Deep learning approaches for predicting pathogenic potentials of novel DNA and RNA sequences

Dissertation zur Erlangung des Grades
eines Doktors der Naturwissenschaften (Dr. rer. nat.)
am Fachbereich Mathematik und Informatik
der Freien Universität Berlin

vorgelegt von

JAKUB MACIEJ BARTOSZEWICZ

Berlin 2022

Betreuer:	Prof. Dr. Bernhard Renard
Erstgutachter:	Prof. Dr. Bernhard Renard
Zweitgutachterin:	Prof. Dr. Manja Marz
Tag der Disputation:	30.06.2022

Abstract

Regular emergence of novel pathogens is one of the greatest threats to global health. DNA and RNA sequencing enable detection of new viruses and microbes, but standard approaches for computational analysis of sequencing data rely on predefined lists of known agents. New pathogens, with genomes highly divergent from available references, remain difficult to recognize.

This problem can be alleviated by training classifiers predicting whether a given sequencing read originates from a possibly novel pathogen. I show that deep neural networks invariant to DNA reverse-complementarity markedly outperform alternatives based on other machine learning algorithms and homology detection by sequence alignment. This holds for both bacteria and viruses. I introduce new methods enabling analysis and visualization of the learned patterns, as well as identification of sequences, genes and genomic regions associated with high pathogenic potential. Modified ResNet architectures combined with real-time mapping of short reads can accurately recognize both known and novel threats as the sequencer is running. Analogous models also work for short fragments of long reads, corresponding to just 0.5 s of sequencing time. I then describe a manually curated database of fungal pathogen genomes facilitating detection of novel threats with both machine learning and alternative approaches. I use learned numerical representations of the genomes in the database to visualize the relationship between taxonomy and the pathogenic phenotype. Finally, I employ the developed neural architectures to classify reads sampled from mixtures of different novel bacteria, viruses, and fungi.

The methods presented here are implemented in the DeePaC and DeePaC-Live packages. They can be easily reused for training, evaluation, and deployment of deep neural networks for DNA and RNA sequences. Although the main focus is placed on identification of emerging pathogens from sequencing data, presented approaches could also be used to screen synthetic sequences and detect engineered threats. The trained networks are capable of predicting abstract, complex traits directly from sequences, without directly relying on close taxonomic matches. In the future, similar 'phenotype models' could find many alternative applications in rapid diagnostics, public health and synthetic biology.

Acknowledgements

First, I would like to thank my advisor Bernhard Renard. I am grateful for all the never-ending support, freedom and trust, for the invitation to work on fascinating, challenging problems, and for creating an environment that made solving them just a matter of time. I could not have wished for a better mentor.

Second, but by no means less important, I thank Melania Nowicka, my partner, friend, and a work-from-home office mate. Thank you for starting the PhD journey together, sharing every step of it, your understanding, support, hours of scientific and non-scientific discussions, celebrating the good, and persevering through the bad. Thank you for your love, and for being there.

I am also grateful to all other members of the Renard lab, also known as NG4, MF1, and finally DACS. In particular, I thank Robert Rentzsch for his support when I was preparing the initial outline of the project; Carlus Deneke for his insights regarding PaPrBaG, a predecessor of the methods presented here; Anja Seidel, Ninon de Meqcuenem, Ulrich Genske and Ferdous Nasri, who joined me in exploring many different aspects and applications of this work as students; Tobias Loka, who is a great table football teacher, for his help whenever I struggled with the language barrier or asked a millionth question; Thilo Muth for all the support of a senior colleague; Vitor Piro, Andreas Andrusch, Christine Matzinger, Elizabeth Yuu, Marta Lemanczyk, Alice Wittig, Jens-Uwe Ulrich, Tom Altenburg, Henri Knobloch, Christoph Schlaffner, Katharina Baum, and everyone else in the group for breakfasts, lunches, coffee breaks and after-work beers, with all the inspiring conversations on everything from science, through cultural exchange, to metal music or cats. The only reason I cannot explicitly name everyone here is the sheer length of the list of great people who built this exceptional team. Further, I would like to thank Prof. Annalisa Marsico, Stefan Budach, and Prof. Lothar H. Wieler for their helpful feedback.

I thank Prof. Heike Siebert and Prof. Tim Conrad for helping me find research opportunities in Berlin, and all reviewers who read my scholarship applications for the trust that they placed in me.

Finally, I am immensely grateful to my parents, Beata and Przemysław, as well as my entire family, especially my grandparents Maria, Irena and Jan, and my great-grandmother Regina. Thank you for your love, sacrifices, encouragement, and for supporting my every educational endeavour, from learning the first words to the doctorate.

Contents

List of Abbreviations

AUPR area under the precision-recall curve

AUC area under the ROC curve

bp base pair(s)

CGR chaos game representation

CNN convolutional neural network

CPU central processing unit

DNA deoxyribonucleic acid

EHEC enterohaemorrhagic *Escherichia coli*

EID emerging infectious disease

FCGR frequency matrix CGR

FPGA field-programmable gate array

GoF gain-of-function

GPU graphics processing unit

HIV human immunodeficiency virus

i.i.d. independent and identically distributed

kb kilobase pair

LCA lowest common ancestor

LRP layer-wise relevance propagation

LSTM long short-term memory network

kNN k-nearest neighbours

PR curve precision-recall curve

NGS next-generation sequencing

ONT Oxford Nanopore Technologies

RAM random access memory

RNA ribonucleic acid

ROC receiver operating characteristic

SFTP Secure File Transfer Protocol

SNP single-nucleotide polymorphism

SVM support-vector machine

WGS whole-genome sequencing

1 Introduction

1.1 Novel pathogens: a global issue

Possible emergence of novel, deadly pathogens has long been considered a realistic, serious and growing threat (Bryan et al., 1994; Lederberg et al., 1992; Morens & Fauci, 2020; Ndow et al., 2019; L. H. Taylor et al., 2001; M. E. J. Woolhouse et al., 2001; M. Woolhouse & Gaunt, 2007; M. E. J. Woolhouse & Gowtage-Sequeria, 2005). The frequency at which new infectious diseases emerge has been rising since the 1940s, even when reporting bias is taken into account (K. E. Jones et al., 2008). Precise definitions of pathogen emergence (or re-emergence) are often difficult to apply due to lack of sufficient historical data, and a certain degree of subjectivity remains (M. E. J. Woolhouse & Gowtage-Sequeria, 2005). Nevertheless, zoonotic diseases originating from wildlife have been identified as the greatest threat of all emerging infectious diseases (EIDs), with a potential to affect public health at a planetary scale (K. E. Jones et al., 2008).

RNA viruses are reported to be the most numerous group of pathogens appearing in new human populations, switching hosts or rapidly growing in incidence, most probably due to their high mutation rates (M. E. J. Woolhouse & Gowtage-Sequeria, 2005; M. E. J. Woolhouse et al., 2005). This has led to predictions that while, in general, the future emergence of new pathogens is expected and most probably just a matter of time, novel, zoonotic RNA viruses are the most likely to cause dangerous outbreaks or even pandemics (M. Woolhouse & Gaunt, 2007). There are multiple examples of such events occurring in the past; the global epidemic of human immunodeficiency virus (HIV) has claimed millions of lives since the virus was first identified. The pandemics caused by novel strains of influenza A in the 20th century (especially the 'Spanish' flu pandemic of 1918) were comparably deadly, and the first two decades of the 21st century were marked by the 'swine flu' pandemic and the outbreaks of novel SARS-CoV-1 and MERS coronaviruses. At the same time, epidemics of Zika and Ebola viruses in the 2010s have shown that previously known threats are still difficult to contain. Respiratory viruses such as influenza or coronaviruses have been suggested to be especially likely to cause new pandemics (Nuzzo et al., 2019). Those predictions came true during my work on

this thesis, when the SARS-CoV-2 virus emerged, causing the COVID-19 pandemic. The virus has quickly adapted, and the World Health Organization has already designated five Variants of Concern (Alpha, Beta, Gamma, Delta and Omicron) by the end of November 2021. Their increased transmissibility, virulence, or resistance towards the preventive, diagnostic or therapeutic measures introduced to contain the virus emphasize how fast pathogen evolution can be. Novel agents are bound to arise – either in the form of new strains of previously known threats or as members of entirely new taxa.

This applies to other pathogen groups as well. DNA viruses are also capable of host switching (M. E. Woolhouse et al., 2016). This includes members of the family *Poxviridae*, such as its smallpox-causing member, the Variola virus (M. Woolhouse & Gaunt, 2007). While the new squirrelpox virus recently isolated in Berlin did not cause any symptoms in exposed humans (Wibbelt et al., 2017), its discovery highlights the still unexplored diversity of this clinically important clade. Further, bacteria, rather than RNA viruses, have been reported as the underlying cause of most new EID 'events' (K. E. Jones et al., 2008). This apparent discrepancy is actually a consequence of the operating definitions used – the authors counted each novel drug-resistant strain as a separate 'event', highlighting the dangerous rise of antibiotic resistance. New bacterial strains may also become more virulent, as shown by the example of the 2011 enterohaemorrhagic *Escherichia coli* (EHEC) epidemic in Germany (Frank et al., 2011). Their emergence is often facilitated by increasing human exposure and population density, as well as by fast evolution of bacteria, especially due to horizontal gene transfer (K. E. Jones et al., 2008; Trappe et al., 2016; Vouga & Greub, 2016). Environmental changes, including climate change, could also play a role, as suggested for bacterial (Vouga & Greub, 2016), vector-borne (K. E. Jones et al., 2008; J. A. Patz et al., 2005) and fungal pathogens (Casadevall et al., 2019; Garcia-Solache & Casadevall, 2010). This included *Candida auris* – a fungus which, since its emergence in 2009, has been named one of the most urgent, drug-resistant threats (Casadevall et al., 2019; CDC, 2019).

Further, accidental or deliberate release of dangerous pathogens could lead not only to localized outbreaks but potentially also to a deadly pandemic (Nuzzo et al., 2019). Known, unmodified agents can be used as bioweapons (Riedel, 2004), and could possibly also emerge via either laboratory escape or even a vaccine challenge trial (Holmes et al., 2021; Rozo & Gronvall, 2015). Recent developments in the field of synthetic biology have raised concerns that even more virulent threats could be created and used for malicious purposes by either bioterrorists or state actors (Nuzzo et al., 2019). A report by National Academies of Sciences, Engineering, and Medicine (2018) classifies making existing viruses more dangerous as an issue of mid-to-high concern. Analogous modification of bacteria is identified as a matter of the highest concern, along with

re-creation of known viral pathogens, proven to be possible by the synthesis of a close relative of the Variola virus the same year (Noyce et al., 2018). In addition to that, strains modified through gain-of-function (GoF) research, intended to enhance preparedness for future emerging infectious diseases (EIDs), could cause epidemics (or pandemics) of their own if spread outside of a secure laboratory. Controversial GoF studies have been published for both influenza and coronaviruses, leading to calls for a moratorium on creating potential pandemic pathogens (Herfst et al., 2012; Imai et al., 2012; Lipsitch & Inglesby, 2014). Several years later, the emergence of SARS-CoV-2 added another dimension to those considerations. While the zoonotic origin of the virus has been assumed from the early days of the pandemic (P. Zhou et al., 2020), the 'lab leak' hypothesis attracted considerable media attention and has also been discussed in the scientific literature, although as of early 2022, it remains unproven and described as unlikely (Andersen et al., 2020; Holmes et al., 2021; Lytras et al., 2021; Relman, 2020).

Irrespective of whether novel pathogens arise with or without human intervention, detecting them faces similar challenges. Targeted diagnostic assays specialize in reporting markers of previously known agents. Therefore, only open-view methods, such as those based on DNA or RNA sequencing, can be expected to detect pathogens highly divergent from all known ones (Calistri & Palù, 2015; Lecuit & Eloit, 2014). Standard computational tools used to analyze the resulting data rely on matching the obtained sequences with genomes of known pathogens, usually based on some measure of sequence similarity. Similar approaches can also be used to assess risks associated with synthetic sequences, also as a form of computational screening performed *before* their synthesis (Carter & Friedman, 2015; Diggans & Leproust, 2019; National Academies of Sciences, Engineering, and Medicine, 2018; National Research Council, 2010). However, dependence on curated lists of potentially dangerous agents leads to a crucial vulnerability, as pathogens sufficiently different from the known ones may remain undetected. This highlights the importance of using a comprehensive and appropriate reference database, a problem also known in other related fields, such as taxonomic classification of metagenomic reads (Piro et al., 2020). On the other hand, generalization to novel species or strains could also be explicitly addressed by the development of improved, specialized computational approaches. In this thesis, I will mainly focus on the applications of deep learning as a promising alternative to methods based on sequence alignment or homology, especially in the context of classification of unassembled next-generation sequencing (NGS) reads and read fragments for fast detection of novel pathogens. I will also stress the importance of the appropriate data setup, including data curation and the underlying assumptions, which affect both the previously existing and newly-developed approaches.

1.2 Pathogen detection using high-throughput sequencing

1.2.1 DNA and RNA sequencing: a brief overview

The genetic information of all known cellular organisms and DNA viruses is encoded in their DNA, a polymer consisting of four nucleotide types, each containing one of the nucleobases: adenine, cytosine, guanine or thymine. Those are conventionally represented as one of four letters: A, C, G or T, with an 'unknown' nucleotide designated by an N. In RNA, used to store the genomic information of RNA viruses, thymine is generally replaced with uracil (U). Many complex processes, including nucleotide modification (in both DNA and RNA), post-transcriptional and post-translational regulation, as well as effects dependent on genome organization and chromatin accessibility (in eukaryotes), contribute to the final expression of the encoded genes, giving rise to the complex biological system of a cell or a virus. At the same time, an agent contained in a sample can be successfully identified based on its genome via next-generation sequencing (NGS) (Calistri & Palù, 2015; Lecuit & Eloit, 2014). Although multiple competing sequencing technologies exist, in this work I will focus on two platforms arguably most relevant for the pathogen detection tasks. Illumina sequencing is an established, highly accurate technique producing short reads of up to around 300 base pairs (bp), depending on the instrument (Goodwin et al., 2016). It is efficient and scalable, with low-error rates facilitating pathogen identification down to the strain level. Although it can only directly sequence DNA, reliable RNA sequencing protocols exist.

A more recent platform by Oxford Nanopore Technologies (ONT) is a type of long-read sequencing commonly generating reads up to two orders of magnitude longer than Illumina, with record lengths measured in millions of base pairs (Amarasinghe et al., 2020). This comes at the cost of lower accuracy, throughput and cost-efficiency, although rapid progress on all those fronts could change this in the future. It is also possible to sequence RNA directly. Another crucial advantage of Nanopore sequencing is that some of the instruments can be very small, allowing for easy setup of mobile labs and fast analyses in the field. This has been demonstrated for molecular surveillance during the Ebola outbreak of 2015 (Quick et al., 2016), *in situ* metagenomics analyses of both clinical and environmental samples (Latorre-Pérez et al., 2020), and even sequencing runs onboard the International Space Station (Castro-Wallace et al., 2017). Finally, ONT sequencing supports real-time analyses by design, which can speed up diagnostics in time-critical applications (Quick et al., 2016) or be used to selectively sequence the organisms of interest (Loose et al., 2016; Ulrich et al., 2022). On the other hand,

approaches for real-time analysis of Illumina runs have also been developed (Lindner et al., 2017; Loka et al., 2019; Tausch, Loka, et al., 2018).

1.2.2 Mapping and alignment-based pipelines

The generated reads correspond to fragments of the original genome with unknown order. They can be analyzed directly or first assembled into contigs – longer, continuous sequences, reconstructing the most probable order of available subsequences. However, assembly is a difficult problem, especially in the metagenomic context, when multiple different species are present in the sample (Breitwieser et al., 2017). In this thesis, I will focus on direct analysis of unassembled reads – this approach is especially suited for supporting real-time pathogen detection and risk assessment procedures for synthetic oligonucleotides. Nevertheless, I will also show that the developed methods can be successfully used on assembled contigs and full genomes as well.

One of the standard methods of analysis used for biological sequences of any length relies on aligning the query sequence to a database of known references, aiming to detect possible homology between them. BLAST (Altschul et al., 1990; Altschul et al., 1997; Camacho et al., 2009) remains one of the most popular sequence alignment tools over 30 years after the original publication (Alser et al., 2021). There are many flavours of BLAST, appropriate for alignments between nucleotide, protein, and translated nucleotide sequences, as well as for either relatively closely related or dissimilar sequences (Camacho et al., 2009). While variants such as blastn or discontiguous megablast are especially fit for comparisons of nucleotide sequences across (potentially novel) species, BLAST can be computationally inefficient on large query sets corresponding to samples of millions to billions NGS reads (Deneke et al., 2017). For this reason, specialized read mappers are usually used to align the reads to reference genomes. Bowtie2 (Langmead & Salzberg, 2012) and BWA (H. Li & Durbin, 2010), including the BWA-MEM algorithm (H. Li, 2013), are the most popular aligners for Illumina reads, even though multiple alternatives have been developed in the last decades (Alser et al., 2021). HiLive and HiLive2 (Lindner et al., 2017; Loka et al., 2019) are special-purpose Illumina mappers enabling real-time analysis during an Illumina sequencing run, which is crucial for some of the use-cases presented in this thesis. Aligners designed with long-read sequencing technologies in mind, such as minimap2, can process the corresponding reads at least an order of magnitude faster, although they can perform slightly worse than BWA-MEM on short reads (Alser et al., 2021; H. Li, 2018). A broad overview of over 100 different read alignment tools, including the algorithmic details, has been presented in a recent review by Alser et al. (2021).

All those methods can be used for pathogen detection and risk assessment, as a match to a reference genome of a known pathogen may suggest its presence in the analyzed sample. However, spurious assignments are also possible, so a single match should not be treated as definitive evidence. On the other hand, in clinical samples, the number of reads originating from the causative agent may be extremely low, so the presence of a pathogen may be reflected in only a handful of matches (Andrusch et al., 2018). Further, the host reads usually greatly outnumber the viral, bacterial or fungal reads, and additional contaminants may be present. Therefore, read mapping is often used as only one of multiple steps in larger, specialized pathogen identification pipelines, such as SURPI (Naccache et al., 2014), Clinical PathoScope (Byrd et al., 2014), PathoScope 2.0 (Hong et al., 2014), Sigma (Ahn et al., 2015) or PAIPline (Andrusch et al., 2018). PathoLive (Tausch, Loka, et al., 2018) extends this to real-time pathogen detection during Illumina sequencing runs. As all these tools rely on read alignment results, they are indirectly affected by all the strengths and weaknesses of the underlying algorithms.

1.2.3 Taxonomic classification and profiling in metagenomics

Clinical samples containing mixtures of host, microbial and viral reads are a type of metagenomic samples. Hence, many of the problems related to pathogen detection and identification are special cases of more general challenges in computational metagenomics. For example, Sigma (Ahn et al., 2015) uses a probabilistic model to estimate relative abundances of strains present in the sample. This task is of interest not only in the context of pathogen detection but also in taxonomic profiling of metagenomic samples in general. Related approaches, relying on statistical analysis of read alignment results, include GAAS (Angly et al., 2009), GRAMMy (Xia et al., 2011), TAMER (Jiang et al., 2012), GASiC (Lindner & Renard, 2013), TAEC (Sohn et al., 2014), and Centrifuge (D. Kim et al., 2016). Other tools specialize in differential abundance analyzes (Fischer et al., 2017), filtering-out unlikely references based on their coverage profiles (Dadi et al., 2017), finding protein matches (Menzel et al., 2016; Mirdita et al., 2021; Steinegger & Söding, 2017; von Meijenfeldt et al., 2019), or enable estimating a measure of genomic distance between previously unknown candidate organisms and the known references (Lindner & Renard, 2015). MEGAN (Huson et al., 2007) can be viewed as a precursor of all those methods, as it also post-processes sequence alignment output. If a read can be matched to multiple leaves in a taxonomic tree, resulting ambiguities are resolved by assigning it to the lowest common ancestor (LCA) of those leaves. Alternatively, a top-down approach based on the concept of the deepest uncommon descendant can also be used (Piro et al., 2016).

Another category of taxonomic profilers, such as GOTTCHA (Freitas et al., 2015), mOTUs (Milanese et al., 2019; Sunagawa et al., 2013) and MetaPhlAn (Beghini et al., 2021; Segata et al., 2012; Truong et al., 2015), uses databases of unique markers to detect signatures of the taxa present in the sample. This improves the processing speed, which is often important for large samples, but also leads to an increased risk of missing the organisms sequenced with low coverage (presumably due to their low abundance), or unrepresented in the marker database. Those problems can be alleviated using taxonomic classifiers assigning reads to taxa based on full reference genomes, but not necessarily relying on read mappers or BLAST. An efficient and popular group of tools, including Kraken (Wood et al., 2019; Wood & Salzberg, 2014), CLARK (Ounit et al., 2015), and ganon (Piro et al., 2020), counts occurrences of taxa-specific k-mers within each read to assign it to the best-matching node in the taxonomy tree. More specialized tools enable correcting for low coverage of less abundant pathogens (Breitwieser et al., 2018), abundance estimation (J. Lu et al., 2017) and even real-time taxonomic classification of reads produced by the sequencer (Tausch, Strauch, et al., 2018). Finally, k-mer frequencies can be used as input for machine learning approaches such as NBC (G. Rosen et al., 2008; G. L. Rosen & Lim, 2012; G. L. Rosen et al., 2011) or PhymmBL (Brady & Salzberg, 2009), and k-mer embeddings have been used in the long short-term memory network (LSTM) architecture of DeepMicrobes (Liang et al., 2020) and the convolutional neural network (CNN) of CNN-RAI (Karagöz & Nalbantoglu, 2021). On the other hand, GeNet (Rojas-Carulla et al., 2019) encoded DNA inputs as sequences of integers.

The sheer number of available tools makes comparing them a difficult challenge. The Critical Assessment of Metagenome Interpretation (CAMI) initiative is intended to alleviate this problem by offering a unified benchmarking platform, including a separate clinical pathogen detection challenge in its second round (Meyer et al., 2021; Sczyrba et al., 2017). A broader overview of the field has been also presented in recent reviews by Breitwieser et al. (2017) and Sun et al. (2021). In this thesis, I will present deep learning approaches for detection of novel pathogens. In this context, machine learning and alignment-based algorithms have been shown to outperform k-mer based taxonomic classifiers (Deneke et al., 2017). Therefore, the main benchmarking focus will be placed on comparing the newly developed methods with state-of-the-art sequence aligners and machine learning models, even though computationally efficient k-mer based methods are also widely used in the field, at least for detecting *previously known* organisms.

1.3 Deep learning in genomics

1.3.1 Predictive models in regulatory genomics

Before finding applications in a wide range of computational biology problems, including metagenomics and pathogen identification, deep learning (LeCun et al., 2015) has been successfully used in eukaryotic regulatory genomics. DeepBind (Alipanahi et al., 2015), DeepSEA (J. Zhou & Troyanskaya, 2015) and Basset (Kelley et al., 2016) have shown the capabilities of CNNs using DNA or RNA sequences as inputs, predicting protein binding sites and regulatory effects of non-coding variants. This has sparked intense research interest, with a focus on human genomics (e.g. Greenside et al. (2018), Kelley (2020), Kelley et al. (2018), Nair et al. (2019), Quang and Xie (2016, 2019), and Zeng et al. (2016)), summarized in a comprehensive review by Eraslan et al. (2019). Recent developments include predictions at base pair resolution (Avsec, Weilert, et al., 2021) and exploring the potential of Transformer architectures (Avsec, Agarwal, et al., 2021). Although those approaches are tailored for problems in eukaryotic regulatory genomics (and as such, the challenges they deal with differ from those encountered in metagenomics or pathogen detection), multiple design choices are compatible with DNA or RNA-based predictive tasks in general.

Many applications rely on supervised models (often CNNs) predicting molecular phenotypes from DNA sequence inputs. The sequences are usually supplied using a distributed orthographic representation, where each nucleotide is one-hot encoded, i.e., represented as a numerical vector of length 4, where a single element (at an index dependent on the particular nucleotide type) is equal to 1, and the other elements are 0. This representation is often simply called 'one-hot encoding' for short, even though the one-hot vectors correspond to sequence elements, not the sequences themselves. This can be easily extended to the RNA alphabet (Budach & Marsico, 2018), and unknown nucleotides (Ns) can be represented as all-zero vectors. As a given DNA sequence and its reverse-complement are often expected to result in identical outputs, the final outputs can be unified by designing a DNA-specific architecture accounting for this fact (Alipanahi et al., 2015; R. C. Brown et al., 2018; Onimaru et al., 2020; Quang & Xie, 2019; Shrikumar et al., 2017b). A comparison of existing approaches to enforcing reverse-complement equivariance in CNNs has been recently presented by H. Zhou et al. (2022). Other possible representations of DNA inputs include encoding read coverage in synthetic images (Poplin et al., 2018), transforming the sequences into their chaos game representation (CGR) (Löchel et al., 2020; Löchel & Heider, 2021; Rizzo et al., 2016) or using k-mer embeddings (Liang et al., 2020), although they are less frequently used

in the context of regulatory genomics. DNABERT, a *k*-mer based pre-trained language model, is a notable exception (Ji et al., 2021).

1.3.2 Interpretability and feature attributions

Interpretability of machine learning models is an important factor in their evaluation, especially if the model's predictions may have a large, direct impact on human lives. This is especially true for applications in biomedical research. Deep learning is often described as a 'black box', and the same criticism can be applied to other popular machine learning algorithms (Rudin, 2019). Rudin (2019) proposes a strict distinction between explainable and interpretable models, with the former defined as standard 'black box' models, explained post-hoc by another model, while the latter are designed to be inherently interpretable. However, this classification has not been universally accepted, and many papers use those two terms interchangeably. Here, I will use the term 'interpretability' also in the meaning of explainability of 'black box' networks, following Lundberg and Lee (2017) and Shrikumar et al. (2017a).

Notably, whether a trained model is interpretable depends not only on the algorithm used but also on the particular domain and task. Rudin (2019) defines a 'black box' model as 'a function that is too complicated for any human to comprehend'. This is a reasonable and intuitive definition, but it follows that interpretability must be, to a large extent, subjective. Even the very concept of 'understanding' has been an object of intense philosophical and psychological studies (Grimm, 2021). It could be argued that to 'understand' a function, one should understand not only 'how' or 'why' the outputs are assigned to its inputs, but also the meaning of inputs and outputs themselves. This consideration has some far-reaching implications for explainability of neural networks in genomics.

More specifically, in contrast to structured tabular data, images or text samples, relevant patterns in raw biological sequences may be difficult to notice even for experts. Depending on the particular task and sequence representation, models or explanations that would be understandable in other contexts often remain opaque. For example, random forests are sometimes considered to be more interpretable than neural networks due to their reliance on relatively simple decision trees, and the importance of individual input features can be evaluated using Gini importance Deneke et al. (2017). However, if the input features are *k*-mer frequencies and the predicted labels correspond to a bacterial phenotype, several open questions remain. *Why* is a particular *k*-mer important for the final decision of the model? How much 'understanding' do we gain by knowing that the

'GGA' trimer is the most important 'marker' of bacterial pathogenicity according to the trained model?

This example is not intended to criticize the efforts of making machine learning models for genomics more interpretable. On the contrary – it highlights an important problem, which makes those efforts especially valuable. Given the wealth and complexity of genomic data, we often do not know what patterns to look for *a priori*. Even in the context of regulatory genomics, where many important sequence motifs have been identified, a lot of research effort is invested into discovering *new* biological rules: new patterns or new regulatory 'grammars', governing the interactions between individual transcription factor binding sites. For this reason, some 'black box' models are trained not primarily for their predictive performance, but rather to be a source of useful explanations (Almeida et al., 2021; Avsec, Weilert, et al., 2021; L. Chen & Capra, 2020). This may be even more important in the context of predicting an abstract phenotype such as pathogenicity or host range from short reads or read fragments originating from taxonomically distant genomes. In this particular case, it is not immediately clear what would even constitute a reasonable pattern to look for (except for direct sequence similarity to parts of reference genomes), so building an inherently interpretable model could prove challenging. Training a 'black box' model and providing some explanations for its behaviour is more feasible. However, even in this case, evaluating the obtained explanations remains difficult.

The need for visualization of discovered patters has been stressed already in the early days of deep learning for genomics. DeepBind's convolutional filters were represented as sequence logos based on most-activating subsequences (Alipanahi et al., 2015), and the most-relevant subsequences influencing DeepSEA outputs were identified via *in silico* saturated mutagenesis (J. Zhou & Troyanskaya, 2015). This was followed by studies visualising important regions of the input with saliency maps (Simonyan et al., 2014), as direct optimization is usually challenging due to the discrete nature of the DNA and RNA sequences (Lanchantin et al., 2016; Lanchantin et al., 2017). Further improvements were brought by a family of general-purpose, backpropagation-based approaches such as layer-wise relevance propagation (LRP) and Integrated Gradients (Bach et al., 2015; Sundararajan et al., 2016), as well as their multiple enhancements (e.g. Jha et al. (2020) and Montavon et al. (2019)). DeepLIFT (Shrikumar et al., 2017a) is conceptually similar to LRP, as it propagates activation differences between a given input and a 'reference input'. It is also one of the best examples of methods developed for applications in computational biology, which is now widely used across the broader deep learning field. Another approach, LIME, relies on learning an interpretable model that will approximate the behaviour of the original model in the proximity of a given input (Ribeiro et al., 2016). LRP, DeepLIFT and LIME all assign a score of importance

to each of the input features, and they can be expressed in a more unified framework as examples of additive attribution methods (Lundberg & Lee, 2017).

1.4 Machine learning for viral and microbial bioinformatics

1.4.1 Recent developments

Applications of machine learning in genomics are not restricted to human (or eukaryotic) genomics, although some authors do treat those terms interchangeably, without mentioning applications to viral and microbial data (Eraslan et al., 2019; R. Li et al., 2021). Rule-based models (Drouin et al., 2016; Drouin et al., 2019; MacDonald & Beiko, 2010; Tamura & D'haeseleer, 2008), regression (Aun et al., 2018; Lees et al., 2018; Lees et al., 2020; Lees et al., 2016) and support-vector machines (SVMs) (Feldbauer et al., 2015; Weimann et al., 2016) have been used to predict multiple phenotypes from bacterial genomes. Antimicrobial resistance is a group of phenotypes that attracted a lot of attention, as the emergence of resistant strains has led to a long-term public health crisis (Levy & Marshall, 2004; Neu, 1992; Piddock, 2012; Ventola, 2015). A broad overview of currently available prediction methods has been recently summarized by X. Li et al. (2021). For some pathogens, such as *Mycobacterium tuberculosis* (the causative agent of tuberculosis), the resistance is usually conferred by single-nucleotide polymorphisms (SNPs), and can be predicted from whole-genome sequencing (WGS) data by random forests, linear models and neural networks with seemingly comparable performance (M. L. Chen et al., 2019; Gröschel et al., 2021; Lees et al., 2020). For many other species, presence of certain genes is the most important factor. Therefore, detecting novel antibiotic resistance genes is needed as well. Neural networks, k-nearest neighbours (kNN) classifiers, logistic regression, and ensemble models can be useful in this case (Arango-Argoty et al., 2018; Z. Wang et al., 2021). This problem is connected to the broader field of protein function prediction, where deep learning has also proven useful (Gligorijević et al., 2018; Gligorijević et al., 2021; Kulmanov & Hoehndorf, 2020; Villegas-Morcillo et al., 2021; You et al., 2021). SVMs, random forests and neural networks have also been shown to predict if bacterial proteins are virulence factors, even though they may perform a wide range of different functions (Gupta et al., 2014; Rentzsch et al., 2020; R. Xie et al., 2021). Finally and most importantly for the work presented here, this line of research can also be scaled to predict whether novel bacteria could be human pathogens (Barash et al., 2018; Deneke et al., 2017).

