viral genomes in human samples. *PLOS ONE*, **14**, e0222271.
35. Eraslan,G., Avsec,Ž., Gagneur,J. and Theis,F.J. (2019) Deep learning: new computational modelling techniques for genomics. *Nat. Rev. Genet.*, **20**, 389–403.
36. Schneider,T.D. and Stephens,R.M. (1990) Sequence logos: a new way to display consensus sequences. *Nucleic Acids Res.*, **18**, 6097–6100.
37. Crooks,G.E., Hon,G., Chandonia,J.-M. and Brenner,S.E. (2004) WebLogo: a sequence logo generator. *Genome Res.*, **14**, 1188–1190.
38. Lanchantin,J., Singh,R., Lin,Z. and Qi,Y. (2016) Deep Motif: Visualizing genomic sequence classifications. arXiv doi: https://arxiv.org/abs/1605.01133, 02 Jun22 2016, preprint: not peer reviewed.
39. Lanchantin,J., Singh,R., Wang,B. and Qi,Y. (2017) Deep motif dashboard: visualizing and understanding genomic sequences using deep neural networks. *Pacific Symp. Biocomput.*, **22**, 254–265.
40. Sundararajan,M., Taly,A. and Yan,Q. (2016) Gradients of Counterfactuals. arXiv doi: https://arxiv.org/abs/1611.02639, 15 November 2016, preprint: not peer reviewed.
41. Jha,A., Aicher,J.K., R. Gazzara,M., Singh,D. and Barash,Y. (2020) Enhanced Integrated Gradients: improving interpretability of deep learning models using splicing codes as a case study. *Genome Biol.*, **21**, 149.
42. Shrikumar,A., Greenside,P. and Kundaje,A. (2017) Learning Important Features Through Propagating Activation Differences. In: Precup,D. and Teh,Y.W. (eds). *Proceedings of the 34th International Conference on Machine Learning, International Convention Centre*. PMLR Vol.70 of Proceedings of Machine Learning Research, Sydney, pp. 3145–3153.
43. Bach,S., Binder,A., Montavon,G., Klauschen,F., Müller,K.-R. and Samek,W. (2015) On Pixel-Wise explanations for Non-Linear classifier decisions by Layer-Wise relevance propagation. *PLOS ONE*, **10**, e0130140.
44. Lundberg,S.M. and Lee,S.-I. (2017) A Unified Approach to Interpreting Model Predictions. In: Guyon,I., Luxburg,U.V., Bengio,S., Wallach,H., Fergus,R., Vishwanathan,S. and Garnett,R. (eds). *Advances in Neural Information Processing Systems 30*. Curran Associates, Inc., pp. 4765–4774.
45. Shrikumar,A., Tian,K., Shcherbina,A., Avsec,Ž., Banerjee,A., Sharmin,M., Nair,S. and Kundaje,A. (2020) Technical Note on Transcription Factor Motif Discovery from Importance Scores (TF-MoDISco) version 0.5.6.5. arXiv doi: https://arxiv.org/abs/1811.00416, 30 April 2020,preprint: not peer reviewed.
46. Altschul,S.F., Gish,W., Miller,W., Myers,E.W. and Lipman,D.J. (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.
47. Camacho,C., Coulouris,G., Avagyan,V., Ma,N., Papadopoulos,J., Bealer,K. and Madden,T.L. (2009) BLAST+: architecture and applications. *BMC Bioinformatics*, **10**, 421.
48. Wu,F., Zhao,S., Yu,B., Chen,Y.-M., Wang,W., Song,Z.-G., Hu,Y., Tao,Z.-W., Tian,J.-H., Pei,Y.-Y. *et al.* (2020) A new coronavirus associated with human respiratory disease in China. *Nature*, **579**, 265–269.
49. Mihara,T., Nishimura,Y., Shimizu,Y., Nishiyama,H., Yoshikawa,G., Uehara,H., Hingamp,P., Goto,S. and Ogata,H. (2016) Linking virus genomes with host taxonomy. *Viruses*, **8**, 66.
50. King,A.M.Q., Adams,M.J., Carstens,E.B. and Lefkowitz,E.J. (eds.) (2012) In: *Virus Taxonomy: Ninth Report of the International Committee on Taxonomy of Viruses*. Academic Press, London; Waltham.