Viral host range prediction is a conceptually similar problem. Bacteriophage hosts can be inferred using alignment or *k*-mer based measures of distance to both other viruses and prospective hosts, as phage sequences tend to be similar to the sequences of their hosts. An overview of different methods has been presented by R. A. Edwards et al. (2016), and similar approaches have also been developed since (Ahlgren et al., 2017; Galiez et al., 2017; Villarroel et al., 2016; W. Wang et al., 2020; Zielezinski, Barylski, et al., 2021; Zielezinski, Deorowicz, et al., 2021). Random forests have been used by M. Zhang et al. (2017) and Boeckaerts et al. (2021) to predict the hosts based on nucleotide composition and phage receptor-binding protein sequences, respectively. Leite, Brochet, et al. (2018) presented a small, shallow neural network relying on protein features and later explored better-performing ensemble models and one-class learning approaches (Leite, Lopez, et al., 2018). Recently presented PredPHI (M. Li et al., 2020) and HoPhage (Tan et al., 2021) used deep learning, with the latter explicitly evaluating the feasibility of inferring hosts directly from metagenomic reads. Phage lifestyle (virulent of temperate) can also be predicted with both random forests and neural networks (Hockenberry & Wilke, 2021; McNair et al., 2012; S. Wu, Fang, et al., 2021).

Simple nucleotide composition features have been successfully used to predict the hosts of eukaryotic viruses as well (Kapoor et al., 2010). Classical machine learning models have been presented for influenza A (Eng et al., 2014; Xu et al., 2017) and coronaviruses (Tang et al., 2015). H. Li and Sun (2018) used a similar approach for both those virus groups, as well as rabies lyssaviruses. Deep neural networks of VIDHOP (Mock et al., 2020) can predict the host directly from genome sequences and were evaluated on influenza A, rabies lyssaviruses and rotavirus A datasets, outperforming previous approaches where direct comparisons were possible. Furthermore, they supported a wider range of possible hosts and yielded accurate predictions also for inputs as short as Illumina sequencing reads. Gałan et al. (2019) presented SVMs and regression models focusing on novel viruses without taxonomic assignment, but they require long input sequences and only fairly broad host categories (bacteria, plants, vertebrates, or arthropods) are supported. Z. Zhang et al. (2019) designed kNN classifiers detecting novel human viruses and tested them also on inputs of just several hundred bp. In contrast, Babayan et al. (2018) analyzed full genomes of human viruses and predicted their unknown reservoirs or vectors with gradient boosting machines. Recently, Mollentze et al. (2021) presented host prediction models trained on a dataset of 36 viral families and evaluated on both a wider selection of held-out viruses and SARS-CoV-2 data. The COVID-19 pandemic has sparked an intensified interest in studies assessing zoonotic potential of coronaviruses specifically (Brierley & Fowler, 2021; Q. Guo et al., 2021; Q. Guo et al., 2020; Qiang et al., 2020; Wardeh et al., 2021). A related line of research aiming to

predict interactions between viral and human proteins (Eid et al., 2016; X. Yang et al., 2019; X. Zhou et al., 2018) has also been affected by both the turn towards deep learning and the current importance of SARS-CoV-2 (Lanchantin et al., 2021; Liu-Wei et al., 2021).

In the metagenomic context, detecting bacteriophage sequences is often needed, especially given the abundance, diversity and ecological significance of bacterial viruses (Roux, Enault, et al., 2015; Roux, Hallam, et al., 2015). Modern approaches have used random forests (Amgarten et al., 2018; J. Guo et al., 2021; Zheng et al., 2019), simple neural networks (Kieft et al., 2020) and deep learning (Auslander et al., 2020; Fang et al., 2019; Miao et al., 2021; Ren et al., 2020; Tampuu et al., 2019). In the phage taxonomic classification task, graph convolutional networks (J. Shang et al., 2021) have been shown to outperform clustering-based (Bin Jang et al., 2019) and alignment-based methods. A CNN-based, hierarchical model has also been proposed for simultaneous detection and taxonomic classification of RNA virus reads (J. Shang & Sun, 2020), while classical machine learning algorithms have been used for subtyping of HIV, dengue, influenza A, hepatitis B, and hepatitis C (Remita et al., 2017; Solis-Reyes et al., 2018).

Those problems are related to taxonomic classification of bacterial reads as well. Promising results have been reported for neural networks of DeepMicrobes (Liang et al., 2020) and GeNet (Rojas-Carulla et al., 2019). A new generation of methods follows the recent turn towards self-supervised, pre-trained models that have originated in the natural language processing field (T. Brown et al., 2020; Devlin et al., 2019; Vaswani et al., 2017) and has also influenced computer vision research (Dosovitskiy et al., 2021). BERTax (Mock et al., 2021) supports eukaryotic, bacterial, archaeal and viral sequences, relying on a Transformer model based on the BERT architecture (Devlin et al., 2019). A classifier using BERTax embeddings outperformed all other benchmarked tools on the set of most divergent sequences and combined with a database-based approach, MMSeqs2 (Mirdita et al., 2021; Steinegger & Söding, 2017), it was the best method overall. In turn, LookingGlass (Hoarfrost et al., 2020) uses a self-supervised LSTM model and a window of only 100bp to train embedding for sequences as short as an Illumina read. It has been shown to facilitate a range of downstream tasks such as predicting the correct translation frame, oxidoreductase activity and optimal enzyme temperature directly from unassembled reads. A similar approach was used to predict mutational escape for influenza haemagglutinin, HIV-1 envelope glycoprotein, and the Spike protein of SARS-CoV-2 (Hie et al., 2021). Another Transformer based on protein clusters has been recently reported to improve detection of bacteriophage contigs (J. Shang et al., 2022). Pre-trained protein embeddings of ProtBert (Elnaggar et al., 2021)

have also been used in a taxonomic classification pipeline for NGS reads (Voigt et al., 2021).

1.4.2 Opportunities and challenges

Machine learning approaches have been successfully deployed in many subfields of microbial and viral bioinformatics, with many applications specifically in metagenomics and genomics. In recent years, many new deep learning approaches have been developed; the work presented in this thesis was a part of this wider research effort. Deep neural networks can often be more expressive than other models, allowing them to learn complex relationships between the genotype and a prediction target, such as a phenotype of interest or a taxonomic unit. This, in turn, facilitates generalization to sequences highly divergent from those seen during training. In contrast to classical machine learning algorithms, CNNs, LSTMs and Transformers do not necessarily need manual feature engineering, which enables discovering new patterns that a human expert would not explicitly consider. An increased research effort on interpretability of the learned models could therefore lead to new, biologically or even clinically relevant discoveries, as in the case of deep learning in human regulatory genomics.

On the other hand, developing and evaluating methods improving explainability is challenging, especially if the ground truth is not available. For microbial and viral data, this is often the case to an even greater extent than in human genomics, as multiple different, diverse taxa have to be taken into account, and many remain understudied or undiscovered. Further, despite the best efforts from the community, the datasets may still be sparsely annotated, incomplete, and difficult to curate and integrate. In general, the proper handling of data is arguably more important than the algorithm used; a selection of potential pitfalls has been recently summarized by Whalen et al. (2021). Importantly, the crucial assumption of machine learning, that the data is independent and identically distributed (i.i.d.), is usually broken. The training and test sets may be characterized by differing distributions of input features x (covariate shift) or target variables y (prior probability shift); the conditional probabilities $P(x|y)$ and $P(y|x)$ may differ between the sets as well (Storkey, 2009; Whalen et al., 2021; K. Zhang et al., 2013). This can make generalization difficult and has been extensively studied in the field of domain adaptation. What is more, data imbalance is often an additional problem, and due to the inherent sampling bias present in the available databases, it is not always clear what the 'representative' ratios of class members should be. The datasets must be constructed carefully to avoid data leakage between training and test sets; this may take the form of dividing the sets based on the sequence similarity between the data points or testing

on representatives of taxonomic units unseen in training. In any case, the data setup is crucial for modelling the target biological task. On the other hand, searching for proper neural architectures and representations has also been an area of intensive research, as the design choices known from other fields of application are not always optimal for metagenomic, viral and microbial data. This applies even to the closely related field of deep learning for regulatory genomics, where, for example, large receptive fields are needed. Hence, dilated convolutions have been adopted in CNN models (Avsec, Weilert, et al., 2021; Kelley, 2020), and recently presented Enformer architectures were reported to detect sequence elements 100kb apart (Avsec, Agarwal, et al., 2021). Receptive field size is not as crucial for classifying short reads or even metagenomic contigs, which are often 1–2 orders of magnitude shorter.

Finally, machine learning models can also be used to generate and optimize DNA, RNA and protein sequences, maximizing their desired activity (Alley et al., 2019; Angermueller et al., 2019; Anishchenko et al., 2021; Biswas et al., 2021; Brookes et al., 2019; Gupta & Kundaje, 2019; Gupta & Zou, 2019; Linder et al., 2019; Ovchinnikov & Huang, 2021; Schreiber et al., 2020; Sinai et al., 2020; J. Wang et al., 2021). This has profound consequences for the field of synthetic biology, enabling computational design of novel proteins with target functions, but also opening new avenues for dual-use research. The risks associated with novel sequences can be difficult to evaluate (Diggans & Leproust, 2019; National Academies of Sciences, Engineering, and Medicine, 2018), but risk prediction models generalizing to previously unseen regions of the sequence space could help alleviate this problem. Such approaches could be related to previously published approaches for predicting bacterial pathogenic potentials (Deneke et al., 2017) and viral host range (Mock et al., 2020; Z. Zhang et al., 2019), or rely on protein function prediction, like the recently proposed SeqScreen framework (Balaji et al., 2021).

1.5 Thesis outline

The overarching theme of this work is the development of new, deep learning approaches for predicting pathogenic or infectious potentials directly from nucleic acid sequences. The final goal is to enable detection of novel pathogens, for which standard tools – relying on homology detection or taxonomic classification – are not enough (Figure 1.1). Although I place the main focus on use-cases based on next-generation sequencing, the methods presented here could in principle be used to screen synthetic sequences as well. In the following chapters, I describe four parts of the project, each corresponding to a more specific prediction task and introducing crucial methodological developments. Each chapter is thought to be relatively self-contained and understandable without

reading the others. At the same time, they are arranged in the intended order of reading, with each chapter building upon the previous ones, extending the methodology and presenting new results.

Chapter 2 introduces DeePaC, a deep learning approach for pathogenic potential prediction. It demonstrates the superior performance of neural networks in the task of predicting whether DNA sequences (next-generation sequencing reads or genomes) originate from novel bacterial pathogens. Further, it presents the `deepac` python package, supporting easy and flexible design, training, and evaluation of multiple neural architectures for DNA sequence inputs, with an additional emphasis on guaranteeing invariance to DNA reverse-complementarity. This is accompanied by a command-line tool facilitating inference with the trained models and integration with bioinformatic pipelines without requiring deep learning expertise from the end user. I conceptualized the project with Bernhard Renard and support from Robert Rentzsch. I also implemented `deepac` and the supplementary R scripts, some of which were based on R scripts by Carlus Deneke originally used for the development and evaluation of PaPrBaG (Deneke et al., 2017). Finally, I collected the data, performed all experiments presented in the paper and wrote the manuscript with feedback from all authors. Anja Seidel performed preliminary, proof-of-concept evaluation of the reverse-complement parameter sharing convolutional neural networks (RC-CNNs) using another implementation focusing exclusively on CNNs (Shrikumar et al., 2017b). This included showing the first evidence of the benefits of input dropout. Her work was a part of her Master's thesis (Seidel, 2018). Robert Rentzsch's insights helped design the data mining strategy employed in the scripts. Bernhard Renard offered advice, supervision and support at all stages of the project. The chapter is based on the following article:

> Bartoszewicz, J. M., Seidel, A., Rentzsch, R., & Renard, B. Y. (2020). DeePaC: predicting pathogenic potential of novel DNA with reverse-complement neural networks. *Bioinformatics*, *36*(1), 81–89. https://doi.org/10.1093/bioinformatics/btz541

I also presented the results in a poster at the Intelligent Systems for Molecular Biology 2019 conference.

Chapter 3 presents an application of DeePaC to another task – predicting if the input sequences originate from human or non-human viruses. We also introduce a new method of convolutional filter visualization, presenting the information content of each input nucleotide along its contributions to the final prediction of the network. We show how mapping predictions and feature attributions back to the individual residues of input sequences allows exploring the network's behaviour at the genome, gene and

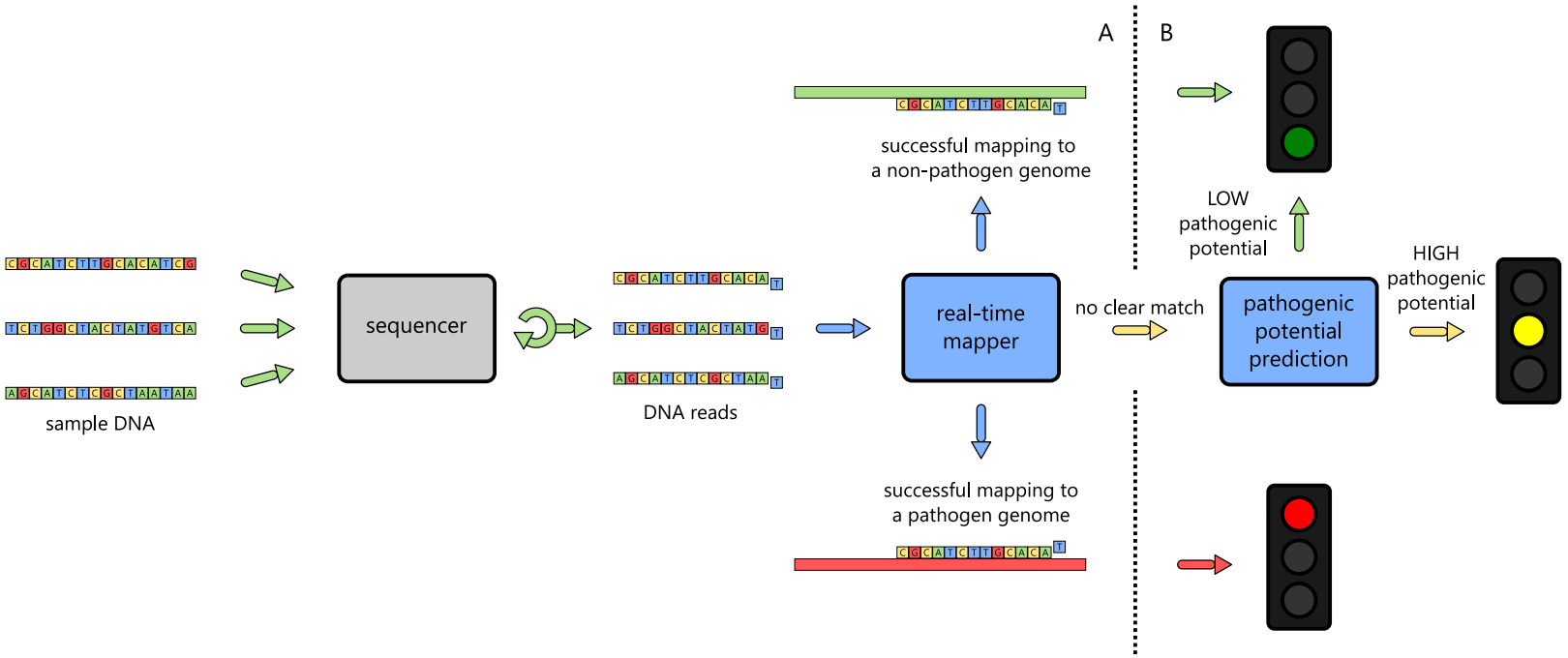


Figure 1.1: Overview of the proposed family of approaches, implemented in the DeePaC package and its plugins. 1.1A: Standard approach. Sample bacterial, viral, or fungal reads produced by the sequencer are mapped against known pathogen and non-pathogen genomes. This can also be done in real time, as the sequencer is running. Alternative methods (e.g. *k*-mer based taxonomic classifiers) can be used instead of sequence alignment. 1.1B: Extended approach. Reads without clear matches (unmapped or low-quality alignments) are passed to a previously trained model, which predicts whether they originate from an agent with a potentially pathogenic phenotype, possibly in real time as well. Putative pathogen reads can then be used for downstream analysis.

protein structure level (including SARS-CoV-2 and its Spike protein). Finally, we integrate those interpretability tools with the main `deepac` codebase, supporting both programmatic and command-line usage. The interpretability part of the project was jointly conceptualized by all authors. Anja Seidel implemented and evaluated prototype interpretability scripts for the RC-CNN implementation of Shrikumar et al. (2017b). This was described in her Master's thesis (Seidel, 2018). I reimplemented crucial parts for compatibility with the `deepac` implementation of RC-CNNs and RC-LSTMs, developed them further, integrated them with the rest of the package and extended the approach to the protein structure level. I also introduced the partial Shapley values to formalize our method of filter visualization and performed all analyses. The viral infectious potential prediction task was jointly conceived by Bernhard Renard and me. I compiled the viral host-range dataset with its subsets, performed all experiments, and wrote the manuscript with feedback from all authors. Bernhard Renard supervised and supported the project with helpful comments and advice at all times. The chapter is based on the following article:

> Bartoszewicz, J. M., Seidel, A., & Renard, B. Y. (2021a). Interpretable detection of novel human viruses from genome sequencing data. *NAR Genomics and Bioinformatics*, *3*(lqab004). https://doi.org/10.1093/nargab/lqab004

I also presented the results in a workshop paper with an accompanying poster:

> Bartoszewicz, J. M., Seidel, A., & Renard, B. Y. (2021b). Interpretable prediction of the infectious potential of novel viruses. *ICLR 2021 AI for Public Health Workshop*

Chapter 4 focuses on a more difficult version of the tasks presented in chapters 2 and 3 – detecting novel pathogens in real-time, as the sequencer is still running, and combining deep learning predictions with HiLive2, a real-time Illumina mapper (Loka et al., 2019). We use deeper architectures – residual networks (ResNets) invariant to reverse-complementarity – for high accuracy on very short Illumina read fragments without sacrificing the performance on full-length reads. We also show that analogous models work well for Nanopore data, outperforming the baseline approach (mapping the full, finished reads) by a significant margin on fragments corresponding to just 0.5s of sequencing time. In experiments modelling the pre-pandemic state of knowledge, we show drastically increased SARS-CoV-2 detection rates from incomplete reads, regardless of the sequencing technology used. I conceived the project with Bernhard Renard. Ulrich Genske evaluated the original DeePaC neural networks, trained some of the models

used in hyperparameter tuning, and implemented the scripts aggregating, processing and plotting the results. This was a part of his Bachelor's thesis (Genske, 2019). He also set up the Nanopore read simulation workflow and simulated the training and validation sets of bacterial Nanopore reads. I prepared all other presented datasets, designed, implemented and trained the reverse-complement ResNets, created the DeePaC-Live plugin linking DeePaC with HiLive2, benchmarked our models against alternatives and wrote the manuscript with feedback from all authors. Bernhard Renard supported and supervised the work throughout the project. The chapter is based on the following article:

> Bartoszewicz, J. M., Genske, U., & Renard, B. Y. (2021a). Deep learning-based real-time detection of novel pathogens during sequencing. *Briefings in Bioinformatics*, (bbab269). https://doi.org/10.1093/bib/bbab269

I also presented the results in a poster at the Intelligent Systems for Molecular Biology 2021 conference, a spotlight talk at Machine Learning in Computational Biology 2021, and two workshop papers with accompanying posters:

> Bartoszewicz, J. M., Genske, U., & Renard, B. Y. (2021b). Real-time pathogenicity prediction during genome sequencing of novel viruses and bacteria. *ICLR 2021 Machine Learning for Preventing and Combating Pandemics Workshop*
>
> Bartoszewicz, J. M., Genske, U., & Renard, B. Y. (2021c). Real-time pathogenicity prediction during genome sequencing of novel viruses and bacteria. *Machine Learning in Computational Biology 2021*

Finally, chapter 5 focuses on the third of the important pathogen groups considered in this thesis – fungi. We compile a manually curated database of pathogenic fungal species and their genomes, labelled with their corresponding host groups. We then apply the modified ResNet architecture introduced in chapter 4 to predict if fungal DNA sequences originate from species with a human host or from non-human pathogens, causing disease exclusively in plants or non-human animals. We also develop and visualize full genome representations using models trained with isolated reads only. Finally, we extend our approach to a multi-class scenario, where – under some simplifying assumptions – we can predict if an NGS read originates from a novel bacterial, viral or fungal pathogen, or from a harmless negative class of bacterial commensals and non-human viruses. Ferdous Nasri compiled a preliminary version of the database and ran pilot analyses with read-wise BLAST, as well as some simpler CNN architectures on the fungal dataset. Her work was the basis of her Master's thesis (Nasri, 2020). I conceived the project with Bernhard Renard, constructed and curated the final database as described here,

implemented the multi-class version of ResNets and performed all presented experiments. Melania Nowicka and I worked together on the visualization of the dataset structure. She designed, implemented and evaluated the scripts for training and plotting the UMAP embeddings; I had the idea to use the averaged read representations, added cosmetic changes to the scripts and performed the analyses presented in the paper. I wrote the manuscript with conceptual contributions by Ferdous Nasri and feedback from all authors. Bernhard Renard supervised and supported all parts of the project. The chapter is based on the following article:

> Bartoszewicz, J. M., Nasri, F., Nowicka, M., & Renard, B. Y. (2022). Detecting DNA of novel fungal pathogens using ResNets and a curated fungi-hosts data collection. *bioRxiv*. https://doi.org/10.1101/2021.11.30.470625
>
> Jakub Bartoszewicz and Ferdous Nasri contributed equally.

Our preliminary results have been presented by Ferdous Nasri in a poster at the Intelligent Systems for Molecular Biology 2021 conference.

2 Predicting bacterial pathogenic potential with DeePaC

Summary

We expect novel pathogens to arise due to their fast-paced evolution, and new species to be discovered thanks to advances in DNA sequencing and metagenomics. Moreover, recent developments in synthetic biology raise concerns that some strains of bacteria could be modified for malicious purposes. Traditional approaches to open-view pathogen detection depend on databases of known organisms, which limits their performance on unknown, unrecognized, and unmapped sequences. In contrast, machine learning methods can infer pathogenic phenotypes from single NGS reads, even though the biological context is unavailable. We present DeePaC, a Deep Learning Approach to Pathogenicity Classification. It includes a flexible framework allowing easy evaluation of neural architectures with reverse-complement parameter sharing. We show that convolutional neural networks and LSTMs outperform the state-of-the-art based on both sequence homology and machine learning. Combining a deep learning approach with integrating the predictions for both mates in a read pair results in cutting the error rate almost in half in comparison to the previous state-of-the-art.

This chapter is based on Bartoszewicz et al. (2020), which is a joint work with Anja Seidel, Robert Rentzsch and Bernhard Y. Renard. A detailed description of the authors' contributions can be found in section Thesis outline.

2.1 Background

2.1.1 Motivation

Our globalized world enables fast transmission of causative agents over great distances, and bacterial pathogens evolve quickly. Virulence genes may be easily exchanged between many bacterial species, leading to the emergence of new biological threats. In 2011, for example, a strain of *Escherichia coli* that acquired Shiga-toxin producing genes caused a major outbreak killing 53 people (Frank et al., 2011).

Since DNA sequencing has become the state-of-the-art in open-view pathogen detection (Calistri & Palù, 2015; Lecuit & Eloit, 2014), new pipelines and algorithms are needed to efficiently and accurately process the wealth of data resulting from every run. The number of sequences deposited in public databases grows exponentially, and a new challenge is to design computational tools for dealing with big sets of very divergent sequences (Piro et al., 2020). Furthermore, up to a trillion microbial species may be inhabiting the planet (Locey & Lennon, 2016). If this debated estimate (Willis, 2016) is correct, about 99.999% of the microbial biodiversity remains to be discovered. Both yet unknown and newly-emerging organisms may pose a public health threat. The risks are difficult to anticipate, but quick assessment is mandatory.

Recent developments in the field of synthetic biology have raised concerns about the possibility of creating new biological threats in the lab, either by accident or for malicious purposes. The National Academies of Sciences, Engineering, and Medicine (2018) identify genetic modification of existing bacteria to make them more dangerous as an issue of the highest concern. New methods for sequence-based classification of potential pathogens must be developed to safeguard future biosecurity and biosafety alike (National Research Council, 2010). Approaches based on comparing sequences to a reference database to detect previously known organisms are insufficient. This is especially important for shorter sequences, like NGS reads or synthetic oligonucleotides. The latter are often not screened before synthesis due to high computational cost and low accuracy of the predictions (Carter & Friedman, 2015; National Academies of Sciences, Engineering, and Medicine, 2018).

Assessing and mitigating risks based on the DNA sequence alone should involve computational methods able to recognize relevant patterns and generate predictions for novel sequences. Therefore, machine learning based approaches are a promising alternative to the traditional sequence analysis tools. For example, deep convolutional networks can be used to predict the lab of origin of the plasmids available in the Addgene repository (Nielsen & Voigt, 2018). This could help track a biological threat back to its source in case of a malicious attack or accidental release. Deneke et al. (2017) presented PaPrBaG, a random forest approach for predicting whether an Illumina read originates from a pathogenic or a non-pathogenic bacterium and showed that it generalizes to novel, previously unseen species. They introduce the concept of a *pathogenic potential* to differentiate between predicted probabilities of a given phenotype and true pathogenicity, which can only be realized in the biological context of a full genome and a specific host.

In this work, we present DeePaC, a Deep Learning Approach to Pathogenicity Classification. We focus on the scenario of pathogen detection from next-generation

sequencing data. However, the method presented here can in principle be used also for other sequences of similar length, both natural and synthetic.

2.1.2 Computational tools for pathogen detection

Taxonomy-dependent

Read-based pathogen detection methods may be roughly divided in two categories. Taxonomy-dependent approaches directly rely on lists and databases of known pathogens, aiming at assigning sequences to taxonomic categories. Read mappers, for example BWA (H. Li & Durbin, 2010) and Bowtie2 (Langmead & Salzberg, 2012) fall into this category. Live mapping approaches such as HiLive and HiLive2 (Lindner et al., 2017; Loka et al., 2019) can even map the reads in real time, as the sequencer is running, leading to a drastic reduction in total analysis time.

In general, read mappers specialize in the computationally efficient alignment of NGS reads to reference genomes with high specificity. For this reason, they are routinely used for the detection of known pathogens, but do not perform well when a sample contains organisms absent from the reference index. Specialized pipelines (Andrusch et al., 2018; Hong et al., 2014) use read mappers in conjunction with additional filtering steps for accurate diagnostics for clinical samples. BLAST (Altschul et al., 1990) offers much more sensitive alignment, appropriate for between-species comparisons, but at the cost of much lower throughput.

Metagenomic profiling tools may also be used as taxonomy-dependent pathogen detection methods. Kraken (Wood & Salzberg, 2014) builds a database of unique 31-mers to assign reads to their corresponding taxonomic units. Ambiguities are resolved by returning the lowest common ancestor. MetaPhlAn2 (Truong et al., 2015) uses around 1 million clade-specific marker genes to detect sequences matching its reference genomes. MicrobeGPS (Lindner & Renard, 2015) identifies potentially unknown organisms present in a metagenomic sample and estimates genomic distances between them and known references. NBC (G. Rosen et al., 2008; G. L. Rosen et al., 2011), a naïve Bayes classifier, is a machine learning based method trained to recognize taxa based on their k-mer frequency profiles.

Taxonomy-agnostic

Taxonomy-agnostic methods strive to predict phenotypes directly from DNA sequences, without performing any taxonomic assignment. They are not entirely independent of taxonomy, however, as they must be trained on available references. This may lead to

bias with regard to the over- and underrepresented taxa in a training set. One goal of the taxonomy-agnostic approaches is to minimize that bias and offer relatively accurate predictions even for novel and divergent sequences. In contrast, the taxonomy-dependent methods are entirely based on the correspondence between a phenotype and a taxonomic classification.

As NBC allows constructing a custom reference database, it may also be used in a taxonomy-agnostic manner. However, it is outperformed by PaPrBaG (Deneke et al., 2017), a random forest approach using a wide range of *k*-mer and peptide-based features. Despite being a read-based method, it can also be used to predict a phenotype from a whole genome. Any long sequence may be fragmented into read-length subsequences. A mean over all the predictions constitutes the final prediction by majority vote. Although this approach is limited to detecting relatively local patterns and cannot use a wider genomic context, it may be useful in practice. Barash et al. (2018) used PaPrBaG, PathogenFinder (Cosentino et al., 2013) and their original tool BacPaCS to predict labels for novel genomes recently deposited in the PATRIC database (Wattam et al., 2017).

2.1.3 Deep neural networks for DNA sequences

Deep learning (LeCun et al., 2015) has been successfully used on genomic data to detect genome accessibility (Kelley et al., 2016) or transcription factor binding sites and disease-associated variants (Alipanahi et al., 2015; Greenside et al., 2018; Quang & Xie, 2016; Zeng et al., 2016; J. Zhou & Troyanskaya, 2015). Budach and Marsico (2018) implemented convolutional neural networks (CNNs) and long short-term memory networks (LSTMs) in a recently published package, pysster.

Identical predictions for a given 'strand' and its reverse-complement can be enforced by passing them both through a network and merging the predictions for each of the strands with a scalar or element-wise operation. Merging may be based for example on the maximum function (Alipanahi et al., 2015; Qin & Feng, 2017) or averaging the results (Quang & Xie, 2019). The resulting networks have a branched architecture resembling Siamese networks.

Shrikumar et al. (2017b) showed that using standard CNNs leads to inconsistent predictions, and that even data augmentation with reverse-complement sequences cannot mitigate the problem. They developed a solution based reverse-complement parameter sharing between pairs of convolutional filters. A limitation of this method is that it is only directly applicable to convolutional networks. Analogous approaches were independently proposed by several groups (Cohen & Welling, 2016; Kopp & Schulte-Sasse, 2017; Onimaru et al., 2020), and extended with Bayesian dropout (R. C. Brown et al., 2018).

2.2 Methods

2.2.1 Data preprocessing

IMG dataset

Similarly to Deneke et al. (2017), we accessed the IMG/M database (I.-M. A. Chen et al., 2019) on April 17, 2018, and followed the procedure described to identify pathogenic and non-pathogenic bacteria with a human host. Briefly, we filtered the database searching for keywords resulting in the unambiguous assignment of a given strain to one of the classes. Previous research successfully used both permanent draft and finished genomes in a similar setting. We decided to include draft genomes in our dataset, too, as their quality is similar to the quality of permanent drafts. For seven species of well-known pathogens (*Campylobacter jejuni*, *Clostridioides difficile*, *Clostridium botulinum*, *Francisella tularensis*, *Listeria monocytogenes*, *Staphylococcus aureus*, and *Staphylococcus epidermidis*), we found between one and two non-pathogenic strains and multiple pathogenic strains; we removed the non-pathogenic strains from further analysis. For *E. coli*, we found multiple strains of both labels. We decided that a study focusing on pathogens should be rather more than less sensitive to this particularly diverse species, especially since some strains may lead to dangerous outbreaks, like the EHEC epidemic in Germany in 2011 (Frank et al., 2011). Therefore, we also removed the non-pathogenic strains of *E. coli* from the final dataset. The procedure yielded a collection of 2,878 strains (2,796 pathogens and 82 non-pathogens) described by data including species names, National Center for Biotechnology Information (NCBI) Bioproject IDs, and class labels. We then linked the Bioproject IDs with GenBank assembly accession numbers found in the GenBank assembly summary file downloaded on April 17, 2018 as well. We downloaded the assemblies and selected one strain per species at random to avoid skewing the model's performance towards species with many sequenced strains available.

Species-level validation

This resulted in a list of 446 species (389 pathogens and 57 non-pathogens), with a genome of a single strain representing each species. We assigned 80% of the species to the training set, and 10% to the validation and test sets each, keeping the ratios between pathogens and non-pathogens the same in all sets. Note that the true distribution of pathogenic and non-pathogenic species is unknown. The observed imbalance is just a result of a bias towards studying pathogens and does not reflect reality. Therefore, using

an imbalanced test set would only propagate the bias of the source database. Since we are equally interested in accurate predictions for both classes, we use a balanced test set.

Using the Mason read simulator with the Illumina error model (Holtgrewe, 2010) and a custom R script, we simulated 10 million single-end Illumina reads per class from the training set and 1.25 million reads per class from the validation set, thus balancing the number of data points in both classes. We simulated 1.25 million paired reads per class from the test set, allowing us to evaluate the classification performance in both the single-end and the paired-end setting. This amounts to 80%, 10%, and 10% of all reads assigned to each of the sets, respectively. Read length was set to 250 base pairs (bp) in all cases, representing what is routinely available using an Illumina MiSeq device; for read pairs, we used a mean fragment length of 600 with a standard deviation of 60.

While this data preprocessing procedure results in a balanced training dataset, the mean coverage of pathogen and non-pathogen genomes is drastically different. To explore an alternative way of solving the class imbalance problem, we also simulated an imbalanced training set, where the total of 20 million training reads was simulated with equal mean coverage from all the training genomes, regardless of their labels. This dataset was then used to train ten networks using a class-weighted loss function. PaPrBaG, constituting the state-of-the-art in machine learning based pathogenicity prediction, does not support error weighting. For BLAST, a method based on sequence homology, this distinction does not apply at all, as its reference database is constructed over whole genomes. However, this does not influence our final results; the class-weighted networks were outperformed by those trained on a balanced set and were not selected as our final models.

Read-level validation

Note that in our primary dataset, we placed separate species in training and validation sets to explicitly force our classifiers to predict pathogenic potentials of reads originating from *novel* species. A simpler approach would comprise simulating the reads first, and then assigning them to the training and validation sets. However, this would lead to solving a similar, but different biological problem (classifying reads originating from *known* organisms). To explicitly test those assumptions, we generated a version of the training and validation set where reads from the same species occur in both training and validation sets. In this setting, the test set remained the same as above, so it would be possible to compare the effects of read-level and species-level validation.

Temporal hold-out data

We accessed the IMG/M database again on January 17, 2019 and preprocessed it as described above. We identified three new species passing the filters applied. All of them were pathogens belonging to the *Pantoea* genus, which is absent from our original dataset. We downloaded the genomes and simulated paired reads as described above. To keep the mean coverage at a similar level to the coverage of the original test set, we simulated 100,000 reads in total. While this supporting dataset is small, it is useful as a case study in conjunction with the results of evaluation on the more diverse main dataset.

Real data case study

To test the performance of our classifier on data from a real sequencing run, we searched the Short Read Archive (SRA) database for sequencing runs performed for a well-known pathogenic species *Staphylococcus aureus*. This species was not present in the training set (it had been randomly assigned to the validation set). We considered runs performed on the Illumina platform with paired-end reads of length 250. Note that in the case of real sequencing reads, the true read length may vary, and some of the reads were significantly shorter. We accessed the archive SRX4814864, originating from a lesion swab from an Italian paediatric hospital (Manara et al., 2018). We downloaded the data from the corresponding run SRR7983698 as unclipped FASTA files.

BacPaCS dataset

Finally, we downloaded all the data used by Barash et al. (2018) in the assessment of their BacPaCS method. We accessed the PATRIC database (Wattam et al., 2017) by the IDs provided by Barash et al. (2018), copying their training and test sets for direct comparability. The original training set consists of 17,811 genomes of pathogens and 3,274 genomes of non-pathogens, while the test set contains 60 pathogens and 40 non-pathogens. Importantly, those genomes do not represent unique species, and the number of strains per species varies greatly. However, following Barash et al. (2018), we treated each strain as a separate entity and used all of them in the analysis. We randomly reassigned 10% of the original training genomes to the validation set, so our BacPaCS training set comprised of 90% of the original. The test set was left unchanged. The read simulation was then performed exactly the same as for the species-level validation.

2.2.2 Reverse-complement networks

DNA encoding and reverse-complementarity

We use distributed orthographic representations of DNA sequences, a method based on one-hot encoding of every nucleotide in the sequence. Namely, a sequence is converted into a 2D binary tensor, where one dimension represents the position of a given nucleotide in the sequence, and the other represents the one-letter nucleotide code as a one-hot encoded vector. For example, adenine is represented by the vector (1,0,0,0) and thymine by (0,0,0,1). Unknown nucleotides may be encoded as (0,0,0,0). Note that reversing the sequence tensor along both axes results in the reverse-complement.

We design networks with two separate branches processing each of the input orientations, respectively. Each of the branches consists of identical layers, and all the parameters are shared between each pair of layers. We propose two variants of this architecture. In the *full* RC-networks, input to the deeper layers consists of concatenated outputs of both the forward and reverse-complement versions of the previous layer. The output of the RC layer is flipped before concatenating, so that the channels i and $n-1-i$ in the final tensor correspond to the same feature on opposing strands. Note that in this case, a *full* RC-CNN is equivalent to the RC-CNNs proposed by Shrikumar et al. (2017b). In the second variant, dubbed a *Siamese* RC-architecture, each of the branches functions separately before the merging layer. That means that the input to a deeper layer is just the output of the previous layer in a branch. We tested both variants using CNNs, bidirectional LSTMs and hybrid networks with both convolutional and LSTM layers, but they are in principle compatible with any other neural architecture.

We note that forward and reverse-complement representations can be merged by any element-wise operation on the 1D output tensors. Similarly to Shrikumar et al., 2017b, we place the dense layers (and the output layer) after representation merging. Apart from summation, we considered two alternative merging functions (note that averaging and adding the representation vectors are essentially equivalent). The *max* function implements the Gödel t-conorm, corresponding to the OR operation in Gödel fuzzy logic. Even though the activations are not restricted to the interval [0,1], high output values can be interpreted as finding a motif on either of the two strands. The Hadamard product is the product t-norm corresponding to the AND operation in the product fuzzy logic. Here, high values may be understood as finding a motif on both strands at the same time. During hyperparameter tuning, we considered all three methods of strain representation merging. Although the differences were small, summation of forward and reverse-complement feature vectors yielded the best performance overall.

Species-level and paired reads predictions

One of the major challenges of pathogenic potential prediction from single reads is the lack of biological context. However, if all the reads in a sample originate from the exact same organism, we can predict the pathogenic potential of that organism by a majority vote. In the context of probabilistic estimates of the class label (returned by both PaPrBaG and our neural networks), we can implement that as a simple mean over predictions for all the individual reads. For BLAST, we can just assign the label predicted for the majority of reads.

Building upon this idea, we can boost read-based performance if we consider read pairs, assumed to originate from the same organism even in metagenomic samples. To this end, we average predictions for the corresponding pairs in our test set. The classifiers may still predict pathogenic potentials for isolated sequences if so desired. We can integrate binary predictions (e.g. returned by BLAST), taking into account the missing and conflicting predictions for some of the reads. We treat missing predictions as undefined values and implement the *accept anything* operator of ternary logic. It returns a positive label if and only if one of the input values is positive, and the other is not negative. Conversely, it returns a negative label if and only if one of the input values is negative, and the other is not positive. The result is undefined when both inputs are undefined, or in the case of conflicting input values.

Hyperparameter tuning

We used Keras 2.2.4 and TensorFlow 1.12. We tested a total of 243 different architectures. All were initialized with He weight initialization (He et al., 2015) or Glorot initialization (Glorot & Bengio, 2010) for the recurrent and feedforward layers respectively, and trained with the Adam optimizer (Kingma & Ba, 2014). We used dropout regularization (Srivastava et al., 2014), including input dropout (interpreted as setting a random fraction of nucleotides to Ns). We also applied batch normalization (Ioffe & Szegedy, 2015) to some of the RC-CNNs and tested the effect of adding an L_2 regularization term in RC-LSTMs . In addition, we trained traditional CNNs and LSTMs without RC parameter sharing, equivalent to the networks that can be implemented with the pysster package (Budach & Marsico, 2018). Finally, we selected the best CNN and the best LSTM model, and prepared a simple ensemble classifier by averaging the predictions of those two models. The selected RC-CNN model consists of 2 convolutional and 2 dense layers with 512 and 256 units, respectively; it was trained with an input dropout rate of 0.25 and without batch-normalization. The best RC-LSTM has one layer of 384 units, and

was trained with the input dropout rate of 0.2. For a more detailed description of the tuning process, see section A.1.

Read-level validation

Generalization from one set of reads to another set of reads should be relatively easy when both sets originate from the same species. However, we expected that even very high read-level validation accuracy would not translate into high test accuracy (see section 2.2.1). After the primary tuning procedure described in the previous sections, we selected the architecture obtaining the highest *training* accuracy, as we assumed little overfitting would be seen at the validation stage in this setting. We trained the network using read-level validation and evaluated it on our primary test set containing reads from species absent in both validation and training sets.

2.2.3 Benchmarking

PaPrBaG

To benchmark our method against the state-of-the-art in pathogenic potential prediction, we trained PaPrBaG random forests on our training set and evaluated them on our test set. We used two different feature settings for PaPrBaG. The original authors extracted more than 900 features from each of the reads, including *k*-mer frequencies and a selection of amino-acid and peptide features inferred by a 'least STOPs' heuristic. However, they show that the translation-based features contribute relatively little to the final classification decision. Therefore, we decided to use both the original feature set and a reduced set comprising the DNA features only.

BLAST

BLAST was shown (Deneke et al., 2017) to achieve the best read-by-read performance among alternatives to machine learning approaches for predicting pathogenic potential. It outperformed both the mapping-based Bowtie2 and Pathoscope2, which builds on the former, as well as two different variants of the *k*-mer based Kraken (Hong et al., 2014; Langmead & Salzberg, 2012; Wood & Salzberg, 2014). All three failed to classify most of the reads. Furthermore, BLAST achieved better classification results than a naïve Bayes classifier (NBC) with a word length of 15 (G. L. Rosen et al., 2011).

For each test read, we performed a search against the database containing all the training genomes and took a label of the best hit for each of the test reads as a predicted label for that read. We also tested a variant of this approach using all the available strains

of the training species to build the database. In both cases, we use the discontiguous megablast task (`-task dc-megaBLAST`), a cutoff E-value of 10 and the default parameters.

BacPaCS

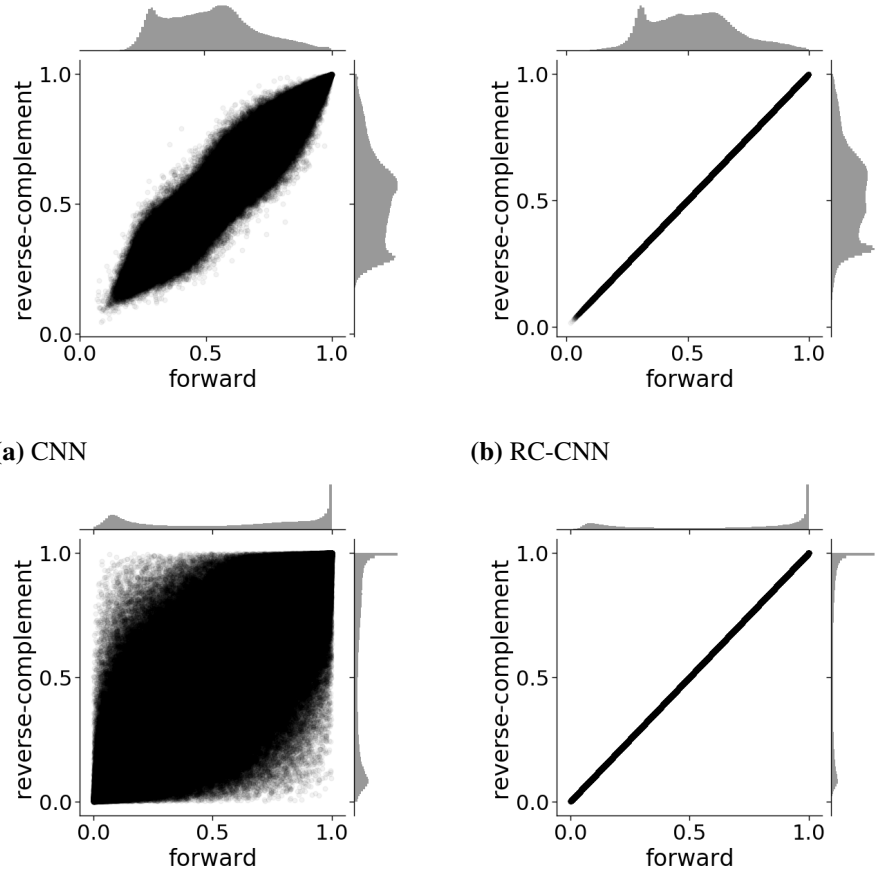
Finally, we compared our approach to BacPaCS, Pathogenfinder, and PaPrBaG using the original BacPaCS test dataset. Without any further tuning, we selected one CNN and one LSTM architecture which worked best on our data. We retrained them both using the BacPaCS training data. Since the BacPaCS dataset treats strains, not species, as the primary entities (and is used by the authors to predict labels for new strains of species *present* in the training set), the classifier designed by Barash et al. (2018) can be treated as specialized in predicting pathogenic potentials for *known* species. Therefore, we assumed that the architecture used for read-based validation would yield high performance, and retrained it on the BacPaCS data as well. We evaluated the networks in a single-species sample setting and compared the results to the originally presented performance metrics for BacPaCS and Pathogenfinder.

Assessing PaPrBaG, Barash et al. (2018) used the random forests trained on the original PaPrBaG dataset. This may lead to inaccurate estimates of the classification error, as the labels they mined from the PATRIC database differ from the labels that the original PaPrBaG forests used. This problem is exacerbated by the imbalance between the number of strains per species in the BacPaCS test dataset. For example, strains of *Acinetobacter baumannii* alone constitute 30% of the non-pathogens in the set. *Fusobacterium periodonticum* amounts to another 10%. Importantly, both of those two species were treated as pathogens in the original PaPrBaG dataset, but are assigned a non-pathogenic label based on the PATRIC metadata. One should therefore expect that the original PaPrBaG forests will predict wrong labels for a significant fraction of the test set if not retrained using the labels extracted by Barash et al. (2018) for their training set. Therefore, we retrained PaPrBaG on the original BacPaCS dataset for an accurate comparison to our networks.

2.3 Results

2.3.1 Reverse-complementarity constraint

We show predictions for each read in the test set against predictions for its reverse-complement in Figure 2.1. Note that the reads are simulated from either of the strands at random. Even though the predicted pathogenic potentials are highly correlated



(a) CNN **(b)** RC-CNN

(c) LSTM **(d)** RC-LSTM

Figure 2.1: Prediction distributions for traditional (2.1a, 2.1c) and reverse-complement networks (2.1b, 2.1d). Predictions for each of the reads (x-axis) are plotted against the predictions for reverse-complements of the same reads (y-axis). The RC-architectures guarantee identical predictions for both strands, which can be strikingly divergent for the traditional LSTM.

(Spearman's $\rho > 0.999$, $p < 10^{-6}$) and identically distributed ($p > 0.999$, two-sample KS test) for both orientations, they are not equal. The differences are striking, especially for the LSTM model. For the RC neural networks, the predictions are guaranteed to be exactly the same. Interestingly, the prediction distributions are very different for the CNN and LSTM models. We can only speculate that this could be related to the specific properties of the architectures – while the CNNs assume translational invariance and hence focus on localized sequence motifs, the LSTMs are able to also detect long-range relationships between more distant fragments of a read.

2.3.2 Pathogenic potential prediction from NGS reads

Single reads

We present the results of evaluation on single NGS reads in Table 2.1. The values given are calculated over the complete test set, treating each read as a separate entity. Precisely, we can calculate the performance measures for the sets of 'left' and 'right' mates separately and then compute the mean for any given measure. Results obtained for the 'left' and 'right' half of the set did not differ from those presented in Table 2.1 by more than 0.001.

As previously shown (Deneke et al., 2017), BLAST fails to classify some of the reads at all. To compare its performance to the machine learning approaches, we define accuracy as the ratio of correct predictions to the number of all data points in a set. Therefore, missing predictions are counted as false positives or false negatives depending on the ground truth label. The fraction of classified reads is presented in the last column.

The neural networks clearly outperform both BLAST and PaPrBaG in terms of prediction accuracy. The traditional deep learning architectures performed worse than their reverse-complement counterparts (data not shown). However, the differences in validation accuracy were below 0.01. Figure 2.1 compares the prediction distributions of standard and RC-networks, showing the importance of employing the RC models, especially for LSTMs .

The RC-CNN model achieves the highest positive predictive value (precision), and the RC-LSTM is the most sensitive. The RC-CNN+LSTM ensemble model aims at a trade-off between those two performance measures. Integrating signals contained in both mates significantly boosts the performance of all the evaluated methods (Table 2.2). In this case, the RC-CNN achieves the highest accuracy. Compared to the previous state-of-the-art (PaPrBaG with original settings, single reads only), it cuts the error rate almost in half (21.9% vs. 11.3%).

A *full* RC-CNN with 2 convolutional and 2 dense layers of 512 and 256 units, batch normalization and no input dropout achieved the highest *training* accuracy and was used for read-level validation. As expected, it performs much worse, even though it achieved an impressive validation accuracy of 0.956. Somewhat surprisingly, it still outperforms both PaPrBaG and BLAST. It is important to note that for all the other methods, the validation accuracy was much lower than test accuracy. It ranged from 57.4 (BLAST) to 78.4 (RC-LSTM) and 78.7 (RC-CNN+LSTM). We hypothesize that this is because the species randomly assigned to the training set are more closely related to the species in the test set than to those in the validation set. We comment on this problem further in the Discussion.

Table 2.1: Classification performance on single reads. $PaPrBaG_{DNA}$ is a variant of PaPrBaG using DNA-based features only. By default, we use the training set to build the BLAST reference database. $BLAST_{All}$ uses all available strains of the training species. $RC\text{-}CNN_{Read\text{-}val}$ was trained using read-level validation. RC-CNN+LSTM is the average of RC-CNN and RC-LSTM predictions. Acc. – accuracy, Prec. – precision, Rec. – recall, Spec. – Specificity, AUC – area under the ROC curve, AUPR – area under the precision-recall curve, Pred. – prediction rate.

Classifier	Acc.	Prec.	Sens.	Spec.	AUC	AUPR	Pred.
PaPrBaG	78.1	75.6	83.0	73.2	85.9	84.4	100.0
$PaPrBaG_{DNA}$	78.2	75.2	84.1	72.3	86.1	84.7	100.0
BLAST	66.5	**84.5**	81.8	51.2	n/a	n/a	75.1
$BLAST_{All}$	67.2	83.5	83.8	50.7	n/a	n/a	76.5
$RC\text{-}CNN_{Read\text{-}val}$	79.0	73.3	91.1	66.8	88.0	85.8	100.0
RC-CNN	84.4	82.5	87.4	81.5	91.8	91.8	100.0
RC-LSTM	83.8	77.8	**94.6**	73.0	91.6	88.8	100.0
RC-CNN+LSTM	**84.8**	79.4	94.1	75.6	**92.9**	**92.2**	100.0

Time performance

The prediction speed is difficult to compare, as PaPrBaG can only run on a central processing unit (CPU), and our neural networks are most efficiently used on graphics processing units (GPUs). Both can be trivially parallelized depending on the number of devices available. We measured the time PaPrBaG needed for a complete prediction task (feature extraction and prediction itself) on a single Intel(R) Xeon(R) E7-4890 v2 CPU. The original version of PaPrBaG was able to classify 107 reads/s, and 47% of that time was used for feature extraction. The DNA-only version classified up to 167 reads/s, and most of the speed-up came from the reduction of feature extraction time (which was 25% of the total elapsed time). The classification performance was nearly identical.

The reverse-complement neural networks were tested on a single, consumer-level GPU (RTX 2080 Ti). We measured the time of prediction from sequences already converted to binary tensors, as the conversion step may be performed independently on separate CPU devices. The selected RC-LSTM model can predict up to 1817 reads/s and the RC-CNN reaches a speed of 4010 reads/s. Using this rapid architecture, a million-read sample may be processed in just over 4 minutes, even on a desktop computer.

Single-species samples

The evaluation scenario presented in Table 2.3 models sequencing a pure, single-species isolate, when it can be assumed that all reads originate from the same genome. Af-

Table 2.2: Classification performance on read pairs.

Classifier	Acc.	Prec.	Rec.	Spec.	AUC	AUPR	Pred.
PaPrBaG	82.2	78.9	87.8	76.5	89.8	88.5	100.0
$\text{PaPrBaG}_{\text{DNA}}$	82.2	78.5	88.7	75.7	90.1	89.0	100.0
BLAST	70.0	84.1	86.6	53.5	n/a	n/a	79.4
$\text{BLAST}_{\text{All}}$	70.4	82.9	88.3	52.5	n/a	n/a	80.5
RC-CNN	**88.7**	**86.8**	91.3	**86.1**	**94.8**	**94.7**	100.0
RC-LSTM	86.6	80.4	**96.7**	76.4	93.6	90.9	100.0
RC-CNN+LSTM	87.8	82.4	96.2	79.4	**94.8**	94.0	100.0

ter averaging the pathogenic potentials over whole genomes, BLAST, PaPrBaG, and RC-CNN performed equally well, with just one false positive and one false negative. The RC-LSTM and RC-CNN+LSTM ensemble models predict one more false positive, which is reflected in their drastically lower specificity. This is in line with the previous results suggesting that those models are more sensitive, even though they do not manage to limit the number of false negatives to zero. Note that the set contained 39 pathogen and only 6 non-pathogen genomes. A bigger and more balanced dataset would allow more reliable error estimates, but collecting more data that could be used in a rigorous, automated data preprocessing workflow poses a non-trivial challenge. It is nevertheless important to evaluate prediction accuracy also on single-organism samples, as this is one of the possible applications of the proposed method (Barash et al., 2018; Deneke et al., 2017). Validation balanced accuracy ranged from 80.8 (All-strains BLAST) to 89.1 (BLAST, RC-LSTM). Interestingly, the RC-LSTM and RC-CNN+LSTM ensemble models achieved higher balanced accuracy on the validation set than on the test set. This may reflect the different evolutionary distances between the members of each of the sets, as discussed for the single-read predictions.

2.3.3 Temporal hold-out

For the temporal hold-out test, we predicted labels for reads originating from *Pantoea brenneri*, *Pantoea septica*, and *Pantoea conspicua*, which were labelled pathogenic based on the metadata extracted from the IMG/M database. They were not available when we prepared our training, validation, and primary test sets. Species-wide, the labels were predicted correctly for all three genomes by both BLAST and the reverse-complement networks (Table 2.4). The RC-LSTM and the RC-CNN+LSTM ensemble model performed best, but the RC-CNN outperformed BLAST as well. PaPrBaG was

Table 2.3: Classification performance on single-species samples. Balanced accuracy (BAcc.) is the mean of recall (Rec.) and specificity (Spec.).

Classifier	Acc.	BAcc.	Prec.	Rec.	Spec.	AUC	AUPR
PaPrBaG	**95.6**	**90.4**	**97.4**	97.4	**83.3**	93.2	**98.8**
$\text{PaPrBaG}_{\text{DNA}}$	**95.6**	**90.4**	**97.4**	97.4	**83.3**	93.2	**98.8**
BLAST	**95.6**	**90.4**	**97.4**	97.4	**83.3**	n/a	n/a
$\text{BLAST}_{\text{All}}$	**95.6**	**90.4**	**97.4**	97.4	**83.3**	n/a	n/a
RC-CNN	**95.6**	**90.4**	**97.4**	97.4	**83.3**	**94.0**	**98.9**
RC-LSTM	93.3	82.1	95.0	97.4	66.7	89.7	97.8
RC-CNN+LSTM	93.3	82.1	95.0	97.4	66.7	92.3	98.5

Table 2.4: Temporal hold-out, read pairs.

Classifier	Acc.	Prec.	Rec.	Pred.
BLAST	88.1	100.0	88.1	88.3
$\text{BLAST}_{\text{All}}$	89.5	100.0	89.5	89.6
RC-CNN	98.4	100.0	98.4	100.0
RC-LSTM	**99.2**	100.0	**99.2**	100.0
RC-CNN+LSTM	**99.2**	100.0	**99.2**	100.0

not used here, as it consistently underperformed in comparison to the deep learning approaches in the previous tests. As mentioned in section Data preprocessing, this dataset is too small to yield reliable error estimates, and the temporal hold-out should only be interpreted as a case study supporting the other results. The *Pantoea* genus was absent from the training database when the training set was compiled; accurate predictions for a novel genus (while trained at the species level) suggest successful generalization.

2.3.4 Real-data results

One of the defining characteristics of real sequencing runs is that the reads generated are not all of the same length. Since the neural networks require a constant input length, we pad the sequences shorter than 250bp with zeroes (interpreted as Ns). If any sequence is longer than this threshold, it is trimmed. This may be a problem for the CNNs with average pooling, as multiple zero-entries significantly lower the activations after a pooling layer. Therefore, we expected those architectures to achieve lower accuracy than the RC-LSTMs on a real dataset. Note that this problem does not apply to RC-CNNs with max pooling. However, those performed worse than their average-pooling counterparts

Table 2.5: Performance on real data, read pairs.

Classifier	Acc.	Prec.	Rec.	Pred.
BLAST	74.6	100.0	74.6	75.0
$BLAST_{All}$	78.7	100.0	78.7	79.0
RC-CNN	90.0	100.0	90.0	100.0
RC-LSTM	**98.4**	100.0	**98.4**	100.0
RC-CNN+LSTM	98.2	100.0	98.2	100.0

in the tuning and validation step. As they essentially work as motif detectors, we suspect that they are more prone to overfitting. We present the results in Table 2.5.

Although the RC-CNN suffers from lower accuracy compared to the RC-LSTM and RC-CNN+LSTM ensemble models, it still outperforms BLAST by a large margin. Based on the prediction speed we can name the RC-CNN our *rapid* model, and the RC-LSTM our *sensitive* model. Their predictions may be aggregated with the RC-CNN+LSTM ensemble model, with almost no additional computation, for a boost in sensitivity at a smaller precision cost.

2.3.5 BacPaCS dataset results

We present the results of the evaluation performed on the BacPaCS dataset in Table 2.6. The RC-networks outperform all of the other methods in terms of balanced accuracy. PaPrBaG's specificity is much higher after retraining (see section 2.2.3). This is also reflected in balanced accuracy, which in this case is even higher than BacPaCS's. The RC-CNN architecture selected based on our primary dataset turns out to be the most specific architecture when trained on the BacPaCS dataset. As expected, the best overall performance is achieved by an RC-CNN with batch normalization and without input dropout, as observed for read-level validation (see section 2.2.2).

2.4 Discussion

2.4.1 Evolutionary distances and pathogenicity

Given the difficulty of the task, the reverse-complement networks offer impressive performance. They are able to predict a high-level, abstract phenotype (namely, pathogenic potential) from isolated reads, without any additional biological context. They also outperformed the traditional deep learning architectures in the tuning and validation step. However, all of the evaluated tools show a noticeable gap between validation and test

Table 2.6: Performance on the BacPaCS dataset. Balanced accuracy is the mean of recall and specificity.

Classifier	Balanced accuracy	Recall	Specificity
BacPaCS	70	92	48
Pathogenfinder	46	63	28
PaPrBaG (Barash et al., 2018)	53	**100**	5
PaPrBaG (retrained)	77	72	83
RC-CNN	81	70	**93**
RC-LSTM	80	75	85
RC-CNN+LSTM	81	72	90
RC-CNN+BN	**84**	78	90

accuracy. We assume that this is a result of the species-level division of the data into the training, validation and test sets – it must be more difficult to predict correct labels for the validation data than for the test data. We hypothesize that the species assigned to the test set must be more similar (in terms of evolutionary distance) to the training species than the validation species. This discrepancy would be therefore a result of treating all the species as separate, independent entities, as if the sequence similarities between individual genomes (and the corresponding reads) were uniform. This is not biologically correct; closely related species have similar genomes, and horizontal transfer of genetic elements (including virulence factors) introduces local similarities.

Pathogenic and non-pathogenic strains may belong to the same species. However, the extracted labels are actually consistent among individual strains for a vast majority of the species in the IMG/M database. Seven well-known pathogens had between one and two non-pathogenic strains that were ignored in the further analysis. We select one strain per species to avoid skewing the performance towards species with more strains available. As expected, we found multiple *E. coli* strains of both classes. We decided to only consider pathogenic ones, so we assume our models to classify any *E. coli* sample as exhibiting pathogenic potential. While this may be seen as a major limitation of the proposed method, it only affects a single species in the dataset. It also reflects the great phenotypic diversity of *E. coli*. Importantly, this problem does not apply to the models trained on the BacPaCS dataset, which includes multiple independently labelled strains per species. DeePaC performs very well also in this setting.