51. Lefkowitz,E.J., Dempsey,D.M., Hendrickson,R.C., Orton,R.J., Siddell,S.G. and Smith,D.B. (2018) Virus taxonomy: the database of the International Committee on Taxonomy of Viruses (ICTV). *Nucleic Acids Res.*, **46**, D708–D717.
52. Holtgrewe,M. (2010) Mason – A Read Simulator for Second Generation Sequencing Data. Technical Report, FU Berlin.
53. Deneke,C., Rentzsch,R. and Renard,B.Y. (2017) PaPrBaG: A machine learning approach for the detection of novel pathogens from NGS data. *Sci. Rep.*, **7**, 39194.
54. Moustafa,A., Xie,C., Kirkness,E., Biggs,W., Wong,E., Turpaz,Y., Bloom,K., Delwart,E., Nelson,K.E., Venter,J.C. *et al.* (2017) The blood DNA virome in 8,000 humans. *PLOS Pathog.*, **13**, e1006292.
55. Gorbalenya,A.E., Baker,S.C., Baric,R.S., de Groot,R.J., Drosten,C., Gulyaeva,A.A., Haagmans,B.L., Lauber,C., Leontovich,A.M., Neuman,B.W. *et al.* (2020) The species Severe acute respiratory syndrome-related coronavirus : classifying 2019-nCoV and naming it SARS-CoV-2. *Nat. Microbiol.*, **5**, 536–544.
56. Simmonds,P. and Aiewsakun,P. (2018) Virus classification – where do you draw the line? *Arch. Virol.*, **163**, 2037–2046.
57. Van Regenmortel,M.H.V. (2018) Chapter One - The Species Problem in Virology. In: Kielian,M., Mettenleiter,T.C. and Roossinck,M.J. (eds). *Advances in Virus Research*. Academic Press, Vol. **100**, pp. 1–18.
58. Li,H. and Durbin,R. (2009) Fast and accurate short read alignment with Burrows–Wheeler transform. *Bioinformatics*, **25**, 1754–1760.
59. Langmead,B. and Salzberg,S.L. (2012) Fast gapped-read alignment with Bowtie 2. *Nat. Methods*, **9**, 357–359.
60. Wood,D.E. and Salzberg,S.L. (2014) Kraken: ultrafast metagenomic sequence classification using exact alignments. *Genome Biol.*, **15**, R46.
61. Nix,R. and Kantarciouglu,M. (2012) Incentive Compatible Privacy-Preserving Distributed Classification. *IEEE Trans. Depend. Secure Comput.*, **9**, 451–462.
62. Matejczyk,S. and Michalak,T. (2015) Solving Influence Maximization Problem UsingMethods from Cooperative Game Theory, In: *ITRIA 2015. Selected Problems in Information Technologies (Conference Proceedings)*. Institute of Computer Science PAS, Warsaw, pp. 95–117.

63. Thorvaldsdóttir,H., Robinson,J.T. and Mesirov,J.P. (2013) Integrative Genomics Viewer (IGV): high-performance genomics data visualization and exploration. *Brief. Bioinform.*, **14**, 178–192.
64. DeLano,W.L. and others (2002) Pymol: An open-source molecular graphics tool. *CCP4 Newsletter Protein Crystallogr.*, **40**, 82–92.
65. Yang,Y.-H., Jiang,Y.-L., Zhang,J., Wang,L., Bai,X.-H., Zhang,S.-J., Ren,Y.-M., Li,N., Zhang,Y.-H., Zhang,Z. *et al.* (2014) Structural insights into SraP-Mediated staphylococcus aureus adhesion to host cells. *PLOS Pathog.*, **10**, e1004169.
66. Stojkova,P., Spidlova,P. and Stulik,J. (2019) Nucleoid-Associated Protein HU: A Lilliputian in Gene Regulation of Bacterial Virulence. *Front. Cell. Infect. Mi.*, **9**, 159.
67. Li,F. (2016) Structure, function, and evolution of coronavirus spike proteins. *Ann. Rev. Virol.*, **3**, 237–261.
68. Marchler-Bauer,A., Bo,Y., Han,L., He,J., Lanczycki,C.J., Lu,S., Chitsaz,F., Derbyshire,M.K., Geer,R.C., Gonzales,N.R. *et al.* (2017) CDD/SPARCLE: functional classification of proteins via subfamily domain architectures. *Nucleic Acids Res.*, **45**, D200–D203.