We resort to the oversimplification of treating species as independent since balancing the read sets to account for sequence similarity is not trivial. The same approach was used by PaPrBaG (Deneke et al., 2017). In turn, BacPaCS (Barash et al., 2018) uses individual

strains as primary biological entities, which leads to similar problems. Nevertheless, both approaches do achieve satisfactory results. In fact, Deneke et al. (2017) have shown that PaPrBaG outperforms taxonomy-dependent methods (BLAST, Kraken, Bowtie2 and Pathoscope2) in two challenging cases: when species belonging to the same genus have different phenotypes, and when a genus is completely absent from the training set (as in our temporal hold-out). This suggests that read-based, taxonomy-agnostic methods like DeePaC and PaPrBaG are actually more robust to the oversimplified assumption of uniform evolutionary distances between species. A future outlook could include investigating in detail how sequence similarities and differences in labels within each taxonomic rank influence the predictions, as well as which parts of a given genome lead to the most confident predictions.

More exact estimates of the classification error could be obtained using nested cross-validation, but this was not computationally feasible – a single training epoch may take up to 6 hours on a state-of-the-art GPU for some of the most demanding architectures. We trained multiple networks in parallel on multiple nodes of a cluster, but performing even simple 10-fold cross-validation would require a drastic reduction of the number of architectures covered. This would be undesirable, as no baseline deep learning model for pathogenic potential prediction was available prior to this study. Multiple combinations of hyperparameter values had to be tested. Therefore, we assume that the accuracy of BLAST, a sensitive alignment-based approach, reflects sequence similarities (and presumed evolutionary distances). This means that a method more accurate than BLAST generalizes well. In addition, we test the classifiers on independent data. Our approach consistently outperforms both BLAST and PaPrBaG. It also fares better than BacPaCS, Pathogenfinder, and PaPrBaG when tested on the original BacPaCS data.

2.4.2 Predictions from single reads

The results above may be counter-intuitive. Virulence factors are often encoded on pathogenicity islands or mobile genetic elements. While the genome assemblies used in this study often include plasmids (which can hence be detected by the models), it is impossible to guarantee that none was missed. Moreover, the same organism may cause disease in one person and not in the other – for those very reasons, we can only predict pathogenic potentials, not a pathogenic phenotype directly. All this context is unavailable in an NGS read. It is not even possible to reliably detect open reading frames; comparing PaPrBaG to $\text{PaPrBaG}_{\text{DNA}}$ proves that attempting to infer putative peptide sequences only introduces noise. Nevertheless, the evaluation clearly shows that the read-based approaches do work. Surprisingly, DeePaC outperforms BacPaCS, which

predicts pathogenic potentials from whole proteomes, treating the most relevant genes as input features (Barash et al., 2018). Apparently, predicting pathogenic potentials based on isolated reads and aggregating the predictions with a simple mean results in accurate phenotype predictions – also when multiple strains of each species are considered.

This could be related to the texture-bias of CNNs. Brendel and Bethge (2019) have recently shown that splitting an image into receptive fields as small as 17×17 px, and subsequent averaging of the class predictions for each of the patches results in high accuracy predictions on the ImageNet dataset. A similar effect may be seen in the pathogenic potential prediction task. A genome is split into short reads, but the local sequence patterns are sufficient to predict a phenotypic label even though establishing a mechanistic link between a read and the phenotype is improbable. Averaging the results leads to confident predictions for whole genomes in single-organism samples.

2.4.3 The definition of a pathogen

It is important to define human pathogens in a reliable and consistent way. However, it is also by no means a trivial task. We do not differentiate between the level of danger individual organisms may pose – dangerous, biosafety level 3 select-agents are put in the same category as opportunistic pathogens causing relatively mild and easily-treated infections. Classifying the risk posed by unrecognized DNA at a higher resolution is an interesting avenue for further research; it is however likely to be challenging due to a comparatively small number of very dangerous bacterial species. We recognize that what constitutes a pathogen is an open problem, and different definitions may be interesting for different purposes. For processing big datasets, the labels must be extracted automatically using handcrafted rules compatible with a particular source database. It is therefore immensely important to be aware of the underlying assumptions, as they will be reflected in the final classification decisions of a trained model. We note that comparing models trained using incompatible labels may result in unreliable error estimates, especially if organisms for which the labels differ constitute a substantial fraction of the test set. We recommend sharing the metadata describing the training organisms along the trained models. For this study, they are available at https://gitlab.com/rki_bioinformatics/DeePaC.

2.4.4 A flexible framework for RC-constrained classification

DeePaC is easily extensible and may be used as a generalized framework for building reverse-complement neural architectures beyond the applications described here. It is not constrained to the read length used here or the pathogenic potential prediction task; any

label or value may be used as a prediction target. For predicting a categorical phenotype directly from NGS reads, we suggest that the target label should be a relatively abstract, general feature. Phenotypes dependent on single genes or small sets of genes will most probably be very difficult to predict from whole-genome sequencing reads. The variant of the RC configuration (*full*, *Siamese* or *none*) may be easily switched, so the framework can also be applied to tasks where the RC-constraint is not important. RNA sequences can be analyzed, provided that they are converted to binary tensors using the appropriate alphabet.

2.4.5 Conclusions

We show that the RC-networks outperform the previous state-of-the-art on both simulated and real sequencing data, accurately predicting pathogenic potentials from isolated NGS reads. The trained models can be used to predict pathogenic potentials for unknown, unrecognized and novel (e.g. synthetic) DNA sequences with a simple script, also if they do not originate from an Illumina sequencing run. It is also possible to filter a read set based on a user-defined threshold to select only the reads for which a network yields confident predictions. This may be used as a crude pre-filter, where the reads with the highest pathogenic potential may be further investigated in downstream analysis. The code, the models trained, and example configuration files are available at https://gitlab.com/rki_bioinformatics/DeePaC.

3 Viral host range prediction and interpretability

Summary

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 NGS 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 also be applied 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 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.

This chapter is based on Bartoszewicz, Seidel, and Renard (2021a), which is a joint work with Anja Seidel and Bernhard Y. Renard. A detailed description of the authors' contributions can be found in section Thesis outline.

3.1 Background

3.1.1 Motivation

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 (Calvignac-Spencer et al., 2014). 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 (Trappe et al., 2016; Vouga & Greub, 2016). 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 (Leendertz et al., 2016), 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 (Calistri & Palù, 2015; Lecuit & Eloit, 2014), 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 (Andrusch et al., 2018). 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 (Herfst et al., 2012; Imai et al., 2012). A possibility of modifying coronaviruses to enhance their virulence triggered calls for a moratorium on this kind of research (Lipsitch & Inglesby, 2014). Synthesis of an infectious horsepox virus closely related to the smallpox-causing Variola virus (Noyce et al., 2018) caused a public uproar and calls for intensified discussion on risk control in synthetic biology (Thiel, 2018).

3.1.2 Current tools for host range prediction

Several computational, genome-based methods exist that allow predicting 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 R. A. Edwards et al. (2016). Prediction of eukaryotic host tropism (including humans) based on known protein sequences was shown for the influenza A virus (Eng et al., 2014). Support-vector machines based on word2vec representations were shown to outperform

homology searches with BLAST and hidden Markov models in the same task, but lost their advantage when applied to nucleic acid sequences directly (Xu et al., 2017). Two recent studies employ *k*-mer based, kNN classifiers (H. Li & Sun, 2018) and deep learning (Mock et al., 2020) 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) (Gałan et al., 2019) uses logistic regression and SVMs 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 3,000 nucleotides. This is incompatible with modern metagenomic 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 (Babayan et al., 2018).

Z. Zhang et al. (2019) designed several classifiers explicitly predicting whether a new virus can potentially infect humans. Their best model, a kNN 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 also worked with 150bp-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').

3.1.3 Deep learning for genomics

While DNA sequences mapped to a reference genome may be represented as images (Poplin et al., 2018), 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. 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 (Rizzo et al., 2016) and coding sequences of HIV for the drug resistance prediction task (Löchel et al., 2020). In this study, we use one-hot encoding with Ns as zeroes, which was previously shown to perform well for raw NGS reads (Bartoszewicz et al., 2020) and abstract phenotype labels.

CNNs and LSTMs have been successfully used for a variety of DNA-based prediction tasks. Early works focused mainly on regulation of gene expression in humans (Alipanahi et al., 2015; Kelley et al., 2016; Quang & Xie, 2016; Zeng et al., 2016; J. Zhou & Troyanskaya, 2015), which is still an area of active research (Avsec, Weilert, et al.,

2021; Greenside et al., 2018; Nair et al., 2019). 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 (Mock et al., 2020) and to predict pathogenic potentials of novel bacteria (Bartoszewicz et al., 2020). DeepVirFinder (Ren et al., 2020) and ViraMiner (Tampuu et al., 2019) 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. (2019).

Interpretability and explainability of deep learning models for genomics are crucial for their widespread 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 (Alipanahi et al., 2015) to create a position weight matrix (PWM) which can be visualized as a sequence logo (Crooks et al., 2004; Schneider & Stephens, 1990). Direct optimization of input sequences is problematic, as it results in generating a dense matrix even though the input sequences are one-hot encoded (Lanchantin et al., 2016; Lanchantin et al., 2017). This problem can be alleviated with Integrated Gradients (Jha et al., 2020; Sundararajan et al., 2016) or DeepLIFT, which propagates activation differences relative to a selected reference back to the input, reducing the computational overhead of obtaining accurate gradients (Shrikumar et al., 2017a). If the bias terms are zero and a reference of all-zeros is used, the method is analogous to layer-wise relevance propagation (Bach et al., 2015). DeepLIFT is an additive feature attribution method, and may used to approximate Shapley values if the input features are independent (Lundberg & Lee, 2017). TF-MoDISco (Shrikumar et al., 2019) uses DeepLIFT to discover consolidated, biologically meaningful DNA motifs (transcription factor binding sites).

3.1.4 Contributions

In this paper, we first improve the performance of read-based predictions of the viral host (human or non-human) from NGS sequencing reads. We show that reverse-complement (RC) neural networks (Bartoszewicz et al., 2020) significantly outperform both the previous state-of-the-art (Z. Zhang et al., 2019) and the traditional, alignment-based algorithm – BLAST (Altschul et al., 1990; Camacho et al., 2009), which constitutes a gold standard in homology-based bioinformatics analyses. We show that defining the negative (non-human) class is non-trivial 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 non-human 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 serious disease. To show that the method can also be used for other biological problems, we investigated the networks trained by Bartoszewicz et al. (2020) 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 (F. Wu et al., 2020).

3.2 Materials and Methods

3.2.1 Data collection and preprocessing

VHDB dataset

We accessed the Virus-Host Database (Mihara et al., 2016) 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 non-reference genomes are available, but considering multiple genomes per virus would skew the classifiers' performance towards the more frequently resequenced ones.

The original dataset contained 14,380 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 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 Z. Zhang et al. (2019), we filtered out viroids and satellites, which are classified as subviral agents and not *bona fide* viruses (King et al., 2012; Lefkowitz et al., 2018). 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, non-human hosts exist. In total, the dataset contained 9,496 viruses (grouped in 7503 species), including 1,309 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 non-human viruses into account, including bacteriophages (bacterial viruses). This is especially important, as most of the 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 fungal viruses), or only animal-infecting viruses. This amounts to four nested host sets: 'All' (8,187 non-human viruses, 7110 species), 'Eukaryota' (5,114 viruses, 4275 species), 'Metazoa' (2,942 viruses, 2351 species) and 'Chordata' (2,078 viruses, 1530 species). Auxiliary sets containing only non-eukaryotic viruses ('non-Eukaryota'), non-animal 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 larger test sets. The same applied to other taxonomic levels, as well as the training and validation sets wherever applicable.

Read simulation

We simulated 250bp long Illumina reads following a modification of a previously described protocol (Bartoszewicz et al., 2020) and using the Mason read simulator (Holtgrewe, 2010). 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 (Bartoszewicz et al., 2020; Deneke et al., 2017). 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 exact 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 into the effects of negative class definition on the prediction performance.

Human blood virome dataset

Similarily to Z. Zhang et al. (2019), we used the human blood DNA virome dataset (Moustafa et al., 2017) to test the selected classifiers on real data. We obtained 14,242,329 reads of 150bp 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 (Gorbalenya et al., 2020). 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 (Gorbalenya et al., 2020). This is separate from its putative taxonomic 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 (Gorbalenya et al., 2020; King et al., 2012; Lefkowitz et al., 2018; Simmonds & Aiewsakun, 2018). In most cases, species are perceived as human constructs rather than biological entities, and host range often is explicitly one of the defining features (Gorbalenya et al., 2020; Van Regenmortel, 2018), 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 (Gorbalenya et al., 2020). 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 (Bartoszewicz et al., 2020), 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 non-redundant. At the same time, redundancy below this level (i.e. multiple instances of the same virus) must be eliminated from the dataset to ensure the 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 Z. Zhang et al. (2019). 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 modelled by selecting a single representative genome per species (Bartoszewicz et al., 2020). 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 section 3.3.2).

3.2.2 Training

We used the DeePaC package (Bartoszewicz et al., 2020) 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 (section A.2, Figure A.1). 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 Ns. 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 250bp (our data, modelling a sequencing run of an Illumina MiSeq device) and 150bp 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 100bp of each read were turned to zeros, mocking a shorter sequencing run while preserving the error model.

Then, we retrained the CNN which had performed best on the original dataset. Since in principle, the Human Blood Virome dataset should not contain viruses infecting non-human Chordata, a 'Chordata'-trained classifier was not used in this setting.

3.2.3 Benchmarking

We compare our networks to the kNN classifier proposed by Z. Zhang et al. (2019), 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 (Moustafa et al., 2017) for an unbiased comparison.

We trained the classifier on the 'All' dataset as described by the authors, i.e. using non-overlapping, 500bp-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 labelled 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 (H. Li & Durbin, 2010) and Bowtie2 (Langmead & Salzberg, 2012)), the indices were constructed analogously. Kraken (Wood & Salzberg, 2014) was previously shown to perform worse than both BLAST and machine learning when faced with the read-based pathogenic potential prediction task for novel bacterial species (Deneke et al., 2017). 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 (Deneke et al., 2017). Therefore, we focus on alignment-based approaches as the most accurate alternative to machine learning in this context.

Note that both alignment and kNN 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. (2020), 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. As the NGS mappers are very precise by design, we treat all obtained matches as relevant and count them separately, but still propagate the label if only one read in a pair was successfully aligned. Missing predictions lower both true positive and true negative rates.

3.2.4 Precision for conflicting predictions

Further, because of frequent conflicting predictions for each of the mates, precision (positive predictive value, PPV) is not trivial to define for read pair classification with kNN. In the binary classification context, PPV it is often defined as (Equation 3.1)

$$PPV = \frac{TP}{TP + FP} \tag{3.1}$$

where TP is the number of true positives, and FP is the number of false positives. In a standard binary classification task, this can also be expressed as (Equation 3.2)

$$PPV = \frac{TP}{TP + N - TN} = \frac{TPR \times Prev}{TPR \times Prev + (1 - TNR) \times (1 - Prev)} \tag{3.2}$$

where N is the number of all negatives positives, TN is the number of true negatives, TPR and TNR are the true positive and true negative rates (or recall and specificity), and $Prev$ is prevalence, or the ratio of positive to negative samples in the dataset. For a balanced dataset (Equation 3.3), $Prev = 1 - Prev = 0.5$, and

$$PPV = \frac{TP}{TP + FP} = \frac{TPR}{TPR + 1 - TNR} \tag{3.3}$$

Note that when missing predictions are present, those two expressions are no longer equivalent. This leads to two competing versions of generalized precision compatible with missing predictions. We will call the first version 'binary precision' (BPPV, Equation 3.4), as it ignores the missing predictions and considers only two subsets of the negative samples N – true negatives TN and false positives FP.

$$BPPV = \frac{TP}{TP + FP} \tag{3.4}$$

The second version, 'ternary precision' (TPPV, Equation 3.5), as it considers all three subsets of the set of all negative samples: true negatives TN, false positives FP and missing negatives MN, i.e. the negative samples for which no label was successfully assigned. The denominator is the sum of the number of true positives and the number of negatives that were not correctly recognized as such. This preserves the relationships between sensitivity, specificity and precision (see the proof in section A.2).

$$TPPV = \frac{TP}{TP + N - TN} = \frac{TPR}{TPR + 1 - TNR} = \frac{TP}{TP + FP + MN} \tag{3.5}$$

Note that if no missing predictions are present, $BPPV = TPPV = PPV$. Therefore, both values are justified generalizations of precision to a scenario with missing predictions. Importantly, while BPPV is similar to the more direct definition of PPV, its values may suggest overly optimistic estimates of a classifier's performance, as it is not affected by missing predictions at all. In contrast, TPPV treats missing negative predictions as if they were indeed false positives. This may make sense in the case of kNN for read pairs, where *all* 'missing' predictions are in fact pairs of contradicting predictions yielded by the classifier. Using the BPPV may obscure the fact that the classifier is intuitively *imprecise*, i.e. many of its individual positive calls are not trustworthy, and that it actually yields a (potentially low-quality) prediction for each individual read. On the other hand, for BLAST or the mappers, where the lack of matches is the main source of missing predictions, BPPV is safe to use since the missing prediction are 'truly' missing. Since BPPV may be more familiar due to its similarity to a more direct definition of precision, we will use the term 'precision' for BPPV in the context of both homology-based approaches and kNN. However, in this chapter, we will additionally report TPPV in the latter case.

3.2.5 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 modelled after the method presented by Alipanahi et al. (2015). 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 (Lundberg & Lee, 2017) of DeepLIFT (Shrikumar et al., 2017a) to extract score-weighted subsequences with the highest contribution score ('max-contrib') or all score-weighted subsequences with non-zero 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 Ns; the 'unknown' nucleotide is also called whenever there is not enough evidence to assign any other to the raw sequencing signal. Therefore, Ns 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 (R. A. Edwards et al., 2016). We also avoid weight-normalization suggested for zero-references

(Shrikumar et al., 2017a), as it implicitly models the expected GC content of all possible input sequences and assumes no Ns 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 classes.

Partial Shapley values

Building sequence logos involves calculating the 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 (Matejczyk & Michalak, 2015; Nix & Kantarciouglu, 2012), we define and use partial Shapley values to visualize contribution flow through convolutional filters. This differs from recently introduced contribution weight matrices (Avsec, Weilert, et al., 2021), 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 (Shrikumar et al., 2017a) and their reinterpretation in SHAP (Lundberg & Lee, 2017), we backpropagate the activation differences only along the paths 'passing through' y_j. In Equation 3.6, we define partial multipliers $\mu_{x_i z}^{(y_j)}$ and express them in terms of Shapley values ϕ and activation differences w.r.t. 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.

$$\mu_{x_i z}^{(y_j)} = m_{x_i y_j} m_{y_j z} = \frac{\phi_i(y_j, x)\phi_j(z, y)}{(x_i - E[x_i])(y_j - E[y_j])} \tag{3.6}$$

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 w.r.t. the reference (Equation 3.7):

$$\varphi_i^{(y_j)}(z,x) = \mu_{x_i z}^{(y_j)}(x_i - E[x_i]) = \frac{\phi_i(y_j,x)\phi_j(z,y)}{y_j - E[y_j]} \tag{3.7}$$

From the chain rule for multipliers (Shrikumar et al., 2017a), 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 3.8).

$$\phi_i(z,x) = m_{x_i z}(x_i - E[x_i]) = \sum_j \varphi_i^{(y_j)}(z,x) \tag{3.8}$$

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 (Equation 3.9-Equation 3.10). First, in this case we do not traverse any non-linearities and can directly use the linear rule (Shrikumar et al., 2017a) 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.

$$\phi_i(y_j,x) = w_i x_i = \begin{cases} w_i, & \text{if } x_i = 1 \\ 0, & \text{otherwise} \end{cases} \tag{3.9}$$

$$\varphi_i^{(y_j)}(z,x) = \frac{w_i \phi_j(z,y)}{y_j - E[y_j]} \tag{3.10}$$

The 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 coloured according to its contribution. Positive contributions are shown in red, negative contributions are blue, and near-zero contributions are grey. 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. Colour 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.

3.2.6 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, 250bp 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 125bp. We predict the infectious potential of each pseudo-read and average the obtained values at each position of the genome. Analogously, we calculate the 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 (Thorvaldsdóttir et al., 2013). For protein structures, we average the scores codon-wise to obtain contribution scores per amino acid and visualize them with PyMOL (DeLano et al., 2002).

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 non-zero filter activations. We use Gene Ontology to connect the identified genes of interest with their molecular functions and biological processes they are engaged in.

3.3 Results

3.3.1 Negative class definition

Choosing which viruses should constitute the negative class is application dependent and influences the performance of the trained models. Table A.1 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

Table 3.1: Classification performance in the fully open-view setting (all virus hosts), read pairs. Bowtie2, BWA-MEM and BLAST yield no predictions for over 35%, 19% and 10% of the samples, respectively. Ternary precision (TPPV) for the kNN is markedly lower (57.8) than the precision value presented in the table due to many conflicting predictions for individual reads in a pair. Best performance in bold.

	Accuracy	Precision	Recall	Specificity
CNN_{All} (ours)	**89.9**	93.9	**85.4**	**94.4**
$LSTM_{All}$ (ours)	86.4	89.0	83.0	89.8
kNN	57.1	86.7	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

accuracy than the 'Chordata'-trained models and the best recall overall. Rebalancing the negative class using the 'Stratified' dataset helps to 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 sections 3.2.2 and 3.3.2).

3.3.2 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 minutes using the faster model; linear speed-ups are possible if more GPUs are available. Therefore, the trained models achieve the high-throughput necessary to analyze NGS datasets. Table 3.1 presents the results of a benchmark using the 'All' test set. Low performance of the kNN classifier (Z. Zhang et al., 2019) 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% (Table A.2). 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

Table 3.2: 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. kNN (genome) and BLAST (genome) use contig-wise majority vote. kNN (contigs) and BLAST (contigs) represent performance on individual contigs treated as separate entities. kNN (reads) was not used, as high conflicting prediction rates made read-wise aggregation impracticable.

		BAcc.	Rec.	Spec.	AUPR
Genomes	CNN_{All} (ours)	**91.7**	89.3	94.2	**91.2**
	$LSTM_{All}$ (ours)	86.3	**96.2**	76.4	85.8
	BLAST (reads)	90.3	85.5	**95.1**	n/a
	kNN (genome)	82.8	93.9	71.6	n/a
	BLAST (genome)	90.5	86.3	94.6	n/a
Contigs	kNN (contigs)	83.0	94.3	71.6	n/a
	BLAST (contigs)	88.4	87.1	89.7	n/a

Table 3.3: Classification performance on the human blood virome dataset. Positive class is the majority class. BAcc. – balanced accuracy, Rec. – recall, Spec. – specificity.

	BAcc.	Rec.	Spec.	AUPR
$CNN_{All\text{-}150}$ (ours)	**96.8**	**97.3**	**96.2**	**>99.9**
$LSTM_{All}$ (ours)	91.8	88.2	95.5	**>99.9**
kNN	83.1	80.9	85.4	99.5

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 kNN and BLAST (Table 3.2). We present performance measures for both individual contigs and whole-genome predictions based on the contig-wise majority vote. We compare them to BLAST with read-wise majority vote (Deneke et al., 2017) and to read-wise average predictions of our networks, analogous to presented previously for bacteria (Bartoszewicz et al., 2020). Our method outperforms BLAST by 1.2% and kNN 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.

Table 3.4: Classification performance, novel species. Top: read pairs (see Table 3.1). BLAST yields predictions for only 64.3% of the pairs. Bottom: whole available genomes or contigs – negative class is the majority class (see Table 3.2). 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 the 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, and that the performance of $CNN_{SP\text{-}All}$ for whole genomes can be further improved by retuning of the classification threshold (see Table 5.3).

		BAcc.	Prec.	Rec.	Spec.
Read pairs	$CNN_{SP\text{-}All}$ (ours)	**74.6**	87.0	**57.9**	**91.4**
	BLAST	47.1	**94.1**	17.8	76.4
Genomes	$CNN_{SP\text{-}All}$ (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
Contigs	BLAST (contigs)	57.9	37.9	33.6	82.1

We benchmarked our models against the human blood virome dataset used by Z. Zhang et al. (2019). Our models outperform their kNN classifier. As the positive class massively outnumbers the negative class, all models achieve over 99% precision. $CNN_{All\text{-}150}$ performs best (Table 3.3). 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 (Table A.3). 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 different in the fully-open view scenario (Table 3.4). The CNN trained on the species-wise division of the 'All' dataset ($CNN_{SP\text{-}All}$) outperforms BLAST by a wide margin in the read-wise setting. Strikingly, $CNN_{SP\text{-}All}$ 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 $CNN_{SP\text{-}All}$ accuracy any further (note that this problem is addressed in chapter 5, Table 5.3). As $CNN_{SP\text{-}All}$ does not reliably outperform its Chordata-trained analogue

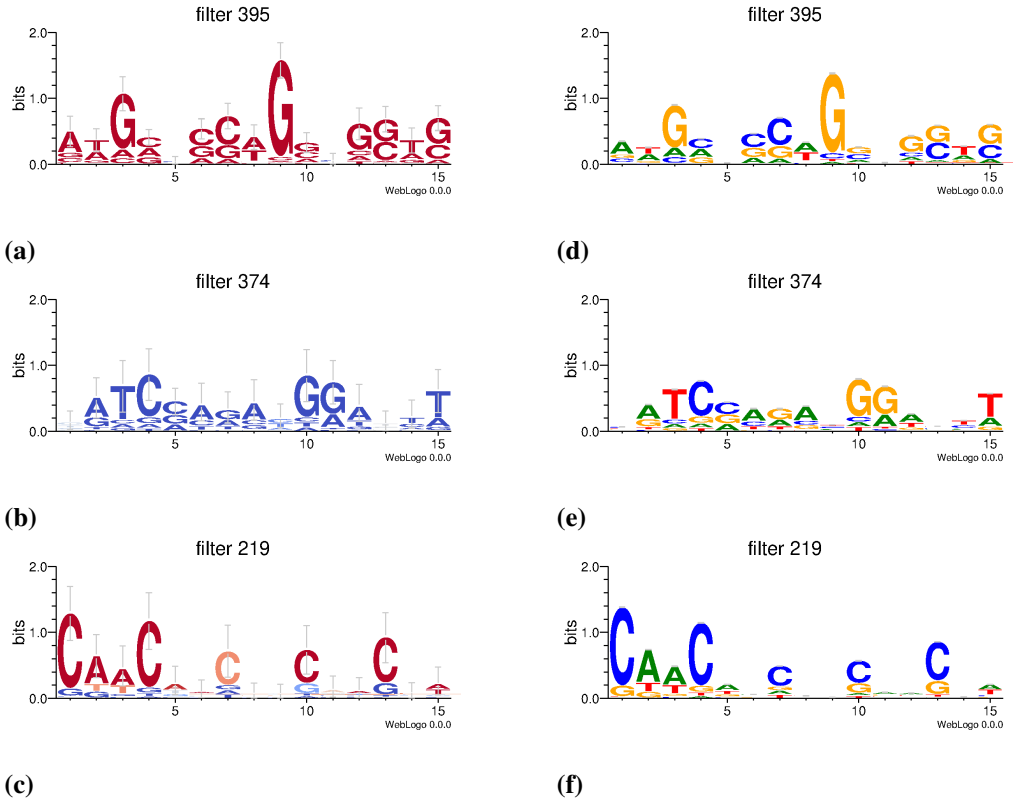


Figure 3.1: Nucleotide contribution logos of example filters. 3.1a: Second-highest mean contribution score (CNN_{All}). Error bars correspond to Bayesian 95% confidence intervals. 3.1b: Lowest mean contribution score (CNN_{All}). 3.1c: Gaps resembling a codon structure, extracted from Bartoszewicz et al. (2020). Consensus sequence: CAWCNNCNNCNNCNN. 3.1d-3.1f: Analogous logos created with the DeepBind-like 'max-activation' approach. Our 'max-contrib' logos visualize contributions of individual nucleotides, including counter-contributions.

on the 'Chordata' dataset ($CNN_{SP\text{-}Cho}$, Table 3.4), 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.

3.3.3 Filter visualization

Over 84% of all contributing first-layer filters in CNN_{All} have positive average contribution scores. We comment more on this fact in section 3.4.3. For CNN_{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 (Ns) – 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 artefacts. 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 Ns are modelled 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 non-zero 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 (Figure A.2). We speculate that this could be caused by the lower quality of the non-pathogen 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 overrepresentation. The choice of the reference input is crucial.

In the Figure 3.1 we present example filters, visualized as 'max-contrib' sequence logos based on mean partial Shapley values for each nucleotide at each position. All nucleotides of the filters with the second-highest (Figure 3.1a) and the lowest (Figure 3.1b) 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 colour, Figure 3.1c). 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 section 3.4.3. Some filters seem to learn gapped motifs resembling a codon structure (Figure 3.1c). We extracted this filter from the original DeePaC network predicting bacterial pathogenicity (Bartoszewicz et al., 2020) where the counter-contributions are common, but we find similar filters in our networks as well (Figure A.3). 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 (Y.-H. Yang et al., 2014). The filter seems to detect serine and glycine repeats in this particular gene (Figure A.4), 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 3.1a) 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 signalling ($q < 10^{-15}$).

3.3.4 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 3.2a). 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 the 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 Ns is predicted to be 'non-pathogenic' with a score of 0.

We ran a similar analysis of *S. aureus* using the built-in DeePaC models (Bartoszewicz et al., 2020) and our interpretation workflow. While a viral genome usually contains 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

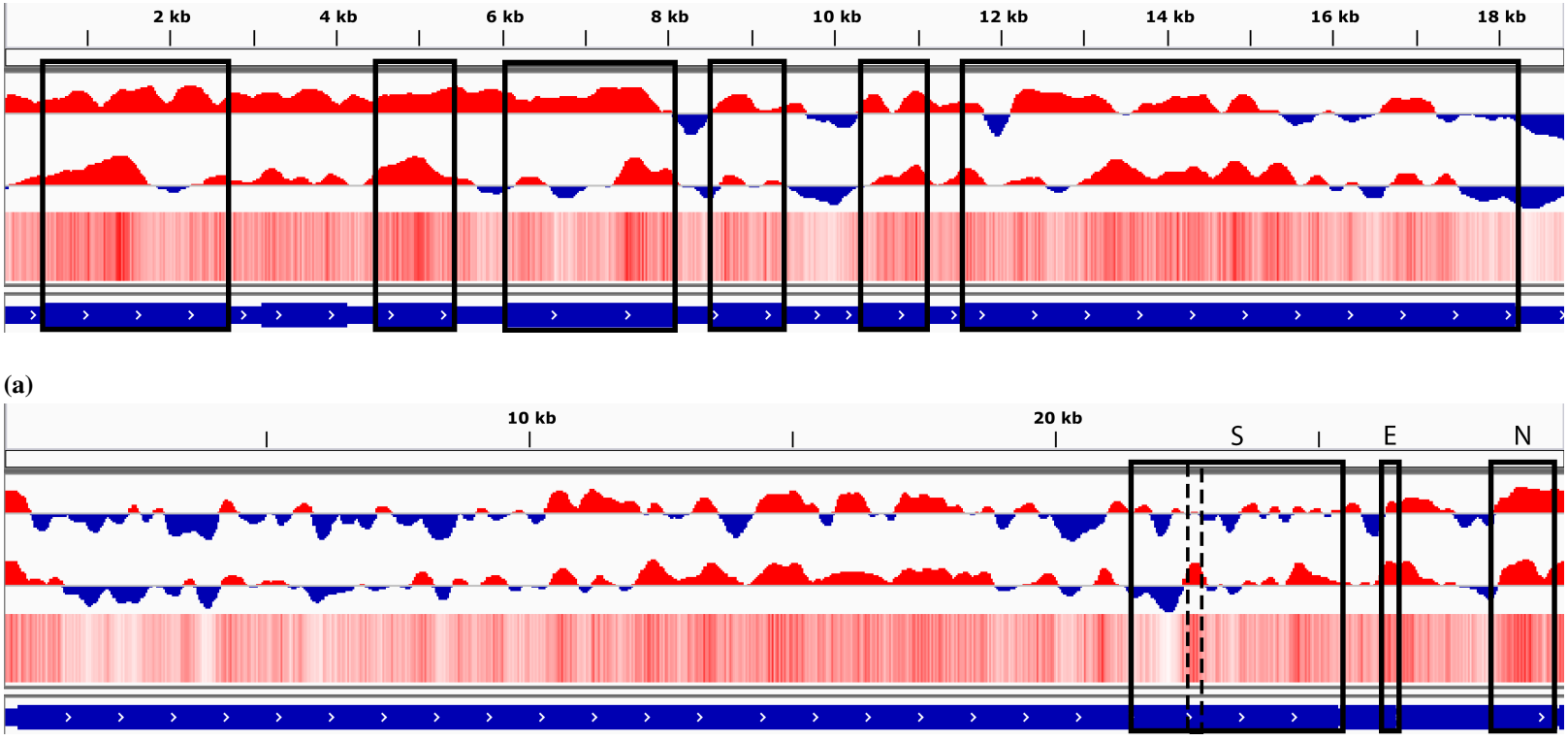
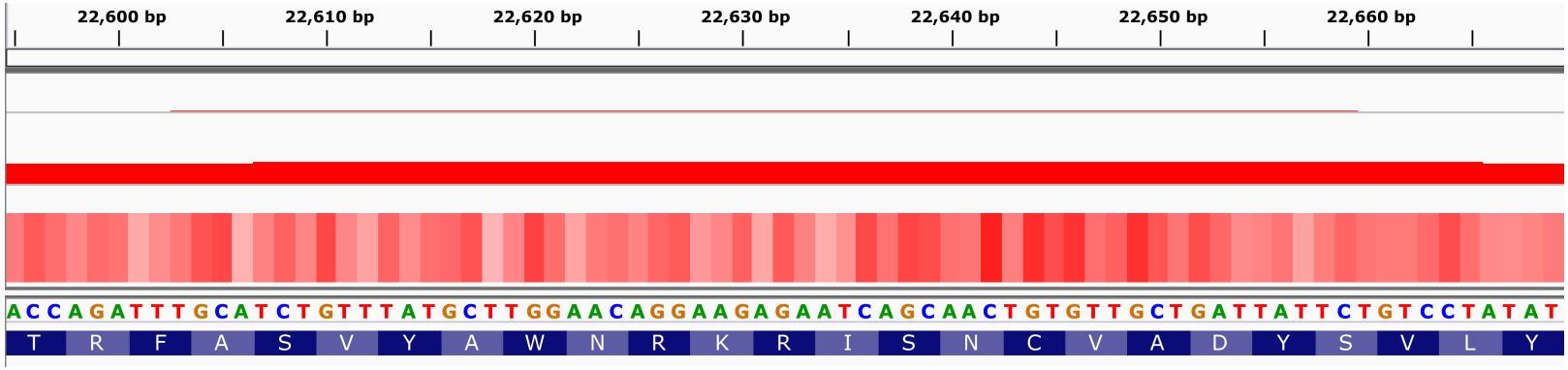
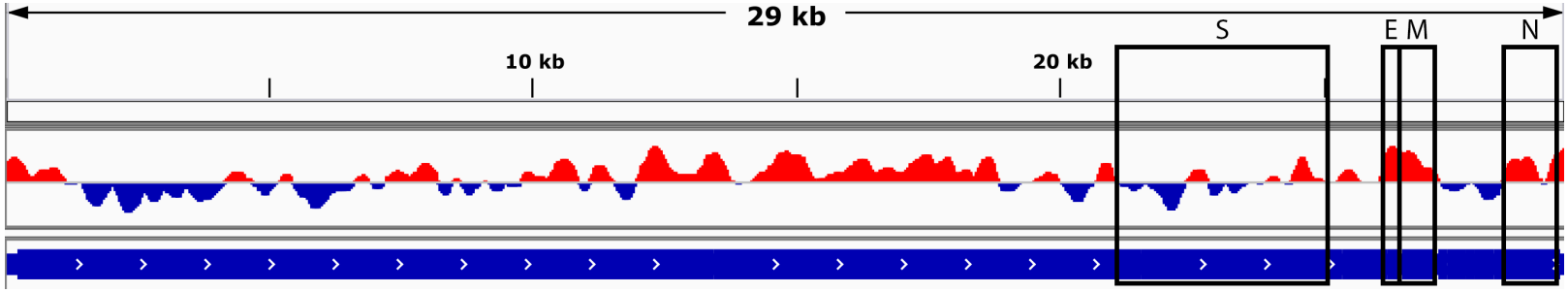


Figure 3.2: Taï Forest ebolavirus and coronavirus genomes. Top: score predicted by LSTM_{All}. Middle: score predicted by CNN_{All}. Heatmap: nucleotide contributions of CNN_{All}. Bottom, in blue: reference sequence. 3.2a: Taï Forest ebolavirus. Genes that can be detected by at least one model are highlighted in black. 3.2b: SARS-CoV-2. Whole genome and sequences encoding the spike protein (S), envelope protein (E) and nucleocapsid protein (N).



(c)



(d)

Figure 3.2: (continued) Taï Forest ebolavirus and coronavirus genomes. 3.2c: SARS-CoV-2 Spike protein gene, a small peak (positions 22,595–22,669, dashed line in Figure 3.2b) within the receptor-binding domain (predicted by CD-search, positions 22,517–23,185). Top: score predicted by $LSTM_{All}$. Middle: score predicted by CNN_{All}. Heatmap: nucleotide contributions of CNN_{All}. Bottom, in blue: reference sequence. 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. 3.2d: GWPA plot of the SARSr-CoV RaTG13 virus scored by the CNN_{All} model. The pattern of predicted infectious potentials is roughly similar to its relative, SARS-CoV-2 (Figure 3.2b). Regions of elevated infectious on the right end of the plot correspond to E, M, and N genes; the wide red region in the middle is collocated with ORF1b. Homologous regions in SARS-CoV-2 are scored highly as well.

pathogenicity (Table A.4). 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 the expression of virulence-involved genes in many pathogens (Stojkova et al., 2019). In contrast to the viral models, both negative and positive contributions are present (Figure A.5), 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 3.2b presents a GWPA plot for the whole genome of the SARS-CoV-2 coronavirus, successfully predicted to infect humans, even though the data was 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 (F. Li, 2016), 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 3.2b and Figure 3.2d), regions that the network has learned to associate with the infectious phenotype are distributed non-uniformly 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 misclassification of RaTG13 could be indicative of a general zoonotic potential of SARS-related coronaviruses. In the Figure 3.2b, 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 3.2c 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 (Marchler-Bauer et al., 2017) using the default parameters. The maximum score of this peak is noticeably higher for SARS-CoV-2 (0.87) than for its analogue in RaTG13 (0.67). Figure 3.3 presents the RBD in the structural context of the whole S protein (PDB ID: 6VSB, (Wrapp et al., 2020)), as well as in complex with a SARS-neutralizing antibody CR3022 (PDB ID: 6W41, (Yuan et al., 2020)). The high score peak roughly corresponds to one of the regions associated with reduced expression of the RBD (Starr et al., 2020), located in the core-RBD subdomain.

It covers over 71% of the CR3022 epitope, as well as the neighbouring site of the N343 glycan. The latter is present in the epitope of another core-RBD targeting antibody, S309 (Pinto et al., 2020). All the per-residue average contributions in the region are positive (Figure A.6), even in the regions of lower pathogenicity score, in accordance with the results presented in Figure 3.2c.

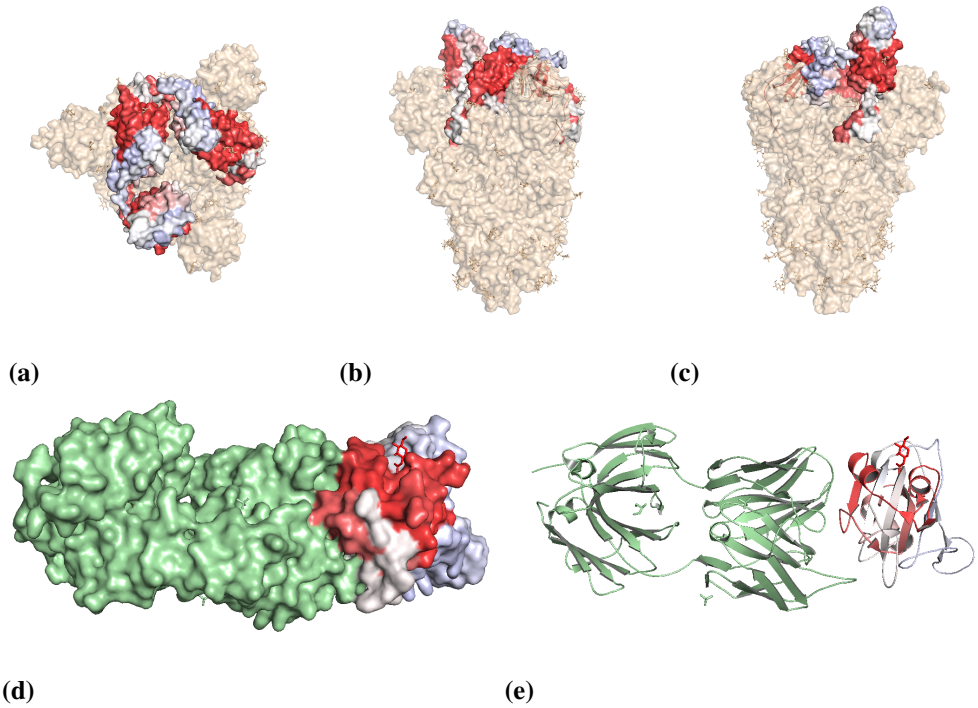


Figure 3.3: Predicted infectious potentials plotted over the SARS-CoV-2 spike glycoprotein receptor-binding domain. 3.3a-3.3c: Top and side view of the spike protein. Three receptor-binding domains (RBDs) are colored in blue, white and red according to the predicted infectious potential of the corresponding genomic sequence. One of the domains is in the 'up' conformation. Red regions corresponding to the peak in Figure 3.2c are located in the core-RBD subdomain. 3.3d: 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 neighbouring 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. 3.3e: Cartoon representation of Figure 3.3d. The red region is centered on two exposed α-helices surrounding the core β-sheet (lower score, white).

3.4 Discussion

3.4.1 Accurate predictions from short DNA reads

Compared to the previous state-of-the-art in viral host prediction directly from NGS reads (Z. Zhang et al., 2019), our models drastically reduce the error rates. This holds also for novel viruses not present in the training set. The 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 to 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 modelling step is also important, as shown by our comparison to the baseline kNN 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 (Bartoszewicz et al., 2020).

In the paired read scenario, the previously described kNN approach fails, and standard, alignment-based homology testing algorithms cannot find any matches in more than 10% of the cases, resulting in relatively low accuracy. On a real human virome sample, where the main source of the negative class reads is most likely contamination (Moustafa et al., 2017), our method filters out non-human 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 (Brendel & Bethge, 2019). 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 (Bartoszewicz et al., 2020), 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 assess the risk of a host-switching event. The interpretability suite presented here aims at shedding some light on this question, but more research is needed.

3.4.2 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 in 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 (Diggans & Leproust, 2019; National Academies of Sciences, Engineering, and Medicine, 2018; National Research Council, 2010), machine learning methods are a promising solution. This has been suggested before for the bacterial DeePaC models (Bartoszewicz et al., 2020), 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 (2020) suggested that while the intended purpose of pathogenic 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 artefacts highlight the fact that only some 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 (Alley et al., 2019; Biswas et al., 2021; Brookes et al., 2019) and regulatory sequence design (Gupta & Kundaje, 2019; Gupta & Zou, 2019; Linder et al., 2019; Schreiber et al., 2020) 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.

3.4.3 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 (Alipanahi et al., 2015) 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 (Lundberg & Lee, 2017) – does not hold in full. Indeed, k-mer distribution profiles are frequently used features for modelling DNA sequences (as shown also by the dimer-shuffling method of generating reference sequences proposed by Shrikumar et al. (2017a)). 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 into 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 non-trivial, and a promising avenue for further research.

3.4.4 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 behaviour 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.

3.4.5 Conclusions

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' behaviour and analyze a recently emerged SARS-CoV-2 virus. The approach presented here is implemented as a python package and a command-line tool easily installable with Bioconda (Grüning et al., 2018) (`conda install deepacvir`, requires setting up Bioconda), Docker (`docker pull dacshpi/deepac`) or pip (`pip install deepacvir`). Detailed installation instructions, a 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. The datasets of simulated reads with associated metadata are hosted at https://doi.org/10.5281/zenodo.4312525.

4 Detecting novel pathogens in real time with DeePaC-Live

Summary

Novel pathogens evolve quickly and may emerge rapidly, causing dangerous outbreaks or even global pandemics. Next-generation sequencing is the state-of-the-art in open-view pathogen detection, and one of the few methods available at the earliest stages of an epidemic, even when the biological threat is unknown. Analyzing the samples as the sequencer is running can greatly reduce the turnaround time, but existing tools rely on close matches to lists of known pathogens and perform poorly on novel species. Machine learning approaches can predict if single reads originate from more distant, unknown pathogens, but require relatively long input sequences and processed data from a finished sequencing run. Incomplete sequences contain less information, leading to a trade-off between sequencing time and detection accuracy. Using a workflow for real-time pathogenic potential prediction, we investigate which subsequences already allow accurate inference. We train deep neural networks to classify Illumina and Nanopore reads and integrate the models with HiLive2, a real-time Illumina mapper. This approach outperforms alternatives based on machine learning and sequence alignment on simulated and real data, including SARS-CoV-2 sequencing runs. After just 50 Illumina cycles, we observe an 80-fold sensitivity increase compared to real-time mapping. The first 250bp of Nanopore reads, corresponding to 0.5 s of sequencing time, are enough to yield predictions more accurate than mapping the finished long reads. The approach could also be used for screening synthetic sequences against biosecurity threats.

This chapter is based on Bartoszewicz, Genske, et al. (2021a), which is a joint work with Ulrich Genske and Bernhard Y. Renard. A detailed description of the authors' contributions can be found in section Thesis outline.

4.1 Background

4.1.1 Motivation

The SARS-CoV-2 coronavirus emerged in late 2019, causing an outbreak of COVID-19, a severe respiratory disease, which quickly developed into a global pandemic of 2020. This virus of probable zoonotic origin (P. Zhou et al., 2020) is a terrifying example of how easily new agents can spread. What is more, many more novel pathogens are expected to emerge. They evolve extremely quickly due to high mutation rates or horizontal gene transfer, while human exposure to the vast majority of unexplored microbial biodiversity is rapidly growing (Trappe et al., 2016; Vouga & Greub, 2016). New biosafety threats may come as novel bacterial agents like the Shiga-toxigenic *Escherichia coli* strain that caused a deadly epidemic in 2011 (Frank et al., 2011). Outbreaks of previously unknown viruses can be even more severe. This is not limited to coronaviruses like SARS-CoV-1, MERS and SARS-CoV-2. Novel strains of the Influenza A virus caused four pandemics in less than a hundred years (from the 'Spanish flu' of 1918 until the 'swine flu' of 2009), killing millions of people. Even known pathogens may be difficult to control, as proven by the outbreaks of Zika and Ebola in the 2010s (Calvignac-Spencer et al., 2014). Importantly, many viruses can switch between more than one host or evolve silently in an animal reservoir before infecting humans. This happened before to HIV, Ebola, many dangerous strains of the Influenza A virus and the coronaviruses mentioned above.

If an outbreak involves a new, unknown pathogen, targeted diagnostic panels are not available at first. Open-view approaches must be used and next-generation sequencing (NGS) is the method of choice (Calistri & Palù, 2015; Lecuit & Eloit, 2014). A swift response is crucial, and analyzing the samples during the sequencing run, as the reads are produced, greatly improves turnaround times. This can be achieved by design using long-read sequencing like ONT. However, lower throughput and high error rates of those technologies impede their adoption for pathogen detection. Scalability, cost-efficiency and accuracy of Illumina sequencing still make it a gold standard, although it may change in the future with the establishment of improved ONT protocols and computational methods (Loka et al., 2019).

4.1.2 Real-time analysis of Illumina sequencing data

Analyzing Illumina reads during the sequencing run poses unique technical and algorithmic challenges. The DRAGEN system relies on field-programmable gate arrays (FPGAs) to speed up the computation and can be combined with specialized protocols to detect clinically relevant variants in the human genome but depends on finished reads (Miller

et al., 2015). An alternative approach is to use the general-purpose computational infrastructure and optimize the algorithms for fast and accurate analysis of incomplete reads as they are produced, during the sequencing run. HiLive (Lindner et al., 2017) and HiLive2 (Loka et al., 2019) are real-time mappers, performing on par with the traditional mappers like Bowtie2 (Langmead & Salzberg, 2012) and BWA (H. Li & Durbin, 2010) with no live-analysis capabilities. However, as read mappers are designed for fast and precise sequence alignment, they are expected to miss most of the reads originating from genomes highly divergent from the available references. Therefore, even though existing live-analysis tools and associated pipelines do cover standard read-based pathogen detection workflows, their performance on novel agents is limited by their dependence on databases of known species. The same problem applies to sequence alignment and taxonomic classification in general, also outside of the real-time analysis context (National Research Council, 2010). In this work, we show that using deep learning to predict if a read originates from a human pathogen is a promising alternative to mapping the reads to known references if the correct reference genome is not yet known or unavailable. We also investigate the trade-offs between sequencing time and classification performance.

4.1.3 Read-based detection of novel pathogens

Deneke et al. (2017) have shown that methods like read-mapping (Langmead & Salzberg, 2012) (with optional additional filtering steps of PathoScope2 (Hong et al., 2014)), BLAST (Altschul et al., 1990; Camacho et al., 2009) or Kraken (Wood & Salzberg, 2014), which all try to assign target sequences to their closest taxonomic matches, fail to yield any predictions for a significant fraction of reads originating from novel pathogens. BLAST was the best of those approaches, missing the least reads and achieving the highest accuracy. More complex detection workflows like PathoScope2 (Hong et al., 2014), Sigma (Ahn et al., 2015) or KrakenUniq (Breitwieser et al., 2018) depend on assigning individual reads to taxa by mapping or *k*-mer matching, so necessarily suffer from the same problems. In contrast, taxonomy-agnostic methods try to reduce their database dependency by assigning putative phenotypes directly to analyzed sequences, deliberately omitting the taxonomic classification step. For example, a naïve Bayes classifier based on *k*-mer frequency features can be trained to classify reads directly into arbitrary classes (G. L. Rosen et al., 2011). However, in the context of detecting novel bacterial pathogens, a random forest approach of PaPrBaG (Deneke et al., 2017) performs much better. Z. Zhang et al. (2019) used a kNN classifier to develop a similar method for detection of human-infecting viruses. An analogous deep learning approach,

DeePaC, outperforms the traditional machine learning algorithms on both novel bacteria (Bartoszewicz et al., 2020) and viruses (DeePaC-vir (Bartoszewicz, Seidel, & Renard, 2021a)), offering an additional level of interpretability on nucleotide, read and genome levels. A similar method (Mock et al., 2020) focuses on detailed predictions for a small set of three viral species and cannot be used in an open-view setting. Preliminary work by Q. Guo et al. (2020) supports it, but the code, models or installables are not available yet at the time of writing, so the method cannot be reused. What is more, the 'novelty' of the viruses in the corresponding test set is difficult to assess – it can contain genomes of viruses present in the training set, as long as they were resequenced after 2018. While pathogenicity prediction methods using contigs, whole genomes or protein sets as input also exist, this work focuses on read-based classification to offer real-time predictions and avoid delays necessitated by assembly pipelines. However, read-based methods have been shown to perform well also for full genomes and assembled contigs, achieving similar or better performance than alignment-based approaches (Bartoszewicz, Seidel, & Renard, 2021a; Bartoszewicz et al., 2020; Deneke et al., 2017). This mirrors successful adoption of CNNs using raw DNA sequences as inputs in other fields ranging from regulatory genomics (Alipanahi et al., 2015; Eraslan et al., 2019; J. Zhou & Troyanskaya, 2015) to viral bioinformatics (Q. Guo et al., 2020; Mock et al., 2020; Ren et al., 2020) and biosecurity (Nielsen & Voigt, 2018).

4.2 Methods

4.2.1 Data preparation

In this chapter, we will use the term *subread* in a special sense: the first k nucleotides of a given sequencing read (in other words, a *prefix* of a read). The original DeePaC and DeePaC-vir datasets (Bartoszewicz, Seidel, & Renard, 2021a; Bartoszewicz et al., 2020) consist of 250bp simulated Illumina reads in fastq format. The training, validation, and test sets contain reads originating from different species of pathogenic (including opportunistic pathogens) or commensal bacteria labelled using the IMG database (I.-M. A. Chen et al., 2019). The DeePaC-vir dataset is built in an analogous way using different viruses mined from the Virus-Host Database (Mihara et al., 2016). Three alternative versions of the viral dataset are available, differing in the negative class definition. We used the fully open-view 'All' dataset, containing all viruses available in VHDB. In all cases, the training set contained 20 million single reads, the validation set contained 2.5 million single reads and the test set – 2.5 million paired-end reads. This setup allows training models correctly handling single, isolated reads, but also

testing their performance on read pairs. All sets were balanced with regard to the class distribution and contained a mixture of reads originating from multiple different species. Most importantly, the training, validation and held-out test sets contain different viruses or bacterial species, so that generalization to 'novel' agents (i.e. unseen in training) can be explicitly evaluated. For more details regarding the dataset generation, we refer the reader to the corresponding publications (Bartoszewicz, Seidel, & Renard, 2021a; Bartoszewicz et al., 2020).

We used the test datasets to generate corresponding subread datasets with subread lengths between 25 and 250 (full read) with a step of 25. Every subread of every subread set had a corresponding subread in all the other sets. Therefore, we explicitly model new information incoming during a sequencing run, as each subread length k corresponds to the kth cycle. To generalize over a large spectrum of possible subread lengths, we built mixed-length training and validation sets by randomly choosing a different k for every read in a set. In this setup, all integer values of k between 25 and 250 were allowed. As Bartoszewicz, Seidel, and Renard (2021a) have previously presented both 250bp and 150bp-trained reverse-complement CNN classifier for viruses, we also generated an analogous 150bp subread bacterial dataset, and used it to train a corresponding CNN.

To test the performance of the bacterial models on real sequencing data, we analyze reads coming from a real sequencing run of a pathogenic bacterium *Staphylococcus aureus*. This species was not present in the training set (it had been randomly placed in the validation set), so it models a 'novel' pathogen without a known reference genome. The same species was used previously by Bartoszewicz et al. (2020) to assess the original version of DeePaC; here, we focus on analyzing the sequences as they are generated by the sequencer as opposed to predicting for full reads after the sequencing run is finished. To this end, we downloaded an SRA archive of 251bp-long paired-end reads (accession number SRR5110368) sequenced with an Illumina MiSeq device (Manara et al., 2018). To evaluate the viral models, we downloaded an archive of 151bp-long paired-end SARS-CoV-2 reads originating from a COVID-19 positive human from San Diego county (SRR11314339). We use untrimmed reads with the quality information to generate Illumina base call file (BCL) files as they would be internally generated by the sequencer. We then run HiLive2 on the BCL data to map the reads to the training reference database; HiLive2 output is then parsed and passed to the models. However, we ignore the last cycle of each mate when generating HiLive2 output and subsequent analyses, as the bad quality of this last nucleotide makes it generally unreliable. We select the predictor that achieves the highest average accuracy on the DeePaC or DeePaC-vir dataset and compare it to the standard, mapping-based real-time analysis with HiLive2 alone.

4.2.2 ResNets and hybrid classifiers

We investigated two architectures shown previously to perform well in the pathogenicity or host range prediction task – a reverse-complement CNN consisting of 2 convolutional layers and 2 fully-connected layers and a reverse-complement bidirectional LSTM. For more design details and the description of the reverse-complement variants of convolutional and LSTM layers, we refer the reader to Bartoszewicz, Seidel, and Renard (2021a) and Bartoszewicz et al. (2020). Those architectures guarantee identical predictions for sequences in their forward and reverse-complement orientations in a single forward pass. Previous work (Bartoszewicz, Seidel, & Renard, 2021a; Bartoszewicz et al., 2020) has shown that they were more accurate than alternative machine learning and homology-based approaches for the read-based pathogenic or infectious potential prediction task. However, as short subread sequences convey less information, we expected the subread classification problem to be more challenging than in the case of relatively long 250bp reads. We suspected that deeper, more expressive networks could perform better. Therefore, we implemented a new architecture – a reverse-complement ResNet extending the previous work with skip connections (He et al., 2016) while satisfying the reverse-complementarity constraint.

We considered 18- and 34-layer ResNet variants where all convolutional layers of a standard ResNet (including size-1 convolutions in skip connections) are replaced with reverse-complement convolutions (Bartoszewicz et al., 2020). We trained them for a maximum of 30 epochs, using early stopping with a patience of 10 epochs (see section A.3, Table A.6, Figure A.7 for architecture details). For all models, we used input dropout, which may be understood as switching a random fraction of the input nucleotides to Ns. As generating subreads already discards some sequence information, we retuned the input dropout rate for the bacterial models, testing the values of 0.2 and 0.25. For the viral models, it was already shown that the dropout rate of 0.25 works better even in the case of 150bp subreads; we therefore only considered the higher value. We compared the CNN, LSTM and ResNet models trained on the mixed-length datasets; in addition to that we also considered the bacterial CNN trained on 150bp subreads analogous to the viral $\text{CNN}_{\text{All-150}}$ from Bartoszewicz, Seidel, and Renard (2021a). The ResNet-18 trained with an input dropout rate of 0.25 achieved the highest accuracy on the bacterial mixed-length validation set and was selected for further evaluation. For viruses, the ResNet models were the best as well – although the ResNet-34 was the most accurate in absolute terms, the error rate improvement over the 18-layer variant was negligible (<0.5%) while the computational cost (measured in wall-clock time of both training and inference) was roughly twice as high. Since inference speed is crucial for

the application presented here, we decided to select the equally accurate but faster and more efficient ResNet-18. For Nanopore data, we retrained only the CNNs, LSTMs and ResNets-18, omitting the ResNet-34 architectures.

Finally, we create hybrid classifiers, which first extract reads mappable to known references with HiLive2 and then predict the phenotype for the remaining unmapped reads using the ResNet (see Figure 4.1). This enables identification of the closest relatives of the analyzed pathogen, while still predicting labels for reads missed by the mapper. The reads associated with the pathogenic or infectious phenotype may then be extracted and used in downstream analysis.

To capture Illumina reads as they are generated by a sequencer, we use HiLive2's BCL file conversion and real-time mapping capabilities (Lindner et al., 2017; Loka et al., 2019). Our workflow, called DeePaC-Live, consists of three asynchronously callable modules. The sender module watches the HiLive2 output directory, detecting BAM files with both mapped and unmapped reads. By default, it selects only the unmapped reads for further analysis, but this can be adjusted by the user to focus on either mapped reads, or all sequenced reads. The output of the sender module may be automatically sent over to a remote server (e.g. a GPU-equipped machine) using the Secure File Transfer Protocol (SFTP). Data privacy issues should be kept in mind.

The receiver module may operate on the remote or local machine depending on the available infrastructure. It captures the sender's output and uses a selected deep neural network to predict pathogenic potentials (standard sigmoid output scores between 0 and 1) for all the selected reads. Then, it filters them according to a predefined decision threshold (typically 0.5), outputting separate files for reads associated with a pathogenic and non-pathogenic phenotype. Finally, the optional refiltering module allows reanalyzing the prediction with an alternative threshold (e.g. to select only the highest-confidence predictions) and averaging the outputs of multiple receiver modules to create a simple ensemble classifier.

Note that DeePaC-Live supports easy substitution of the underlying neural network of any custom Keras model, allowing future improvements in the architecture details and real-time predictions for tasks other than pathogenicity prediction. We also support seamless integration with the built-in DeePaC (Bartoszewicz et al., 2020) and DeePaC-vir (Bartoszewicz, Seidel, & Renard, 2021a) models, although we recommend using the newer, updated models, especially in the real-time analysis scenario. As the prediction functions of DeePaC were not optimized for fast inference, we added a possibility to adjust the inference batch size to fully utilize the computing power of a given GPU. We set the batch size to 1536, being the highest multiple of 512 that would not cause out-of-memory errors for any of the tested models. While the batch size could be further

increased for the CNN and ResNet-based networks used in this study, this did not speed up inference any more.

4.2.3 Benchmarking

Bacteria

We compare the ResNets and hybrid classifiers to the original DeePaC models, as well as an alternative random forest approach, PaPrBaG (Deneke et al., 2017). We trained a DNA-only PaPrBaG forest (Bartoszewicz et al., 2020) on the mixed-length bacterial dataset. For both machine learning approaches, we average the predictions for both mates of a read pair for a boost in accuracy (Bartoszewicz et al., 2020).

In addition to that, we evaluate two alignment-based methods – HiLive2 in the 'very-accurate' mode (Lindner et al., 2017; Loka et al., 2019) and dc-megablast (Camacho et al., 2009) with an E-value cutoff of 10 and the default parameters. A successful match to a pathogen reference genome is treated as a positive prediction; a match to a non-pathogen is a negative. In case of multiple matches, the top hit is selected. We build the HiLive2 FM-index and the BLAST database using all the genomes used for training read set generation. If BLAST aligns two mates of a read pair to genomes with conflicting labels (i.e. one pathogen and one non-pathogen), we treat them both as missing predictions. For HiLive2, we treat them separately, as the high precision of HiLive2 warrants considering all the obtained matches as relevant. If only one mate has a match, we propagate the match to the other mate. We calculate the performance measures taking all the reads in the sample into account. Hence, missing predictions affect both true positive and true negative rates.

Viruses

We use an analogous approach to evaluate the classification performance on reads originating from novel viruses. However, since PaPrBaG is a method developed for bacterial genomes, we benchmark the models against a k-nearest neighbours (kNN) virus host classifier (Z. Zhang et al., 2019). We train the kNN as described by the authors, using non-overlapping 500bp long 'contigs' generated from the source genomes. Training based on simulated reads was not possible due to high computational cost, but Z. Zhang et al. (2019) showed that a model trained this way can be used to predict pathogenic potentials of short NGS reads. As kNN yields binary predictions, we integrate them using the same approach we use for BLAST. Finally, we compare the models to the original DeePaC-vir models.