69. Wrapp,D., Wang,N., Corbett,K.S., Goldsmith,J.A., Hsieh,C.-L., Abiona,O., Graham,B.S. and McLellan,J.S. (2020) Cryo-EM structure of the 2019-nCoV spike in the prefusion conformation. *Science*, **367**, 1260–1263.
70. Yuan,M., Wu,N.C., Zhu,X., Lee,C.-C.D., So,R.T.Y., Lv,H., Mok,C. K.P. and Wilson,I.A. (2020) A highly conserved cryptic epitope in the receptor binding domains of SARS-CoV-2 and SARS-CoV. *Science*, **368**, 630–633.
71. Starr,T.N., Greaney,A.J., Hilton,S.K., Ellis,D., Crawford,K.H., Dingens,A.S., Navarro,M.J., Bowen,J.E., Tortorici,M.A., Walls,A.C., Veesler,D. and Bloom,J.D. (2020) Deep Mutational Scanning of SARS-CoV-2 Receptor Binding Domain Reveals Constraints on Folding and ACE2 Binding. *Cell*, **182**, 1295–1310.
72. Pinto,D., Park,Y.-J., Beltramello,M., Walls,A.C., Tortorici,M.A., Bianchi,S., Jaconi,S., Culap,K., Zatta,F., De Marco,A., Peter,A. *et al.* (2020) Cross-neutralization of SARS-CoV-2 by a human monoclonal SARS-CoV antibody. *Nature*, **583**, 290–295.
73. Brendel,W. and Bethge,M. (2019) Approximating CNNs with Bag-of-local-Features models works surprisingly well on ImageNet. In: *International Conference on Learning Representations*. New Orleans, LA.
74. National Research Council (2010) In: *Sequence-Based Classification of Select Agents: A Brighter Line*. The National Academies Press, Washington, DC.
75. National Academies of Sciences, Engineering, and Medicine (2018) In: *Biodefense in the Age of Synthetic Biology*. The National Academies Press, Washington, DC.
76. Diggans,J. and Leproust,E. (2019) Next Steps for Access to Safe, Secure DNA Synthesis. *Front. Bioengin. Biotechnol.*, **7**, 86.
77. O'Brien,J.T. and Nelson,C. (2020) Assessing the Risks Posed by the Convergence of Artificial Intelligence and Biotechnology. *Health Secur.*, **18**, 219–227.
78. Brookes,D., Park,H. and Listgarten,J. (2019) Conditioning by adaptive sampling for robust design. In: *International Conference on Machine Learning* pp. 773–782.
79. Alley,E.C., Khimulya,G., Biswas,S., AlQuraishi,M. and Church,G.M. (2019) Unified rational protein engineering with sequence-based deep representation learning. *Nat. Methods*, **16**, 1315–1322.
80. Biswas,S., Khimulya,G., Alley,E.C., Esvelt,K.M. and Church,G.M. (2020) Low-N protein engineering with data-efficient deep learning. bioRxiv doi: https://doi.org/10.1101/2020.01.23.917682, 24 January 2020, preprint: not peer reviewed.
81. Gupta,A. and Zou,J. (2019) Feedback GAN for DNA optimizes protein functions. *Nat. Machine Intel.*, **1**, 105–111.
82. Gupta,A. and Kundaje,A. (2019) Targeted optimization of regulatory DNA sequences with neural editing architectures. bioRxiv doi: https://doi.org/10.1101/714402, 28 July 2019, preprint: not peer reviewed.
83. Linder,J., Bogard,N., Rosenberg,A.B. and Seelig,G. (2019) Deep exploration networks for rapid engineering of functional DNA sequences. bioRxiv doi: https://doi.org/10.1101/864363, 04 December 2019, preprint: not peer reviewed.
84. Schreiber,J., Lu,Y.Y. and Noble,W.S. (2020) Ledidi: Designing genomic edits that induce functional activity. bioRxiv doi: https://doi.org/10.1101/2020.05.21.109686, 25 May 2020, preprint: not peer reviewed.
85. Grüning,B., Dale,R., Sjödin,A., Chapman,B.A., Rowe,J., Tomkins-Tinch,C.H., Valieris,R. and Köster,J. (2018) Bioconda: sustainable and comprehensive software distribution for the life sciences. *Nat. Methods*, **15**, 475–476.