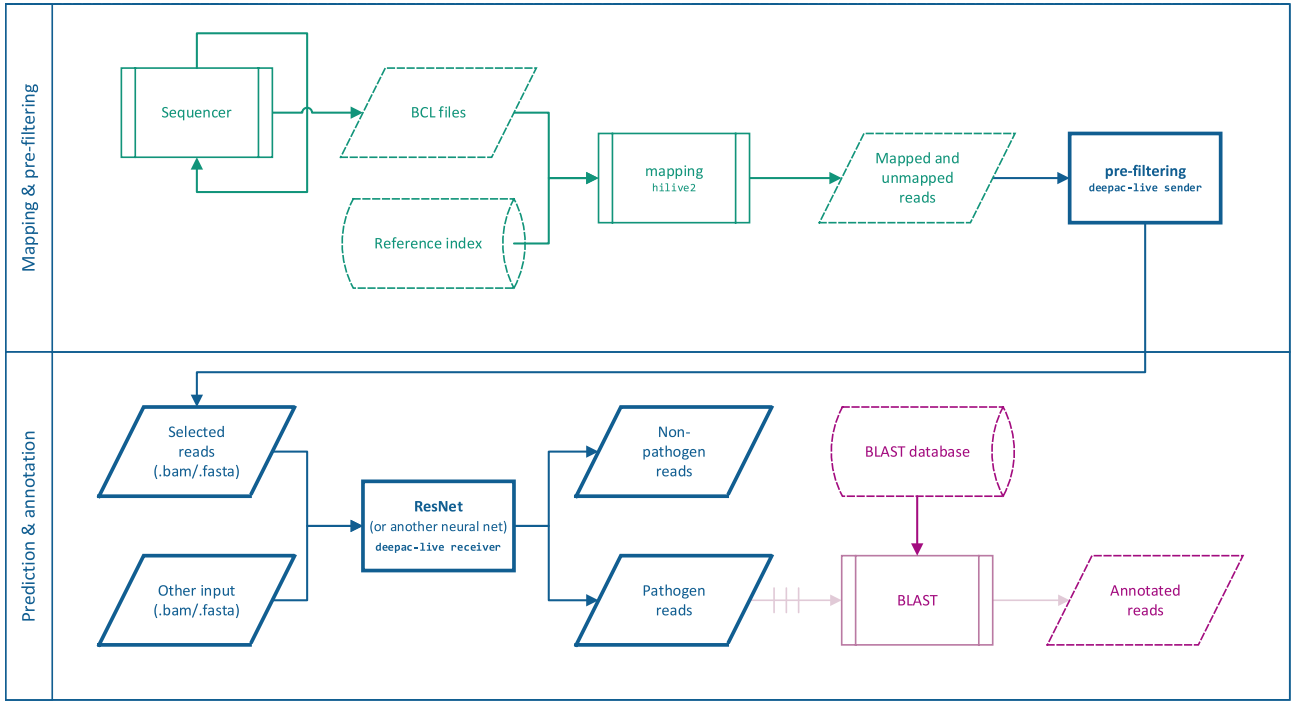


Figure 4.1: DeePaC-Live workflow, including real-time mapping with HiLive2, classification with ResNets, and optional downstream analysis with BLAST. Mapping and prefiltering can be performed on the same machine as prediction and annotation, but the prefiltered reads can also be sent to a GPU-equipped remote server. Green: An Illumina sequencer generates the binary BCL files. They are captured by HiLive2, which maps the reads to an index of known references. Blue: The core of the workflow, implemented in the DeePaC-Live package. HiLive2 output is captured by the sender module, which selects a subset of reads for further analysis. Selecting the unmapped reads (default) corresponds to the hybrid classifier (HiLive2+ResNet); selecting both mapped and unmapped reads means that all reads will be classified with the ResNet. Alternatively, other inputs in fasta or bam format can be used instead (e.g. basecalled Nanopore reads). The receiver module performs classification of incoming reads with the appropriate ResNet (which can be substituted for another classifier if necessary) and outputs files filtered by the predicted class. Purple: optional follow-up analysis with BLAST, as performed for the Illumina SARS-CoV-2 sequencing run. Putative pathogen reads can be annotated by manually passing them to BLAST using an appropriate database.

Real Illumina runs

We evaluate the workflow on real data from *Staphylococcus aureus* (SRA accession: SRR5110368, Manara et al. (2018)) and SARS-CoV-2 novel coronavirus (SRR11314339) sequencing runs. The virus was not present in the training database, as it had not yet been discovered when the DeePaC-vir datasets were compiled. *S. aureus* was also absent from the corresponding training set and was previously used to evaluate DeePaC (Bartoszewicz et al., 2020). To showcase how the approach can be used for rapid detection of novel biological threats, we test the performance of the classifiers after just 50 sequencing cycles. As the predictions of the deep learning approaches do not offer any information about the closest known relative of a novel pathogen, we extend the workflow using BLAST on reads prefiltered by the models. This enables a drastic increase in the pathogen read identification rate while also providing insight into their biological meaning. Using BLAST on full NGS datasets is usually not feasible because of the computational cost. What is more, it has been previously shown (Bartoszewicz, Seidel, & Renard, 2021a; Bartoszewicz et al., 2020; Deneke et al., 2017) that machine learning approaches perform better in pathogenic potential prediction tasks. Therefore, we see the combination of a filtering step with a BLAST follow-up as an in-depth analysis of the subreads of interest while discarding the potentially non-informative ones.

Nanopore

Finally, we predict infectious potentials from more noisy subreads of Nanopore long reads. To this end, we resimulated the bacterial and viral datasets using the exact same genomes and the context-independent model of DeepSimulator 1.5 (Y. Li et al., 2020). We set the target average read length to 8kb and discarded reads shorter than 250bp. Then, we extracted 250bp-long subreads for training and evaluation of the classifiers, but kept full reads for benchmarking against minimap2 (H. Li, 2018), a popular Nanopore mapper. We chose 250bp as this allows fair comparison with other models and corresponds to information available after ca. 0.5 s (Rang et al., 2018). Successful predictions after such a short time could be used together with real-time selective sequencing (Loose et al., 2016) to enrich the samples in reads originating from pathogens and save resources. We trained new models for the bacterial and viral Nanopore datasets and compared them with minimap2 and models trained on 250bp Illumina reads. Evaluation of minimap2 was performed analogously to HiLive2's, selecting the representative alignment if a chimeric match was found. In addition to the simulated data prepared as explained above,

we also used two real SRA datasets: a SARS-CoV-2 isolate (SRR11140745, collected on 14 Feb 2020) and a clinical *S. aureus* sample (SRR8776887, Dilthey et al. (2020)).

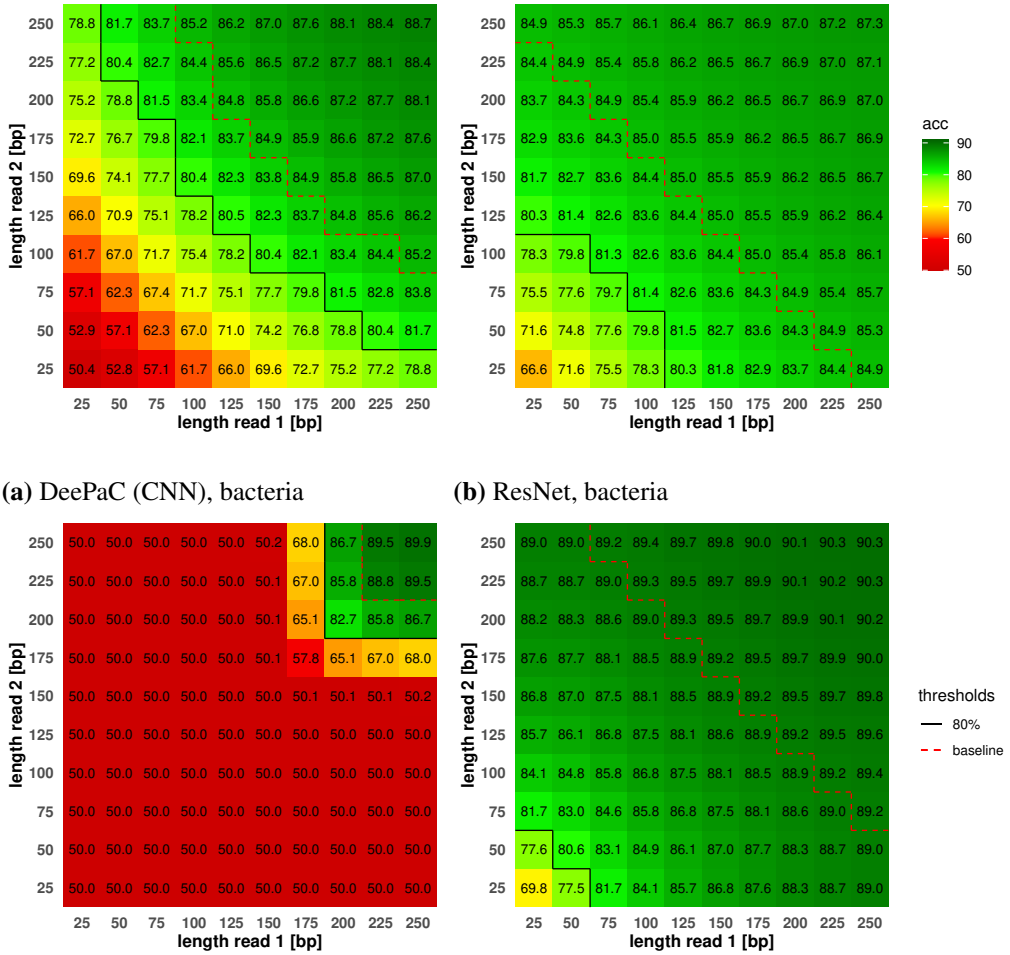
4.3 Results

4.3.1 Subread models

In Illumina paired-end protocols, the barcodes are sequenced after the first mate, making live-demultiplexing possible but problematic. Changing the barcode sequencing order is not trivial, as initial clustering requires sufficient sequence diversity in the first several cycles. A possible workaround uses asynchronous paired-end sequencing protocols (Loka et al., 2019), sacrificing the first read's length for faster demultiplexing and relying on the second mate to compensate for the lost information. We tested the models in 100 settings corresponding to different lengths of the first mate (modelling different length-time trade-offs) and the second mate (modelling incoming information after demultiplexing). As shown in the Figure 4.2, the previous state-of-the-art for the bacterial dataset (Bartoszewicz et al., 2020) is outperformed by the ResNet trained on mixed-length subreads across most of the spectrum of read length combinations. For the longest read pairs, where the sum of read lengths is 400bp or more, accuracy is slightly lower. For the read pairs with a total length below 375bp, the ResNet performs better. This could be related to the old model being explicitly optimized for 250bp-long sequences. Performance of the DeePaC-vir's 250bp-trained CNN (Bartoszewicz, Seidel, & Renard, 2021a) collapses for viral reads shorter than 200bp, while the ResNet trained on the mixed-length reads maintains accuracy higher than 80% for reads as short as 50bp. What is more, the ResNet slightly outperforms the previous state-of-the-art also on full-length reads, with accuracy over 90% for pairs of 225bp or more.

4.3.2 Hybrid models

Table 4.1 and Figure 4.3 present classification performance over the whole sequencing run (all cycles for both mates) for the bacterial dataset. The highest accuracy is achieved by the ResNet-based hybrid classifier. High recall of DeePaC (CNN) is actually an artefact – its predictions for shorter subreads are extremely imprecise (precision for 25bp is 50.6%), suggesting that the network simply classifies an overwhelming majority of short subreads as positive regardless of their actual sequence. This effect does not occur for the hybrid classifier, suggesting that although it achieves the second-highest true positive rate overall, it is likely the most sensitive method useful in practice.



(a) DeePaC (CNN), bacteria

(b) ResNet, bacteria

(c) DeePaC (CNN), viruses

(d) ResNet, viruses

Figure 4.2: Accuracy heatmaps for different lengths of reads in a read pair. Protocols with asynchronous read lengths (a shorter first mate) allow earlier demultiplexing during the sequencing run. Varying the length of the second mate models obtaining new information during sequencing. The black line corresponds to an arbitrarily set 80% accuracy threshold. The baseline (dashed red line) is the accuracy of each model on 250bp single reads. Crossing the baseline means that the additional sequence information of the second mate contains useful signal, rather than just noise hurting the final predictions. DeePaC (CNN) models were trained on 250bp reads and ResNet on 25–250bp subreads. The accuracy matrices are almost symmetric (with negligible deviations), proving that the models' performance is on average identical for the first and the second mate.

Table 4.1: Average performance on reads from novel bacterial species across the whole sequencing run. The hybrid classifier combining HiLive2 and ResNet achieves the highest accuracy. Recall of DeePaC (CNN) is inflated by its unreliable predictions for short subreads. HiLive2 is the most precise method.

	Accuracy	Precision	Recall
HiLive2+ResNet	**83.6**	80.0	89.8
ResNet	82.1	79.2	86.7
DeePaC (CNN)	79.3	75.6	**91.4**
DeePaC (LSTM)	79.7	75.2	87.0
PaPrBaG	74.5	72.7	78.1
BLAST	60.8	84.8	76.1
HiLive2	22.2	**97.3**	36.3

For the viral dataset, the ResNet performs slightly better than HiLive2 even on the reads that HiLive2 is able to map. If only the mapped reads are considered, the ResNet correctly labels 90.7% of them, compared to 89.8% for HiLive2 itself. The effect is especially strong for reads 50bp and longer, where the average accuracy of the ResNet rises to 91.8%, while HiLive2's stays the same. It seems to stem mainly from increased sensitivity of the deep learning approach (89.3% for reads over 50bp, compared to 81.8% for HiLive) at some cost in precision (95.3% and 99.2%, respectively).

This is most probably why the ResNet has higher accuracy than the hybrid classifier also when unmapped reads are considered, as presented in Table 4.2 and Figure 4.4. It is also the most sensitive prediction method overall. The hybrid classifiers offer a good trade-off between accuracy and precision. As in other real-time analysis tasks (Loka et al., 2019; Tausch, Strauch, et al., 2018), the latter is high even for very short subreads (Figure A.8).

4.3.3 Runtime

To estimate the sample size that can be analyzed in real-time after parsing or mapping with HiLive2, we measured how many reads per second could be processed by the pathogenicity prediction methods compared in this study. Then, we calculated the number of predictions feasible in a time-frame corresponding to 25 cycles (with wall-time per cycle as in Loka et al. (2019)). In the Table A.7 we present how many reads can be analyzed with no delays if the output is produced every 25 cycles, together with more detailed information on the hardware used and the effect of adjusting the inference batch size. The 18-layer ResNet is faster than the original DeePaC models and only

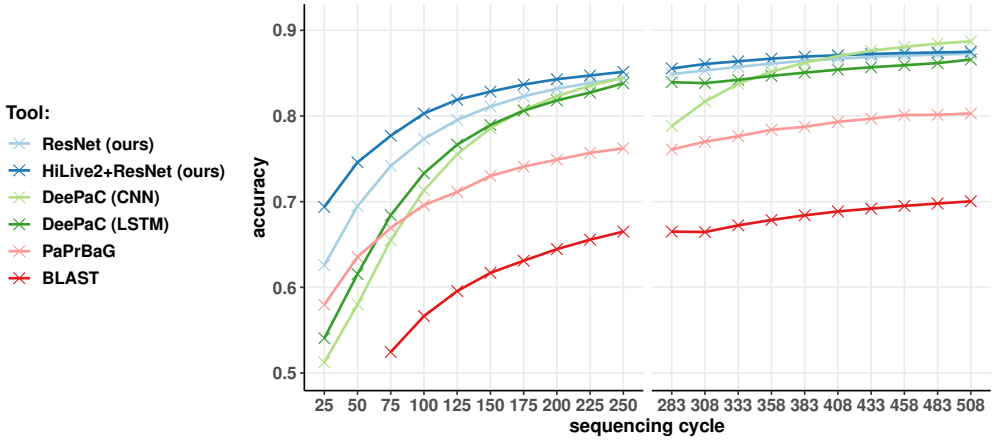


Figure 4.3: Accuracy for the bacterial dataset. Only the tools achieving more than 60% average accuracy are shown (we omit HiLive2). For BLAST, accuracy values below 50% are not shown. Cycle numbers for the second mate are shifted by 8 due to an 8nt-long simulated barcode present in the BCL files, which is removed at mapping and prediction time. PaPrBaG underperforms even though it was retrained specifically for a subread classification scenario. The mapping approach, represented by HiLive2, is the most precise. The low accuracy of both BLAST and HiLive2 reflects a high missing prediction rate (crossing 80% for HiLive2 at cycles 225–250), although BLAST performs better due to its less strict and more sensitive alignment criteria. Combining HiLive2 with the ResNet results in the best average accuracy overall.

Table 4.2: Average performance on reads from novel viruses across the whole sequencing run. ResNet alone achieves the highest accuracy and is the most sensitive method overall, while HiLive2 is the most precise.

	Accuracy	Precision	Recall
HiLive2+ResNet	85.9	93.1	77.6
ResNet	**86.5**	90.5	**81.5**
DeePaC (CNN-150)	84.8	92.3	75.6
DeePaC (CNN)	62.5	48.0	26.7
DeePaC (LSTM)	79.0	90.0	66.0
kNN	60.7	61.7	54.1
BLAST	73.2	97.2	73.7
HiLive2	51.1	**99.2**	50.3

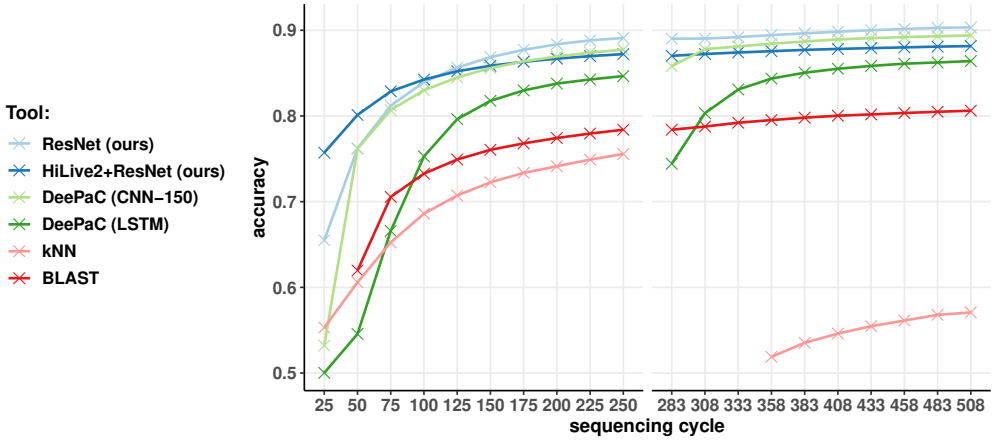


Figure 4.4: Accuracy for the viral dataset. Only the tools achieving more than 60% average accuracy are shown (we omit HiLive2). For BLAST and kNN, accuracy values below 50% are not shown. DeePaC (CNN-150) corresponds to DeePaC-vir's $\text{CNN}_{\text{All-150}}$. DeePaC (CNN) trained on 250bp reads is omitted due to its subpar performance on reads shorter than 200bp (see Table 4.2 and Figure 4.2c). Cycle numbers for the second mate are shifted by 8 due to an 8nt-long simulated barcode present in the BCL files, which is removed at mapping and prediction time. The kNN classifier performs worse than BLAST even for the first mate. Its performance collapses as the second mate is introduced due to many conflicting predictions for both mates, resulting in missing predictions for many read pairs. The ResNet has the best average accuracy overall.

marginally slower than optimized CNNs and LSTMs, analyzing over 56 million reads in the given time-frame (5656 reads/s). This is over 6 times faster than the best non-deep learning based method for both bacteria and viruses, at a much lower cost in terms of computational resources. This prediction speed is enough to guarantee real-time predictions for Illumina iSeq 100, MiniSeq and MiSeq devices, with a maximum of 4 and 25 million reads per run. For sequencers with even higher maximum throughput, like the obsolesced HiSeq and newer NovaSeq 550 machines (with up to 400 million reads per run), further speed-up is possible by distributing the computation across more than one GPU. Multiple receiver instances can be easily assigned to dedicated GPUs to handle different barcodes, cycles or both in parallel. ResNet training time for the maximum of 30 epochs corresponded to 48h on two Tesla V100 GPUs; trained models are available for future use (see 'Data availability').

4.3.4 Real sequencing data

We benchmarked the best bacterial model on data from a real *S. aureus* sequencing run. As this dataset contains reads from just one 'novel' pathogen (a species that was

not present in the training database), the true positive rate (recall) and accuracy are equivalent. The hybrid HiLive2+ResNet classifier crosses the 90% threshold just after 75 cycles (when the true positive rate equals 89.8%) and reaches 98.8% in the last analyzed cycle (Figure A.9). HiLive2 is only able to identify 5.8% of the reads at its best cycle, which drops down to 2.0% at the end of the sequencing run, when longer sequences are analyzed.

We further evaluated the approach on data from a real SARS-CoV-2 sequencing run. Note that the training database did not contain a SARS-CoV-2 reference genome, mimicking the pre-pandemic state of knowledge. In this setting, we used BLAST as an example follow-up analysis of the reads filtered with the hybrid classifier or the pure deep learning approach after just 50 cycles (Table 4.3). As in the case of the open-view DeePaC-vir dataset, the neural network itself performs better than the hybrid classifier, being more accurate even on the reads mappable with HiLive2. More specifically, HiLive2 suffers from a high false negative rate of 96.3% even when only the mapped reads are considered, which is probably because it was designed for mapping against known references. The ResNet is substantially more accurate on the same reads (with a recall of 10.9%, compared to HiLive2's 0.6%). Because of that, omitting the mapping functionality of HiLive2 (and using it just for parsing the BCL files generated by the sequencer) results in better performance. Notably, even the spurious non-pathogenic identifications of HiLive2 can be useful – 99.7% of the mapped reads are identified as originating from coronaviruses, and 90.5% are identified as bat coronaviruses, including the *Rhinolophus* (horseshoe bat) coronaviruses and bat SARS-like viruses, which are probably closely related to SARS-CoV-2.

Similar identifications can be made with BLAST on the much larger set of ResNet-filtered subreads. As the deep learning models consistently outperform BLAST, we use them as predictors to extract subreads of interest. A BLAST follow-up analysis annotates the selected 50bp subreads with their closest taxonomic matches (which may include non-pathogens) wherever a match is found. Alternatively, we can create a consensus predictor by treating BLAST as a confirmatory analysis to focus only on subreads which are predicted to originate from the positive class *and* have a positive BLAST match. In the latter case, we observe a significant enrichment in sequences more similar to the pathogenic SARS-CoV-1 virus. 99.3% of the subreads identified as 'pathogenic' by the consensus ResNet+BLAST workflow are matched with human SARS viruses present in the training database, while the number of identified subreads is almost 15 times higher than for HiLive2 alone. The results suggest that predictions of the ResNet, even without a BLAST follow-up, are also reliable, while offering a recall rate 80 times higher than HiLive2. However, further analysis steps (with BLAST or other approaches, e.g.

Table 4.3: Reads identified as pathogenic from the SARS-CoV-2 sequencing run. The ResNet alone is able to identify the most reads, but cannot annotate them with matches to the closest known references. It is more accurate than HiLive2 even on the reads that the latter is able to map, which explains why it performs better than the HiLive2+ResNet hybrid classifier. Combining HiLive2 or BLAST with the ResNet identifies taxonomic signals while extracting more reads than pure mapping. BLAST output can be used to annotate the reads with the closest taxonomic match only (annot.), or to form a consensus predictor (cons.) by selecting subreads assigned to the pathogenic class by both the ResNet and BLAST.

	Recall	Annot. rate
HiLive2	0.6	0.6%
HiLive2+ResNet	41.0	0.6%
ResNet	**51.3**	0.0%
HiLive2+ResNet+BLAST (annot.)	41.0	27.7%
ResNet+BLAST (annot.)	**51.3**	**37.9%**
HiLive2+ResNet+BLAST (cons.)	6.8	6.8%
ResNet+BLAST (cons.)	9.3	9.3%

taxonomic classifiers) are required to gain more fine-grained insights into the origin of ResNet-filtered reads.

4.3.5 Synthetic biology and biosecurity

The high performance of the models suggests that the protocol could be useful also beyond the sequencing context, e.g. to improve screening workflows for safe and secure synthetic biology, where new methods are needed as engineering of modified pathogens becomes increasingly realistic. Host range of viral pathogens can be deliberately modified (Herfst et al., 2012; Imai et al., 2012), and a virus similar to the Variola virus (the cause of smallpox and a bioweapon) was synthesized (Noyce et al., 2018; Thiel, 2018). Lipsitch and Inglesby (2014) speculated on modifications increasing the pathogenicity of coronaviruses. On the other hand, a report by the National Academies of Sciences, Engineering, and Medicine (2018) sees virulence-enhancing manipulation of existing bacteria as the issue of the highest concern. Computational screening of ordered sequences is a standard, but challenging precaution measure used by the DNA synthesis industry; evaluation of novel sequences requires significant computational resources and expert analysts.

Standard screening approaches rely on homology-based pipelines for pathogen detection and functional annotation (Balaji et al., 2021; Diggans & Leproust, 2019). As they depend on sequence alignment against databases of known threats, it suffers from the

same problems as other taxonomy-dependent pathogen detection methods. Evaluating sequences shorter than 200bp is usually not feasible due to high false positive rates and the computational burden; a PhD-level proficiency in bioinformatics is required to both implement the pipelines and analyze the results (Diggans & Leproust, 2019). Taken together, those challenges warrant investigating deep learning alternatives to traditional workflows.

Comprehensive evaluation of such systems is challenging, as even attempting to design synthetic DNA encoding novel, harmful functions would not be ethically admissible. Importantly, the screening must work well also for known threats, where evaluation is easier. Assessing performance on novel sequences can only be done indirectly, by removing relevant sequences from the underlying database. This approach has been previously mentioned in Bartoszewicz, Seidel, and Renard (2021a) and Bartoszewicz et al. (2020), where the potential of deep learning as a better alternative to purely homology-based identification was first shown, and is also used here. We note that after the completion of this work, an analogous approach was also presented to test the performance of the independently developed SeqScreen pipeline, explicitly focusing on biosecurity applications (Balaji et al., 2021). SeqScreen extends homology-based pathogen sequence identification with predicting the functions of proteins to which successful homology hits have been found.

Our models deliver high accuracy and precision for sequences well below the established 200bp limit, and their false positive rates can be lowered even more if a decision threshold higher than the default 0.5 is used. Given the inference speed of the classifiers, we envision a system where the suspicious sequences are filtered with DeePaC-Live and piped into a follow-up analysis akin to BLAST, lowering the computational burden of sequence alignment and improving the performance.

4.3.6 Nanopore reads

Finally, we evaluated the Nanopore models to investigate possible applications to noisier long-read sequencing technologies (Table 4.4). The Nanopore-trained ResNets achieved higher validation accuracy than the CNNs and LSTMs trained on the same data and were selected for further evaluation. As expected, mapping with minimap2 is the most precise method, and Illumina-trained neural networks of DeePaC underperform in this context. Noisy reads are especially challenging for Illumina-trained LSTMs. Their precision and true positive rates become unstable, resulting in relatively low accuracy. Using Nanopore error models for training promotes more robust models. Strikingly, the first 250bp of a read are enough for the ResNets to noticeably outperform minimap2, even when it

Table 4.4: Performance on Nanopore data. Minimap2 was evaluated on both full reads and 250bp subreads. ResNets were trained on Nanopore data with identical species composition as the Illumina data used for DeePaC CNNs and LSTMs, and evaluated on 250bp subreads. Minimap2 yields no matches for between 13% (viruses, full length) and 69% (bacteria, 250bp) of the reads.

		Accuracy	Precision	Recall
Bacteria	ResNet	**78.5**	73.4	89.3
	DeePaC (CNN)	77.8	73.1	87.8
	DeePaC (LSTM)	73.7	66.5	**95.7**
	minimap2 (250bp)	30.5	**97.2**	46.6
	minimap2 (full)	66.7	91.4	79.7
Viruses	ResNet	**88.5**	90.3	**86.4**
	DeePaC (CNN)	81.4	84.0	77.6
	DeePaC (LSTM)	78.5	92.3	62.1
	minimap2 (250bp)	59.3	**99.2**	61.8
	minimap2 (full)	80.2	98.9	82.4

uses whole reads. This holds for real data as well. When the correct reference is not yet available (as before the pandemic), minimap2 recalls 66.9% of full-length *S. aureus* reads and only 9.9% of full SARS-CoV-2 reads, compared to 94.7% and 52.7% for the ResNets respectively (Table A.8).

These results suggest that the classifiers can find applications in selective sequencing workflows, enabling targeted analysis of reads originating from novel pathogens while discarding potentially less-interesting non-pathogen reads. Although a given read could contain sequences matching to pathogen references located after the initial 250bp, the risk of premature termination seems to be mitigated by the classifier's superior performance, especially in the case of novel viruses. This risk can be further adjusted to the user's needs by selecting an alternative classification threshold, manipulating the expected sensitivity, precision and false positive rates as shown by the receiver operating characteristic (ROC) and precision-recall curves (Figure A.10).

4.4 Discussion

4.4.1 Predictions for subreads and real-time detection

All the limitations of the previously described read-based methods of pathogenic potential prediction (Bartoszewicz, Seidel, & Renard, 2021a; Bartoszewicz et al., 2020; Deneke

et al., 2017; Z. Zhang et al., 2019) apply to this study too. The models presented here assign probability-like scores to DNA and RNA sequences without establishing a mechanistic link between a given sequence and the predicted phenotype. This is, however, an advantage of the proposed approach as well. By resigning from speculation on that link (e.g. by sequence alignment to known relatives), models trained on carefully selected data outperform the traditional methods in terms of both prediction speed and accuracy, yielding predictions for all sequences in the sample. Any assumptions and biases affecting the labels will be reflected by the trained classifier. While this is also the case for read alignment and *k*-mer based classification, they do not require retraining after a database update. On the other hand, as the classifiers generalize well to sequences absent from the training database, they may be updated less frequently while maintaining the desired performance.

A separate question is whether the captured signal has any underlying functional meaning, or is purely taxonomic in nature. BLAST, as a sensitive method of homology detection, is a gold standard for finding taxonomic relationships. Therefore, it could be assumed that outperforming BLAST is a sign of learning more than just evolutionary distances. On the other hand, the very nature of read-based predictions renders any broader biological context inaccessible – the analyzed sequences are simply too short to contain reliable peptide-level features (Bartoszewicz et al., 2020; Deneke et al., 2017), let alone information about the structural or functional characteristics of the encoded proteins (or intergenic regions). However, interpretability workflows like Genome-Wide Phenotype Potential Analysis (Bartoszewicz, Seidel, & Renard, 2021a) have shown that read-based pathogenicity prediction models assign high pathogenic potentials to reads originating from genes engaged in virulence, also specifically in the case of *S. aureus*. What is more, they show that regions of higher pathogenic potential are non-uniformly distributed in both bacterial and viral genomes with 'peaks' of elevated potential aligning with relevant genes. This suggests that even though detecting pathogenicity islands or virulence factors directly is not possible, reads associated with the pathogenic phenotype do originate from important regions of interest. On the other hand, one can expect similar sequences to yield similar predictions. This makes distinguishing between closely related pathogens and non-pathogens challenging, as shown previously for SARS-CoV-2 and its relative, RaTG13 (Bartoszewicz, Seidel, & Renard, 2021a). The problem can be explicitly modelled by including related viruses infecting different hosts (as in the DeePaC-vir dataset used in this study) or by training classifiers targeting novel strains of known bacterial species (Bartoszewicz et al., 2020). Nevertheless, occasional misclassifications of similar sequences are a real possibility that has to be kept in mind. This also applies to alignment-based and *k*-mer based approaches.

The accuracy achieved by the models clearly shows that predicting if a read comes from a bacterial pathogen or a human-infecting virus is indeed possible, even if there is no reference genome available. This may actually be a form of texture bias – CNNs have been shown to correctly classify images based on fragments of as little as 17x17 pixels (Brendel & Bethge, 2019). Here, a local DNA or RNA pattern is often predictive of the phenotype label assigned to the genome. However, the models do not return any information on the closest possible match, which is generally necessary in any pathogen detection task. In this study, we proposed to solve this problem by combining the deep learning approach with an alignment-based one. Alternative methods of taxonomic classification could be used instead – based on either *k*-mers or machine learning. Kraken (Wood & Salzberg, 2014), a *k*-mer approach, was outperformed by both BLAST and PaPrBaG (Deneke et al., 2017). Nevertheless, we can imagine that using a well-trained taxonomic classifier, preferably one yielding at least putative species-level predictions for every read, would be a very useful tool for follow-up analyses of reads prefiltered with the models. On the other hand, since the reads associated with pathogenicity are often co-localized within relevant genomic features (Bartoszewicz, Seidel, & Renard, 2021a), assembly of the filtered reads could recover longer contigs corresponding to genes or perhaps even gene clusters.

A combination of mapping and ResNets could also form a part of more complex real-time pathogen detection workflows like PathoLive (Tausch, Loka, et al., 2018). For example, PAIPline (Andrusch et al., 2018) identifies pathogens in metagenomic and clinical samples via mapping and a BLAST follow-up analysis, but can only start the analysis after the sequencing is finished. Exchanging Bowtie2 (Langmead & Salzberg, 2012) for HiLive2 with a hybrid classifier directing potentially informative reads to the BLAST confirmatory step could serve as a backbone of an extended, real-time version of the pipeline. Alternatively, PathoLive could be extended with ResNet and BLAST follow-up steps. To fully handle metagenomic samples, the classifiers would have to be retrained in a multi-class setting encompassing a broader spectrum of clinically relevant pathogen groups. On the other hand, if there are reasons to believe that the disease-causing agent is a virus or a bacterium, the models presented here may suffice.

As our classifiers rely on either BAM or fasta input, they are not necessarily dependent on HiLive2 and can be used in combination with alternative approaches to accelerated sequencing analysis, for example the DRAGEN system. Nanopore-trained models perform relatively well despite higher sequencing noise, and we imagine incorporating pathogenicity prediction into real-time selective sequencing workflows (Loose et al., 2016). Since 250bp subreads are enough to make predictions more accurate than possible

with mapping even fully sequenced reads, it would be possible to terminate sequencing of some reads quickly to focus on sequencing those originating from pathogens.

4.4.2 Conclusions

We present a new workflow for real-time prediction of the pathogenic potential of novel bacteria and viruses, accessing the intermediate files of an Illumina sequencer. Deep learning models specialized in inference from incomplete short- and long-read sequencing data outperform alternatives on both simulated and real reads. Combining deep learning with homology search recovers reads originating from novel pathogens without sacrificing performance on known agents, highlighting close relatives. The protocol could also be used for sequence-based tasks beyond NGS analysis, for example as a screening system for synthetic DNA sequences difficult to evaluate before. The workflow is available as a command-line tool and a Python package, DeePaC-Live. It can be installed with Bioconda (Grüning et al., 2018), Docker or pip. The code and installation instructions are available at https://gitlab.com/dacs-hpi/deepac-live (real-time inference and HiLive2 integration) and https://gitlab.com/dacs-hpi/deepac (ResNet training and data preprocessing). The datasets are hosted at https://doi.org/10.5281/zenodo.4456857 along the trained models and config files describing the model architecture and hyperparameters (https://doi.org/10.5281/zenodo.4456008).

5 Fungal host prediction and detecting multiple pathogen classes

Summary

Emerging pathogens are a growing threat, but large data collections and approaches for predicting the risk associated with novel agents are limited to bacteria and viruses. Pathogenic fungi, which also pose a constant threat to public health, remain understudied. Relevant, curated data remains comparatively scarce and scattered among many different sources, hindering the development of sequencing-based detection workflows for novel fungal pathogens. No prediction method working for agents across all three groups is available, even though the cause of an infection is often difficult to identify from symptoms alone. We present a curated collection of fungal host range data, comprising records on human, animal and plant pathogens, as well as other plant-associated fungi, linked to publicly available genomes. We show that the resulting database can be used to predict the pathogenic potential of novel fungal species directly from DNA sequences with either sequence homology or deep learning. We develop learned, numerical representations of the collected genomes and show that human pathogens are separable from non-human pathogens. Finally, we train multi-class models predicting if next-generation sequencing reads originate from novel fungal, bacterial or viral threats. The presented data collection enables accurate detection of novel pathogens from sequencing data. It is also a comprehensive resource that can find use beyond this particular task. This can include possible applications in proteomics and genomics, employing both machine learning and direct sequence comparison.

This chapter is based on Bartoszewicz et al. (2022), which is a joint work with Ferdous Nasri, Melania Nowicka and Bernhard Y. Renard. Jakub Bartoszewicz and Ferdous Nasri contributed equally. Jakub Bartoszewicz compiled the final version of the database and performed all analyses presented here. A detailed description of the authors' contributions can be found in section Thesis outline.

5.1 Background

Many species of fungi are dangerous plant, animal, or human pathogens. Importantly, even usually harmless opportunists can be deadly in susceptible populations. For example, *Candida albicans* causes common, relatively benign infections like thrush and vulvovaginal candidosis, affecting up to 75% of women at least once in their lifetime and often re-occurring multiple times (Sobel, 2007). It is also frequently found in healthy humans without leading to any disease, and has been reported to be capable of stable colonization (Raimondi et al., 2019). However, invasive *Candida* infections, especially bloodstream infections, can reach mortality rates of up to 75%, rivaling those of bacterial and viral sepsis (G. D. Brown et al., 2012). A related species, *Candida auris*, has been first recognized in a human patient in 2009 (Satoh et al., 2009) and quickly became one of the most urgent threats among the drug-resistant pathogens (CDC, 2019), reaching mortality rates of up to 60% (Spivak & Hanson, 2018). It might have originally been a plant saprophyte which has adapted to avian, and then also mammalian hosts, possibly prompted by climate change (Casadevall et al., 2019). Strikingly, it seems to have emerged in three different clonal populations on three continents at the same time, for reasons that currently remain unexplained (Lockhart et al., 2017).

Despite their importance, research on fungal pathogens is consistently neglected and underfunded ("Stop neglecting fungi", 2017). Even though fungal infections are estimated to kill 1.6 million people a year, they remain understudied and underreported (Chowdhary et al., 2016; Huseyin et al., 2017; "Stop neglecting fungi", 2017). Estimates suggest that between 1.5 million (Hawksworth, 2001) and 5.1 million (Blackwell, 2011), or even 6 million (D. L. Taylor et al., 2014) different species of fungi exist, but only a small fraction of them has been sequenced.

This poses a major challenge especially for pathogen detection workflows based on next-generation sequencing (NGS). Standard methods are based on recognition of known taxonomic units by homology detection, using either sequence alignment (Ahn et al., 2015; Altschul et al., 1990; Andrusch et al., 2018; Camacho et al., 2009; Hong et al., 2014; Langmead & Salzberg, 2012; H. Li, 2018; H. Li & Durbin, 2010; Naccache et al., 2014), *k*-mer based approaches (Breitwieser et al., 2018; Piro et al., 2020; Wood et al., 2019) or combinations thereof (Piro et al., 2017). This in turn requires curated databases of fungal, as well as bacterial, viral, and other species labelled with information regarding the corresponding pathogenic phenotype or host information. Limited host information is available in the NCBI Genome browser (Sayers, Beck, et al., 2021), Database of Virulence Factors in Fungal Pathogens (T. Lu et al., 2012) and the U.S. National Fungus Collections Fungus-Host Database (Farr & Rossman, 2021). Those

resources are partially complementary, and none of them encompasses all the available data. What is more, multiple literature sources describe fungal pathogens and their hosts without referring to the corresponding genomes, even if they are indeed available in databases such as GenBank (Sayers, Cavanaugh, et al., 2021) or FungiDB (Basenko et al., 2018), which store genomic data without clear-cut host annotation. The ENHanCEd Infectious Diseases Database (EID2) (Wardeh et al., 2015) aims to detect all 'carrier'-'cargo' relationships, not limited to fungi or pathogens specifically, although it does contain fungal pathogens as well. It relies on automatically mining the 'host' field in NCBI Taxonomy (Schoch et al., 2020) and finding co-occurrences of species names in articles indexed by PubMed (Sayers, Beck, et al., 2021), providing links to the associated nucleotide sequences. This method is efficient and scalable, but automated processing based on a concise set of simplifying assumptions may sometimes lead to spurious results. Many 'cargo' and 'carrier' species can be mentioned in the same paper even though one is not really a host of the other. This is often the case in literature reviews, articles discussing phylogenetic classification, or taxonomy updates, and holds also for this work. The 'host' field in a database as large as NCBI Taxonomy may also contain outdated, inaccurate or incomplete information. For example, *Pneumocystis jirovecii*, the causative agent of deadly pneumocystis pneumonia, was previously called *Pneumocystis carinii*. While the latter name is now reserved for a species infecting exclusively rats and not humans (Stringer et al., 2002), records in NCBI Taxonomy (and, possibly by consequence, EID2) still list humans as the hosts of *P. carinii* at the time of writing. What is more, many sequences included in EID2 are not genome assemblies, but single genes, which are not enough for open-view fungal pathogen detection based on shotgun sequencing. For this and similar applications, a new resource is needed. It should be based on three complementary sources: manually confirmed labels mined from the large, general-purpose databases, specialized, fungal databases accumulating biologically relevant evidence for each label, and literature.

In this work, we compiled a collection of metadata on a comprehensive selection of fungal species, annotated according to their reported pathogenicity towards humans, non-human animals, and plants. We also include plant-associated fungi without a clear pathogen annotation. We store the metadata in a flat-file database and link them to the corresponding representative or reference genomes, if available. To showcase the possible applications of the database, we model a scenario of novel fungal pathogen detection. While to our knowledge, this is a first systematic evaluation of feasibility for the novel fungal pathogen detection task, we note that it mirrors similar problems in bacterial and viral genomics (Barash et al., 2018; Bartoszewicz, Seidel, & Renard, 2021a; Bartoszewicz et al., 2020; Bergner et al., 2021; Brierley & Fowler, 2021;

Deneke et al., 2017; Gałan et al., 2019; Q. Guo et al., 2021; Mock et al., 2020; Tang et al., 2015; Wardeh et al., 2021; Z. Zhang et al., 2019). We expect new agents to emerge due to environmental changes, host-switching events and growing human exposition to the unexplored diversity of potentially harmful fungi, as shown by the example of *C. auris*. What is more, advances in recombinant DNA assembly allow construction of whole synthetic fungal chromosomes and genomes (Burgess, 2017; Dai et al., 2020; Richardson et al., 2017), as well as optimization of yeast strains with new phenotypes (Luo et al., 2018; Szymanski & Calvert, 2018). Genetic modifications of filamentous fungi also grow in popularity (Amores et al., 2016; Martins-Santana et al., 2018). As some of them are dangerous pathogens (Chowdhary et al., 2016), this may lead to the development of new dual-use applications and potential bioterrorism risks. Biosecurity and biosafety regulations are supported by computational screening of ordered, potentially novel, sequences at DNA sequencing facilities, using adaptations of methods developed originally for pathogen detection from sequencing data (Balaji et al., 2021; Bartoszewicz, Genske, et al., 2021a; Diggans & Leproust, 2019).

Therefore, we evaluate if detecting homology between previously unseen species and their known relatives accurately predicts if a novel fungus is capable of colonizing and infecting humans. BLAST (Altschul et al., 1990; Camacho et al., 2009) represents the gold standard in pathogen detection via taxonomic assignment to the closest relative. Although read mappers or k-mer based taxonomic classifiers are more computationally efficient on large NGS datasets (Alser et al., 2021; Breitwieser et al., 2017; Ye et al., 2019), BLAST has been shown to be more accurate in similar tasks of detecting novel bacterial and viral pathogens (Bartoszewicz, Seidel, & Renard, 2021a; Deneke et al., 2017). However, convolutional neural networks of the DeePaC package have been proven to outperform BLAST in both those scenarios (Bartoszewicz, Seidel, & Renard, 2021a; Bartoszewicz et al., 2020) for both isolated NGS reads and full genomes, and a recently presented variant of residual neural networks (ResNets) outperforms all alternatives on short NGS reads and their fragments (Bartoszewicz, Genske, et al., 2021a). Therefore, we also train a similar ResNet to predict if a novel fungal DNA sequence originates from a potential human pathogen. We then develop trained numerical representations of all genomes in the database and show the separation of potential human pathogens and non-human pathogens. Finally, as previously presented models were limited exclusively to bacteria or viruses, we extend our neural networks to a multi-class setting, which we envision to be applicable in open-view, clinical scenarios with only minimal assumptions regarding a novel, previously undiscovered causative agent. Following the convention introduced previously for bacteria (Deneke et al., 2017) and viruses (Bartoszewicz, Seidel, & Renard, 2021a), we focus on the pathogenic potential

Table 5.1: Key information on each species included in the database.

Name	Species name, GenBank organism name and infraspecific name for the representative genome.
NCBI TaxID	Taxonomic identifiers of the species and the representative genome in the NCBI Taxonomy database.
Assembly accession & name	GenBank assembly accession number and assembly name for the representative genome.
Pathogenicity & host group	Manually curated information regarding pathogenicity and the relevant host groups.
Putative host group	Automatically extracted information suggesting a host group. Includes non-pathogenic associations.
Label sources	References to resources used to label the species for all host groups and evidence levels.
Source name and TaxID	Species name and TaxID as mentioned in the first resource used to label it.
Assembly Level	Assembly level of the representative genome.
Sequence release date	Sequence release date as reported in GenBank.
FTP path	GenBank FTP path to the representative genome.
Label date	Date of adding the species to the database or updating its labels.

of the analyzed species, rather than the pathogenicity itself. This distinction captures the fact that the pathogenic phenotype may or may not be manifested depending on specific host-pathogen interactions; whether this happens cannot be fully predicted outside the context of a particular host. We can only predict if a given sequence originates from a species that could cause disease under some circumstances (i.e. its pathogenic potential), not whether an invasive infection actually occurred.

5.2 Data description

The database presented here is designed to facilitate sequence analysis and machine-learning approaches to pathogen detection. Therefore, we use the concept of a 'positive' and a 'negative' class, where the former corresponds to the species pathogenic to humans, and the latter can be defined in many ways depending on the downstream task. In our showcase of example applications of the database, we focus on pathogens (and hence our negative class consists of non-human animal and plant pathogens), but we include more records in the database for future use. More specifically, we collect metadata on species infecting humans, animals or plants, supplemented with information on other plant-associated species. To this end, we integrate multiple literature and database sources, relying on manual curation, but also including the automatically extracted data for future reference. Table 5.1 summarizes the data we collected for each species.

First, we accessed the Database of Virulence Factors in Fungal Pathogens (DFVF) (T. Lu et al., 2012) on October 9, 2021. The database contains records on virulence factors of a wide selection of fungal species together with their NCBI TaxIDs and information on their hosts and the disease caused. It also includes separate parts for animal and plant pathogens. We searched the records for all proteins in the database; if the 'Disease' or 'Disease host' fields contained the word 'human', we added the corresponding species

to the list of human pathogens. We added the remaining species from the animal part of the database to the list of animal pathogens. Finally, we put all species mentioned in the plant part of the database on the list of plant pathogens, irrespective of whether humans were also mentioned as a possible host. For all those species, we also extracted the TaxID. As some of those TaxIDs corresponded to taxa below the species level, we linked them to their species-level ancestors in the NCBI Taxonomy tree. This resulted in a list of unique, labelled species names and species-level TaxIDs. Note that as DFVF was built using manual curation of text-mining results, we treat the obtained records as manually curated as well.

Further, we searched the Taxonomy database (Schoch et al., 2020) using the query '"host human"[Properties] AND Fungi [orgn]', obtaining a list of species with a putative human host and their TaxIDs. We then used a comprehensive selection of literature reviews and comparative studies (Bombassaro et al., 2020; Brunner-Mendoza et al., 2019; Chowdhary et al., 2014; Colombo et al., 2011; Dellière et al., 2020; Franzen & Müller, 2001; Han & Weiss, 2017; Hu et al., 2015; Jančič et al., 2015; Kjærbølling et al., 2018; Köhler et al., 2015; Kwon-Chung et al., 2017; Paulussen et al., 2017; Prakash et al., 2017; Seyedmousavi et al., 2018; L. H. Taylor et al., 2001; Teixeira et al., 2017) to collect additional evidence and extend the collected list of human pathogens. We manually extracted the species names and searched the NCBI Taxonomy (including synonyms) to link them to their names as reported in Taxonomy and the respective species-level TaxIDs. In both cases (searching with extracted TaxIDs or species names), we handled the possibility of multiple, ambiguous matches by preferring exact name matches to any of the synonymous names recorded in Taxonomy. If this was not possible (e.g. because of additional annotations in the 'name' field in Taxonomy), we considered matches where the first two words exactly match the query name, and varieties of a query species (i.e. matches containing the abbreviation 'var.'). If no exact match was found, we selected the first record containing the query species name. We ignored species hybrids and *formae speciales*, excluded the organism names explicitly expressing taxonomic ambiguity (marked with 'sp.', 'cf.' or 'aff.'), and filtered out fungal viruses. If the species name was not mentioned in Taxonomy at all, we still included it in the database along with its associated host group label. This will enable easy extension of the database in the future, as more fungal sequences become available.

Similarly, we consulted the literature on animal (Bałazy et al., 2008; Becnel & Andreadis, 2014; Chandler et al., 2000; Cissé et al., 2021; Evans et al., 2011; Franzen & Müller, 2001; Gerson et al., 2008; Han & Weiss, 2017; Leonhardt et al., 2018; Lovett & St. Leger, 2017; Rehner & Buckley, 2005; Schulenburg & Félix, 2017; Seyedmousavi et al., 2018; Seyedmousavi et al., 2015; Y. Shang et al., 2015; Smith, 2006; St. Leger &

Wang, 2020; Sung et al., 2007; Teixeira et al., 2017; van der Geest et al., 2000; S. Wu, Toews, et al., 2021) and plant pathogens (Barbara & Clewes, 2003; Costa et al., 2021; Coutinho et al., 2017; Doehlemann et al., 2017; Gurung et al., 2015; D. R. Jones & Baker, 2007; S. W. Kim et al., 2012; W. Kim et al., 2019; Leonhardt et al., 2018; Schardl et al., 2013; Stukenbrock et al., 2012; Vacher et al., 2008) and linked the results to the respective species TaxIDs wherever possible. Further, we accessed the Fungal Databases of the Germplasm Resources Information Network (GRIN). On October 9, 2021, we searched for all records with 'Homo sapiens' as a host (Farr & Rossman, 2021). We also added the species listed in the Nomenclature Fact Sheets for plant-associated fungi with quarantine importance and the Fungal Diagnostic Fact Sheets of invasive and emerging fungal pathogens (Farr & Rossman, 2021) to the list of plant-pathogenic species, and searched NCBI Taxonomy as described earlier.

We then linked the resulting lists of species TaxIDs with their representative of reference genomes in Genbank, accessed on October 9, 2021. Further, we selected all species with available genomes, but without a label assigned by any of the sources used. On the same day, we manually searched the GRIN Fungal Database for each of those species, allowing for synonyms. We checked the 'Host, 'Disease', and 'Notes' fields; if the disease of a plant or animal host was clearly described, we added the query species to the appropriate list. If no disease was mentioned for a confirmed plant host, we added the species to a list of plant-associated fungi.

Next, we used the EID2 database (Wardeh et al., 2015) to extract species with putative human, animal, or plant hosts. Note that those labels were collected automatically and are therefore prone to errors. Moreover, they do not consist of pathogens only, and may include commensals or symbionts. We nevertheless added them to our database for completeness, albeit noting that they represent only a putative, automatically extracted carrier-cargo relationship, rather than a manually confirmed, true pathogenic potential of a species. The same, 'putative' category includes human-hosted species found in NCBI Taxonomy and GRIN, unless confirmed in other literature sources as well. We then selected all species with a putative human host and manually searched the Atlas of Clinical Fungi (de Hoog et al., 2020) with their names and synonyms noted in NCBI Taxonomy, also considering additional species referred to in the search results. If the Atlas confirmed a species to be a pathogen, we added it to the appropriate category. Note that only three of the human-hosted species (all belonging to the genus *Malassezia*) were clearly described as human commensals without any reports of causing disease (and hence, we excluded them from our list of pathogens). In cases when the Atlas mentioned two names to be synonyms, even though NCBI Taxonomy listed them as two separate species, we followed the nomenclature suggested by the Atlas. We retained both records

to explicitly reflect this in the database, keeping one name unlabelled and linking it to the main record with an appropriate annotation regarding the synonym TaxID and name. Finally, we linked all labelled species to their GenBank genomes, if available.

After 12 weeks, we updated the database by extracting the records on reference and representative fungal genomes present in GenBank on January 2, 2022, selecting those with accession numbers and TaxIDs not already present in our database, and filtering the organism names as described above. We then linked the 67 new genomes to the previously collected metadata (29 of those species were absent from the first version of the database), and manually searched GRIN following the same procedure as for the main part of the database. If a human host was mentioned, we added the corresponding putative label to the appropriate list. Finally, we checked all putative human host labels in the Atlas of Clinical Fungi (de Hoog et al., 2020). The update added 27 new labelled genomes, including 12 plant-associated fungi without proven pathogenic labels. One plant pathogen and four plant-associated species were absent from the first version of the database. We used the added pathogen genomes for our temporal benchmark.

The database contains 14,555 records in total. In the following parts of the manuscript, we will focus on what we will call the core database, comprising metadata on genomes of 954 manually confirmed pathogens (including 332 species reported to cause disease in humans), available on October 9, 2021. This forms a collection of species most relevant to the pathogen detection task, belonging to 6 phyla, 37 classes, 82 orders and 182 families. A 'temporal benchmark' subset contains 15 further pathogens (including one infecting humans), collected in a database update on January 2, 2022. We also include records on 486 plant-associated fungi. The supplementary part of the database contains information on 481 putatively labelled genomes, 1,147 unlabelled species with available genomes, 8 synonyms (with 6 alternative genomes) derived from the Atlas of Clinical Fungi (de Hoog et al., 2020), 885 labelled species without genomes (including 284 species without TaxIDs) and 10,579 putatively labelled species without genomes (including 9 without TaxIDs). This subset will enable easy updating of the database in the future, as more genomes of already labelled species are sequenced. It also serves as a record of all screened genomes and species to ensure reproducibility and facilitate future extensions (e.g. adding new data or sources of evidence). Figure A.11 presents the logical relations between the genomes with manually confirmed labels and genomes for which putative labels could be found in EID2.

5.3 Methods

5.3.1 Training, validation and test sets

While we envision a wide range of possible applications of the database, we present an example use-case that allows taking advantage of the wealth of collected data – detection of novel fungal pathogens from NGS data. The core of the database contains 332 genomes of human pathogens (including opportunists), forming the positive class. The negative class comprises 622 species not reported to infect humans; this includes 565 plant pathogens and 58 non-human animal pathogens. To evaluate the performance of the selected pathogenic potential prediction methods, we divided the corresponding genomes into non-overlapping training, validation and test sets. In this setup, the training set is used as a reference database for the methods based on sequence homology and to train the neural networks, while the performance metrics are calculated on the held-out test set. The validation set is used for hyperparameter tuning and to select the best training epoch. While evaluating performance on genuinely unknown species is by definition impossible, we effectively model the 'novel species' scenario by testing on a wide range of sequences removed from the database.

We manually placed two clinically important pathogens – *Candida auris* (a recently emerged threat (CDC, 2019; Satoh et al., 2009)) and *Aspergillus fumigatus* (a causative agent of aspergillosis, reaching up to 95% mortality rates (G. D. Brown et al., 2012)) – in the test set. We also added two non-human pathogens: *Pyricularia oryzae* (syn. *Magnaporthe oryzae*; voted the most important fungal pathogen by a panel of almost 500 fungal pathologists (Dean et al., 2012) as it causes up to 30% of global rice production losses (Skamnioti & Gurr, 2009)) and *Batrachochytrium dendrobatidis* (blamed for recent decline of 500 amphibian species and complete extinction of 90 of them (Scheele et al., 2019)). The rest of the species were assigned randomly. Overall, we placed 10% of all species in test and validation sets each, while the remaining 80% formed the training set.

To model the task of predicting pathogenic potentials from NGS samples, we simulated training, validation and test reads using Mason (Holtgrewe, 2010). First, we generated 10 million single reads, 1.25 million single reads, and 1.25 million paired reads per class for the training, validation and test sets respectively, keeping the number of reads per genome proportional to each genome's length. This mirrors the protocol used for bacteria (Bartoszewicz et al., 2020) and viruses (Bartoszewicz, Seidel, & Renard, 2021a), resulting in equal mean coverage for all genomes. A side effect of such an approach is that reads originating from longer genomes could be over-represented compared to

those from shorter or incomplete genomes, possibly causing generalization problems for trained machine learning models. To tackle this issue, we also simulated a second version of the training set, where the number of reads per genome is proportional to a logarithm of a given genome's length. We call this version of the training set the 'logarithmic-size' set to differentiate it from the previous ('linear-size') approach. Note that while the 'logarithmic-size' training set may help balance the resulting classifiers' performance, in a real sample, we expect the coverage to be approximately equal for equally abundant species. Therefore, we only used the 'linear-size' versions of validation and test sets. In both cases, the datasets maintain the 8-1-1 proportions on both read and genome levels, with a total count of 25 million reads per class.

A crucial difference compared to previous pathogenic potential prediction studies is that fungal genomes are orders of magnitude longer than bacterial or viral ones. While the procedure mentioned above produced a mean coverage of 1.82 for the bacterial dataset presented in (Bartoszewicz et al., 2020), it results in a mean coverage of only 0.15 on our data. We expected this could cause training problems, as the machine learning models would only have access to a small fraction of the overall sequence diversity of the training set. Therefore, we also simulated 'high-coverage' versions of both the 'linear-size' and 'logarithmic-size' training sets. In this setup, we increased the total number of training reads to 240 million, compared to 20 million for the 'low-coverage' versions described before. The 'high-coverage' versions keep the mean coverage of 1.82. As the reads were simulated randomly, we expected the 'low-coverage' validation and test sets to be representative, and correctly model a common case of low abundance of the target pathogen compared to other DNA sources in the sample (e.g. the host). We therefore used only 'low-coverage' validation and test sets. In summary, we generated four versions of the training read set and one version of the validation and test set each. The datasets contain the same number of reads for each of the classes and can be easily reused for future machine learning and benchmarking applications. Finally, we simulated a temporal test set based on the genomes added to GenBank up to 12 weeks after the main part of the database was compiled. We generated 625,000 reads using the 'linear-size' setup and balancing the number of reads per class; the average coverage matched our main test set.

5.3.2 Pathogenic phenotype prediction

Next, we evaluated the feasibility of pathogenic potential prediction for novel fungal species. We used a ResNet architecture implemented in the DeePaC package, previously shown to outperform alternatives based on deep learning, traditional machine learning

and sequence homology in the context of novel bacteria and viruses (Bartoszewicz, Genske, et al., 2021a). Briefly, the architecture consists of 17 convolutional layers of between 64 and 512 filters (with the filter size of 7 for the first layer and 5 for the following layers) using skip connections, followed by a global average pooling layer and a single-neuron output layer with sigmoid activation. It returns an output score between 0 and 1, where a threshold of 0.5 is used by default as a boundary between 'positive' and 'negative' predictions. The network uses batch normalization and input dropout, which can be understood as randomly switching a predefined fraction of input nucleotides in training samples to Ns. It also guarantees identical predictions for any given sequence and its reverse-complement via parameter sharing. For more details, we refer the reader to the corresponding publication (Bartoszewicz, Genske, et al., 2021a).

As previous research reported that the input dropout rate is an important hyperparameter (Bartoszewicz, Genske, et al., 2021a; Bartoszewicz, Seidel, & Renard, 2021a; Bartoszewicz et al., 2020), we investigated two different values – the default 0.25 and 0 (no dropout). To this end, we trained the networks using the 'linear-size' and 'logarithmic-size' versions of the low-coverage training set for the default maximum of 30 epochs with an early stopping patience of 10 epochs, selecting the epoch and input dropout rate resulting in the highest accuracy on the validation set. We then retrained the networks using the selected input dropout rates and both versions of the high-coverage training set.

As shown in (Bartoszewicz et al., 2020), predictions for two reads in a read pair can be averaged, improving prediction accuracy; training the networks on single reads allows to use them for both single reads and read pairs. Further, averaging over predictions for all reads originating from the same organism (e.g. for classification of whole genomes or single-species samples) has been demonstrated to yield accurate species-level predictions (Bartoszewicz et al., 2020; Deneke et al., 2017). We adapted those steps to our models as well. However, since the networks are explicitly trained for read-based classification only, the default classification threshold may be suboptimal for the genome-wise classification scenario. Therefore, we hypothesized that it should be retuned for optimal performance according to a preselected metric, e.g. balanced accuracy. To test this assumption on a fully independent dataset first, we used the 'novel viral species' dataset from Bartoszewicz, Seidel, and Renard (2021a) and their convolutional neural network (CNN) without any retraining. We selected the threshold optimizing the balanced accuracy on the appropriate validation set, rounded to two decimal places. We compared the original and retuned models and applied the procedure resulting in the better test set performance to our fungal models.

To classify novel fungal sequences based on their closest taxonomic matches, we used BLAST (Altschul et al., 1990; Camacho et al., 2009) as described for bacterial data in (Bartoszewicz et al., 2020). The reference database was created from the training set genomes. We used discontiguous megablast with an E-value cutoff of 10 and default parameters, selecting the label of the top hit ('pathogenic to humans' or not) as a predicted label for each query sequence. Note that if no hits are found, no prediction can be returned for a given read. This lowers the true positive and true negative rates, defined as the ratios of correct positive or negative predictions to all reads in the positive or negative class, respectively. For read pairs, a single match is enough to assign a label to a pair, but conflicting matches result in no prediction. Similarly to Bartoszewicz, Seidel, and Renard (2021a), we considered two approaches for generating species-level predictions. First, we found BLAST matches of contigs from the test set, using a majority vote over all contigs belonging to the same species to assign a corresponding label. Second, we used the majority vote over predictions for all reads originating from a given species. The latter case is especially relevant for a use-case of a sequencing sample containing a single species, and is more directly comparable to the genome-level evaluation of the ResNet.

5.3.3 Genome representations and dataset structure

To visualize the structure of the dataset as learned by the trained classifier, we developed numerical representations for the collected genomes. This poses a challenge, as the networks are trained on reads rather than full genomes. However, we observe that final outputs of the network can be averaged over all reads originating from a single genome to generate a prediction for the genome in question (Bartoszewicz et al., 2020). Analogously, we can average the activations of the intermediate layers to construct vector representations for whole genomes based on the corresponding reads. Note that averaging the activations of the penultimate layer is approximately equivalent to using a full genome as input (assuming full coverage), as our architecture uses global average pooling just before the output layer.

More formally, by averaging the activations of the penultimate layer for a set of reads belonging to the same genome and passing the output to the final layer, we obtain the value $s(E[z])$, where $s(x) = 1/(1+e^{-x})$ is the sigmoid activation and $z = \mathbf{w}^\top \mathbf{h} + b$, for the penultimate activation vector $\mathbf{h}$, weights of the last layer $\mathbf{w}$ and its bias b. Note that this is *not* mathematically equivalent to $E[s(z)]$, or the expected output of the network for reads originating from a given genome, which was used for the genome-level predictions in previously published literature (Bartoszewicz, Seidel, & Renard,

2021a; Bartoszewicz et al., 2020), as well as in this study. However, we suspected that although usually $s(E[z])$ and $E[s(z)]$ are not identical, those terms could have similar values in practice, as they both express a notion of aggregating information contained in individual reads to yield a prediction for a full genome. If this is the case, mean activations $E[\mathbf{h}]$ for a set of reads originating from the same genome can be used as if they were the internal genome representations of our species-level classifiers, even though the models rely on aggregated read-level predictions $E[s(z)]$. To investigate the relationship between $s(E[z])$ and $E[s(z)]$, we performed additional experiments.

Note that $E[s(z)]$ represents averaging in in the space of predicted probabilities ('proba'-average), while $s(E[z])$ is averaging in the logit space ('logit'-average). To check if $s(E[z]) \approx E[s(z)]$, we plotted the 'logit'-average predictions for each species against their 'proba'-average equivalents. As any effects found could be dataset-dependent, we perform this not only on the fungal validation and test sets, but the 'novel viruses' dataset used for the DeePaC-vir networks (and also for our multi-class classifiers). Further, we use simulated logit values (z) to gain deeper insight into the relationship between both versions of read averaging. To accurately model the problem of aggregating predictions for individual species in the context of binary classifications, we first defined two classes, '0' and '1'. We then simulated 100 'species' belonging to each class. Each species S is described by a distribution of logit values $z \sim \mathcal{N}(\mu_S, \sigma_S^2)$, where μ_S and σ_S are the mean and standard deviation of z for that particular species and $\mathcal{N}$ is the normal distribution. To generate μ_S and σ_S for each species, we sample from additional distributions: $\mu_S \sim \mathcal{N}(\mu_c, \sigma_c^2)$ and $\sigma_S \sim HalfNormal(\mu_\sigma, \sigma_\sigma^2)$, where $HalfNormal$ is the half-normal distribution and c is the class index for classes '0' and '1'. Note that for simplicity, we sample σ_S from the same distribution irrespective of the species's class. Finally we generate 1000 values of z per species (corresponding to 1000 reads) and plot the 'logit'-averages $s(E[z])]$ against the 'proba'-averages $E[s(z)]$. Hence, to each species belonging to a given class, we assign its own mean and standard deviation used to generate read representations according to the species's class. First, we perform two experiments using the following parameters:

- $\mu_0 = -3, \mu_1 = 3, \sigma_0 = \sigma_1 = 2, \mu_\sigma = 0.25, \sigma_\sigma = 0.25$
- $\mu_0 = -3, \mu_1 = 3, \sigma_0 = \sigma_1 = 2, \mu_\sigma = 3, \sigma_\sigma = 1$

This models two well-separated classes, and the only difference between the two settings is the distribution of within-species variances of z.

We then test if the relationship between 'logit'- and 'proba'-average changes if no separable classes are present in the dataset. To this end, we perform two additional experiments with only one 'class', '0':

- $\mu_0 = 0, \sigma_0 = 3, \mu_\sigma = 0.25, \sigma_\sigma = 0.25$
- $\mu_0 = 0, \sigma_0 = 3, \mu_\sigma = 3, \sigma_\sigma = 1$

Finally, we investigate if any effects found could be conditional on the presence of a species-level signal, i.e. on differences in the distribution of z for each species S. We generate 100,000 values of $z \sim \mathcal{N}(\mu_r, \sigma_r^2)$, where $\mu_r = 0$ and $\sigma_r = 20$, and assign the 'reads' to 100 simulated 'species' arbitrarily (first 1000 reads to the first species, second 1000 reads to the second species, etc.). We also repeat the experiment simulating 5000 'reads' per 'species' (corresponding to a 5-fold increase in coverage).

Assuming that $s(E[z]) \approx E[s(z)]$ (see Figures A.12-A.13), we extracted penultimate activations for all simulated reads in the low-coverage, 'linear-size' training, validation and test sets. We then used the averaged activation vectors for each species to map the distances between them as learned by our networks. The last convolutional layer has 512 filters, but we removed the filters for which activations were equal to zero for all samples in the dataset and used UMAP (McInnes et al., 2020) to embed the resulting representations in a 2-dimensional space for visualization. We used the Euclidean distance metric with a minimum possible distance of 0.1 and a neighbourhood size of 15. The inputs had been randomly shuffled beforehand to avoid artefacts that can appear if an embedding is learned based on representations ordered by class.

5.3.4 Multi-class evaluation

Finally, we investigated an application requiring merging the 'positive' subset of our database with previously available resources for pathogenic potential prediction in bacteria and viruses (Bartoszewicz, Genske, et al., 2021a; Bartoszewicz, Seidel, & Renard, 2021a; Bartoszewicz et al., 2020). We aimed to integrate the separate classifiers for fungi, bacteria and viruses into a single, multi-class model capable of predicting whether unassembled NGS reads originate from (possibly novel) pathogens present in a human-derived sample. To this end, we extended the DeePaC package adding the multi-class classification functionality. The resulting architectures differ from the previously described ResNets (Bartoszewicz, Genske, et al., 2021a) only by the output layer, which has as many units as the number of considered classes and uses a softmax activation. They use the default input dropout rate, as required by the addition of viral and bacterial data. We train the models for 50 epochs due to the increased difficulty of the task.

We note that in practice, only human-hosted fungi are expected to be found in clinical samples. In this context, a slightly constrained view is admissible: we assume that only

human-pathogenic fungi, human-hosted bacteria (pathogenic or commensal), human viruses and non-human viruses (mainly bacteriophages) will be present in the sample. Further, bacteriophage sequences tend to be very similar to the sequences of their bacterial hosts (Zielezinski, Barylski, et al., 2021; Zielezinski, Deorowicz, et al., 2021) and difficult to differentiate, but both commensal bacteria and non-human viruses can be viewed here as a joint 'negative' (i.e. harmless) class. Hence, learning a precise decision boundary between them can be omitted. Human reads can be ignored, as they can be relatively easily filtered out with traditional methods based on read mapping of k-mers (Ahmed et al., 2021; Loka et al., 2018; Wood et al., 2019).

Therefore, we fused the previously published datasets used in DeePaC (Bartoszewicz, Genske, et al., 2021a) for bacteria (pathogens vs. commensals) and viruses (human vs. non-human) with the 'positive' (human-pathogenic) class of our database. As the original datasets contain only 12.5 million reads per class (with the same 8-1-1 proportion of the training, validation and test sets, stratified by bacterial species or virus), we used the low-coverage versions of our 'positive' read sets, matching this number. We also merged the original negative classes (commensal bacteria and non-human viruses) into a joint non-pathogen class, downsampling accordingly. Further, the original datasets contain reads of between 25 and 250bp instead of 250bp only, which makes the problem more challenging, but increases robustness on very short sequences (Bartoszewicz, Genske, et al., 2021a). To make our fungal reads compatible with this setup, we randomly shortened the reads in our validation and low-coverage training sets to 25–250bp. We only used the test sets of 250bp in order to highlight the upper performance limit of the resulting classifier.

The final result is a dataset divided into 4 classes: nonpathogenic bacteria and non-human viruses, bacterial pathogens, human-infecting viruses, and human-pathogenic fungi, in either the 'linear-size' or the 'logarithmic-size' variant. Each of the classes contains 12.5 million reads divided into the training, validation and test subsets while keeping separate species or viruses in each of the subsets. This is intended to model a scenario of analyzing a clinical sample containing a mixture of previously unseen pathogens and non-pathogens, while keeping the task feasible by resigning from differentiating between bacteriophages and commensal bacteria, as well as fungi with human and non-human hosts. Note that even in this case, the 'negative' part of the presented database is useful, allowing us to constrain our view to a curated set of clinically relevant fungi only.

Using this dataset, we trained two models including all four classes (using the 'linear-size' or the 'logarithmic-size' variant of the fungal training set). We further evaluated the one resulting in higher validation accuracy and a simple ensemble averaging the

predictions of both models. Then, we trained a 3-class model including only the bacterial and viral classes. This allows us to measure the 'difficulty' of integrating the fungal dataset with the others within a single network in terms of resulting differences in prediction accuracy on the original DeePaC datasets. By comparing the performance of our models to the performance of the original binary classifiers (Bartoszewicz, Genske, et al., 2021a), we can disentangle the 'difficulty' of adding the fungal class from the 'difficulty' of integrating the bacterial and viral classes, and assess how much performance is 'lost' by using a more open-view classifier. Note that in the case of the purely viral dataset, spurious assignments to the bacterial pathogens class may be treated as detection of bacteriophages infecting the bacterial species of this class, and hence reassigned into predictions for the non-pathogen class by adding the predicted probabilities for both classes. This effectively merges the non-pathogen and bacterial pathogen classes at test time when appropriate, but still keeps the possibility to use the trained networks in a fully open-view setting (with all classes) without the need for retraining. We performed an additional comparison to BLAST with a pre-selected training database (bacterial for bacteria (Bartoszewicz et al., 2020), viral for viruses (Bartoszewicz, Seidel, & Renard, 2021a)). This resulted in an estimated upper bound on the performance of non-machine learning approaches on those datasets (as extending the training database with irrelevant reference genomes can only lower BLAST's performance). Finally, we also evaluated the neural networks on the full dataset of all four classes, reporting performance measures for each class separately and the 'macro' average performance over all of them.

5.4 Results

5.4.1 Fungal pathogenic potential prediction

The networks trained without input dropout achieved higher validation accuracy compared to those trained with dropout, and the 'logarithmic-size' training set resulted in higher accuracy than the 'linear-size' set in all cases. As expected, the 'high-coverage' training set also improved performance, although the training time of around 8h45' per epoch on four Tesla V100 GPUs was much higher than around 45 minutes per epoch for the 'low-coverage' variant on the same hardware. The best network, trained on the high-coverage, 'logarithmic-size' dataset without dropout, required 8 days of training on four Tesla V100 GPUs and was selected for further evaluation. Proper retuning of the classification threshold for species-level predictions appears to be a necessary step for an independent, viral dataset (Table 5.3), so we also retuned the threshold (0.46 instead of the default 0.5) for the respective fungal ResNet setup.

Table 5.2: Classification performance comparison between BLAST and the ResNet model on single reads, read pairs and genomes from the test dataset. The best performance value for each performance metric (row) and setting is displayed in bold. Read datasets are balanced, and the differences in performance for the first and second mate were negligible for all metrics. In genome datasets, negative class is the majority class. We use BLAST as shown in Bartoszewicz et al. (2020); since this only returns binary labels, AUC and AUPR are not defined. Overall, BLAST is much more precise, but the ResNet yields only slightly less accurate predictions for all reads, even those impossible to match with BLAST, in a fraction of the time. AUC and AUPR are marginally higher (95.2 and 90.2) for the genome-level ResNet if only first mates are used compared to using both mates (95.1 and 90.1). Other metrics are equal, and the computation time is 50% lower. BAcc. – balanced accuracy (equivalent to accuracy on read sets), Prec. – precision; Rec. – recall, Spec. – specificity, AUC – area under the ROC curve, AUPR – area under the precision-recall curve, Time (CPU) – prediction time on 2x AMD EPYC 7742 (256 threads), Time (GPU) – prediction time on 1x Tesla V100 GPU (not possible for BLAST), Pred. – prediction rate.

		BAcc.	Prec.	Rec.	Spec.	AUC	AUPR
Single reads	ResNet	64.7	65.8	**61.3**	68.1	70.9	72.0
	BLAST	**66.2**	**92.6**	61.1	**71.3**	-	-
Read pairs	ResNet	68.5	69.8	**65.1**	71.9	75.7	76.5
	BLAST	**69.7**	**94.5**	62.3	**77.2**	-	-
Genomes	ResNet	88.4	77.5	**91.2**	85.7	95.1	90.1
	BLAST (reads)	**90.3**	**90.6**	85.3	**95.2**	-	-
	BLAST (contigs)	89.5	87.8	85.3	93.7	-	-

		Time (CPU)	Time (GPU)	Pred.
Single reads	ResNet	**25 min.**	**3 min.**	**100.0**
	BLAST	181 min.	-	77.6
Read pairs	ResNet	**50 min.**	**6 min.**	**100.0**
	BLAST	362 min.	-	79.8
Genomes	ResNet	**25–50 min.**	**3–6 min.**	**100.0**
	BLAST (reads)	181–362 min.	-	**100.0**
	BLAST (contigs)	2190 min.	-	**100.0**

Table 5.3: Classification threshold tuning for full available genomes on the 'novel viral species' dataset of DeePaC-vir (Bartoszewicz, Seidel, & Renard, 2021a). Average prediction over all reads from a genome are used as a prediction for a given species. BLAST (reads) corresponds to a majority vote over all reads from a genome, and BLAST (genome) to the majority vote over all contigs. The negative class is more numerous in the dataset, which affects the precision values. The retuned classifier uses the threshold of 0.45 instead of the default 0.5 and achieves the best balanced accuracy. BAcc. – balanced accuracy, Prec. – precision; Rec. – recall, Spec. – specificity.

	BAcc.	Prec.	Rec.	Spec.
CNN (retuned)	**75.9**	31.1	**70.8**	81.1
CNN (Bartoszewicz, Seidel, & Renard, 2021a)	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

Table 5.4: Classification performance comparison between BLAST and the ResNet model on single reads and read pairs from the temporal dataset. The best performance value for each performance metric (row) and setting is displayed in bold. While the read number per class is balanced, the dataset is small and extremely imbalanced on the genome level – only a single human pathogen genome and 14 non-human pathogen genomes were added to GenBank in the time window of our temporal benchmark. Therefore, the presented results are not fully representative, and the higher performance of the ResNet on read pairs cannot be guaranteed in general. However, the overall results are similar to those obtained for our main held-out test dataset. BAcc. – balanced accuracy (equivalent to accuracy on read sets), Prec. – precision, Rec. – recall, Spec. – specificity, AUC – area under the ROC curve, AUPR – area under the PR curve, Pred. – prediction rate.

		BAcc.	Prec.	Rec.	Spec.	AUC	AUPR	Pred.
Single reads	ResNet	68.9	69.6	66.9	**70.8**	74.9	74.6	**100.0**
	BLAST	**71.2**	**87.4**	**81.3**	61.2	-	-	82.6
Read pairs	ResNet	**74.2**	74.6	73.4	**75.1**	81.8	82.1	**100.0**
	BLAST	71.9	**87.4**	**77.5**	66.3	-	-	80.1

Overall, prediction accuracy for reads and read pairs is suboptimal for both BLAST and the ResNet, probably reflecting the extreme difficulty of the task (Table 5.2). The error estimates based on the held-out test dataset are consistent with the results of the temporal benchmark Table 5.4. The difference in performance for the first and second mate of the read pairs is negligible; we present the mean values. Interestingly, while BLAST's accuracy on the test set is higher, its accuracy on the validation set is lower by a similar margin (61.6%, compared to 63.3% for the ResNet). While this comparison should not be overinterpreted, as the ResNet epoch maximizing validation accuracy was chosen, those results suggest that the performance differences between the two approaches are small and could be dependent on a particular composition of species in the test set. Although this could be explicitly tested using a nested cross-validation scheme, such a setup would be computationally prohibitive.

We do not observe much overfitting (the single read training accuracy for the selected epoch is 68.9%), which suggests that the unsatisfactory performance is a form of underfitting. However, we speculate that even those predictions could be useful in some circumstances. The high precision of BLAST proves that its positive predictions are trustworthy. On the other hand, the ResNet offers only slightly worse accuracy, but for all reads in the sample (BLAST finds no matches for over 20% of reads even if read pairs are used). What is more, the ResNet can process 1.25 million reads in just 3 minutes on a single GPU, while BLAST needs over three hours and 256 threads on a high-performance computing node for the same task. The ResNet is slower when used on CPUs, but still outperforms BLAST more than sevenfold. This is very important in practice, as screening of large NGS datasets must be performed quickly to remain feasible. For this reason, BLAST is usually too slow for NGS analysis. Note that in this work, it represents the upper bound on accuracy of homology-based approaches; faster alternatives like NGS mappers or k-mer approaches have been shown to underperform in the pathogenic potential tasks for other pathogen groups (Bartoszewicz, Seidel, & Renard, 2021a; Deneke et al., 2017).

Both methods perform much better on full genomes, suggesting that the main performance bottleneck is the total amount of information (total sequence length) available as input. This is consistent with previous observations (Bartoszewicz, Seidel, & Renard, 2021a; Bartoszewicz et al., 2020; Deneke et al., 2017), although more drastic than in the case of bacteria or viruses. Interestingly, BLAST's precision is actually a bit lower for full genomes than for reads, while the ResNet becomes more precise when more information (more reads) is considered. Contrary to what was reported for viruses, using raw, unassembled reads improves BLAST's performance compared to relying on assembled contigs. A probable reason is that when a majority vote over all contigs is

taken, all contigs have the same voting weight, which means that the ratio of voting weight to sequence length can be much higher for short contigs of potentially lower quality than for the very long ones. In contrast, when reads are used, all regions of the genome weight the same. Using contigs also markedly increases the processing time. This slowdown is probably best explained by the computational difficulty of the local alignment of long contigs compared to a low-coverage set of representative reads. All approaches correctly classify the single human pathogen in the temporal benchmark dataset, but the ResNet achieves higher specificity (92.9) than read-based (78.6) and contig-based BLAST (85.7).

Despite the low coverage, the test set seems to be indeed representative. The genome-wide predictions are equally accurate if all test reads are used, and when only a half of the dataset (corresponding to either the first or the second mate) is analyzed. Therefore, computations can be sped up by considering first mates only, as they are enough to deliver an accurate prediction. Strikingly, this is the case even though they correspond to a mean coverage below 0.08. As a result, the read-based ResNet yields only slightly less accurate predictions than contig-based BLAST, but requires 700-fold less time if a GPU is used.

Both BLAST and the ResNet correctly predict *C. auris* and *A. fumigatus* to be pathogens, even though they were kept in the held-out test dataset unused during training. *P. oryzae* is also recognized as a member of the negative class. The ResNet (but not BLAST) incorrectly classifies *B. dendrobatidis*, a deadly amphibian pathogen, as able to infect humans. This prediction could be in principle indicative of a hidden zoonotic potential that could be realized under some yet unreported circumstances (e.g. in susceptible individuals or tissues). However, without further evidence, we assume this is simply an error of the model – perfect generalization is rarely possible, even though, as noted in Table 5.2, the prediction performance of the ResNet on full genomes is still high. It is possible that the erroneous prediction for *B. dendrobatidis* is caused by the underrepresentation of related species in the training set, as the core database contains only two other species of the same order, *Chytridiomycota*, hindering the classifier from correctly modelling their pathogenic potential.

5.4.2 Read-based genome representations

As shown in Figure A.12a and Figure A.12b, the values of $s(E[z])$ and $E[s(z)]$ are indeed similar for both the validation and test set. They are highly correlated (Spearman's $\rho > 0.99$, $p < 10^{-15}$) and linearily dependent. This justifies using $E[\mathbf{h}]$ as our genome representations. However, for the 'novel viruses' dataset, the relationship actually

resembles a sigmoid function (Figure A.12c). To investigate the relationship between the two versions of read averaging, we performed further experiments using simulated logit values z. In the binary classification case, if σ_S is low, the relationship between 'logit'- and 'proba'-averages is linear (Figure A.13a), and if σ_S is high, the relationship is sigmoid (Figure A.13b). We suggest an intuitive explanation of this effect. High within-species variance leads to many large absolute values of z within a given species. Note that extreme outliers disproportionately influence the mean, or $E[z]$. The sigmoid function applied *before* averaging 'squashes' those extreme values, diminishing their influence. This explains why $s(E[z])$ values tend to be more extreme than $E[s(z)]$). The effect is stable even if no separable classes are present in the dataset. Strikingly, we can easily reproduce the linear (Figure A.13c) and sigmoid (Figure A.13d) behaviour just by manipulating the within-species variance.

What is more, the effect requires a species-level signal to be present, i.e. each species must have its own mean and standard deviation assigned. If the distributions of z are the same for each species, the differences in $E[s(z)]$ are expected to be minimal. However, if the variance of z is extremely high, sampling effects can cause large dispersion of $s(E[z])$. As shown in Figure A.13e, 'logit'-averaging indeed causes spurious, high-confidence predictions in this setting (since $\mu_r = 0$, correct predictions should be close to 0.5). Those are most probably artefacts caused by the extreme values of some z outliers. 'Proba'-averaging is more robust in this case and correctly yields low confidence predictions close to 0.5. Note that this can also be understood as the 'logit'-averaging amplifying noise generated by sampling effects. Therefore, the problem can be at least partially mitigated with higher read numbers per genome – if we increase the number of 'reads' per 'species', we see the expected decrease in effect strength (Figure A.13f). In general, higher sensitivity of 'logit'-averaging could also be useful in some applications, depending on the levels of different sources of variance in the dataset.

Note that even in the settings with high within-species variance, where the relationship between $s(E[z])$ and $E[s(z)]$ becomes sigmoid, its approximate monotonicity is maintained (Spearman's $\rho > 0.98$, $p < 10^{-15}$ for both Figure A.13b and Figure A.13d). This suggests that even in this case, $E[\mathbf{h}]$ could potentially be used as genome representations. However, one must exercise caution – the the steeper the sigmoid relationship, the more distorted the distances between such representations would become. In our fungal dataset, the within-species variance is seemingly low enough to guarantee a linear relationship, so this problem does not affect it.

5.4.3 The landscape of fungal pathogenicity

Good overall accuracy of the ResNet is reflected in the visualization of learned genome representations for the entirety of the core database. Figures 5.1-5.3 present UMAP embeddings of the extracted representations for all labelled genomes, i.e. a sum of the training, validation and test datasets. Although some noise is present, the positive and the negative class are mostly separated. As shown in Figure 5.1, several clusters of human pathogens and non-human pathogens are present. The ResNet correctly recovers most of the labels, including many of the 'positive' members of the otherwise 'negative' clusters (Figure 5.2). Classification errors seem to originate from an interpolation based on neighbouring data points – within clusters, the predicted labels are more homogeneous than the ground truth annotations. This is expected, as the clusters represent similarity in the space of learned representations. The network should in general assign similar labels to inputs similar in this space. In contrast, BLAST works analogously to a *k*-nearest neighbours classifier in the input sequence space (finding the single closest match for each query). The ResNet, interpolating between multiple data points, may be less efficient in modelling situations where a small set of 'negative' data points is embedded within a larger 'positive' cluster of similar species, or vice versa. This hypothesis is supported by the visualization of BLAST-predicted labels in the learned representation space (Figure A.14). BLAST recovers mixed, contrasting labels within clusters more accurately, and its errors seem to be more evenly distributed across the space. At the same time, its slightly lower sensitivity is especially visible within the diverse *Sordariomycetes* class placed in the right-most cluster.

The clusters themselves are noticeably related to the taxonomic units represented in the database (Figure 5.3). The number of units at most levels of taxonomy poses a challenge for intuitive visualization of the dataset structure. Nevertheless, some correspondence between cluster membership, label and the taxonomic rank of a class is clearly visible (note that the 'class' as a taxonomic term is distinct from the concept of a positive or negative class in machine learning). This extends also the the lower and higher taxonomic ranks of order and phylum, respectively (Figure A.15-Figure A.16). On the other hand, this is importantly not a simple one-to-one mapping. Some orders of one class may belong to different clusters. For example, the orders *Chaetothyriales*, *Eurotiales* (including the genus *Aspergillus*) and *Onygenales* all belong to the class *Eurotiomycetes* (Figure 5.3, in red), but are members of four different clusters, with *Eurotiales* split into two (Figure A.15). Even more strikingly, the left-most cluster contains at least one member of all six phyla, with two most noticeable groups corresponding to the orders *Saccharomycetales* and *Mucorales* in phyla *Ascomycota* and *Mucormycota*, respectively

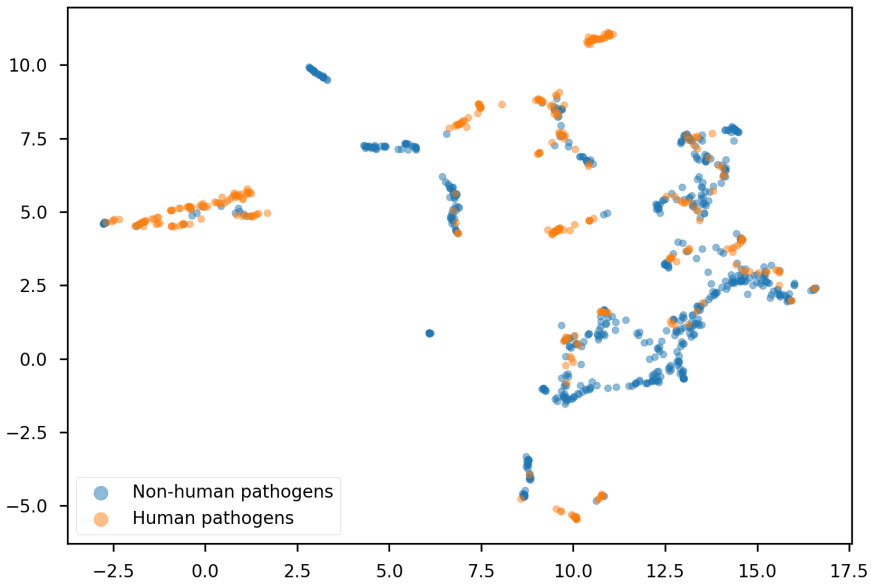


Figure 5.1: UMAP embeddings of the learned genome representations for the core database; true labels. Each point represents a genome of a single species, coloured by its ground-truth label. The learned representations offer a way of visualizing the core database along the relevant labels for each genome.

(Figures 5.3, A.15, A.16). 89% of the members of both those orders are at least opportunistic human pathogens; this includes *Candida* species in *Saccharomycetales* and causative agents of mucormycosis, also known as the 'black fungus', in *Mucorales*. Therefore, we hypothesize that the grouping represents a cluster of 'conserved' potential pathogens, relatively easy to classify with an acceptable accuracy. A common signal at the functional level or technical artefacts could be alternative explanations; they are however less likely due to the large phylogenetic distance on one side, and a consistent grouping of almost all species from those well-represented orders on the other.

5.4.4 Multi-class models

For the final evaluation of our database, we focused on a generalized version of the pathogenic potential prediction task. Here, we aimed to develop a model capable of classifying NGS reads originating from novel viruses, bacterial and fungal species into appropriate pathogen and non-pathogen classes. We trained the multi-class ResNets on data including four classes (human-pathogenic fungi, bacterial pathogens, human

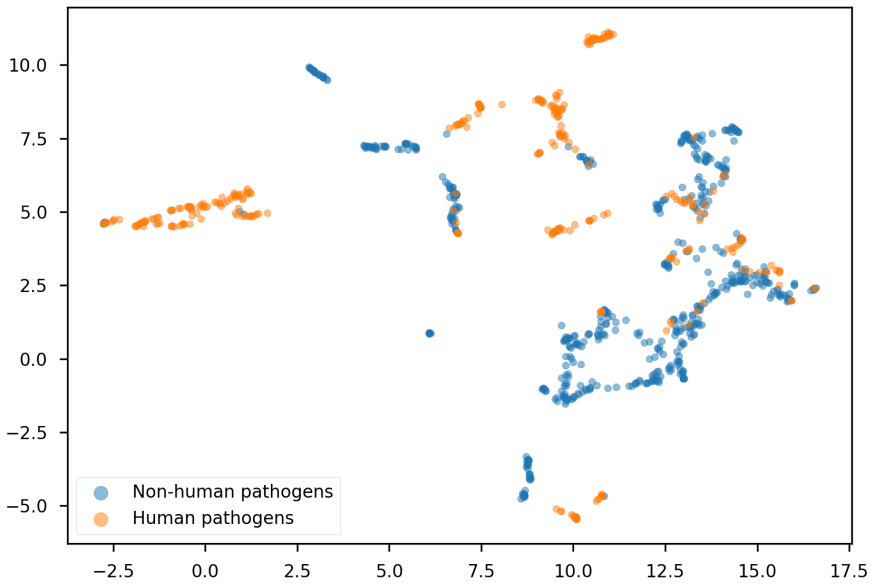


Figure 5.2: UMAP embeddings of the learned genome representations for the core database; labels predicted by the ResNet (retuned threshold). Each point represents a genome of a single species, coloured by its predicted label. The ResNet correctly classifies most of the genomes. Its predictions within clusters are more homogeneous than the ground truth, but many 'positive' members of otherwise 'negative' clusters are detected.

viruses and non-pathogens) as described in section Multi-class evaluation. The network trained on the dataset containing the 'logarithmic-size' version of the fungal positive class achieved slightly better validation accuracy and was selected for further evaluation, but the difference was small (<0.5%). We then evaluated a simple ensemble of both 4-class ResNets.

First, we used the DeePaC datasets consisting of bacteria and viruses to compare the 4-class models to a classifier including the three non-fungal classes only, as well as the original binary ResNets (Bartoszewicz, Genske, et al., 2021a) and BLAST. This procedure allows us to a) measure the effect of integrating the fungal dataset with the bacterial and viral data in one task, and b) disentangle the effects of adding the fungal data from the effects of merging the bacterial and viral datasets. We expected the fungal sequences to be relatively easy to differentiate from the others, but whether the ResNet architecture would be expressive enough to accurately represent all those diverse sequences was unclear.

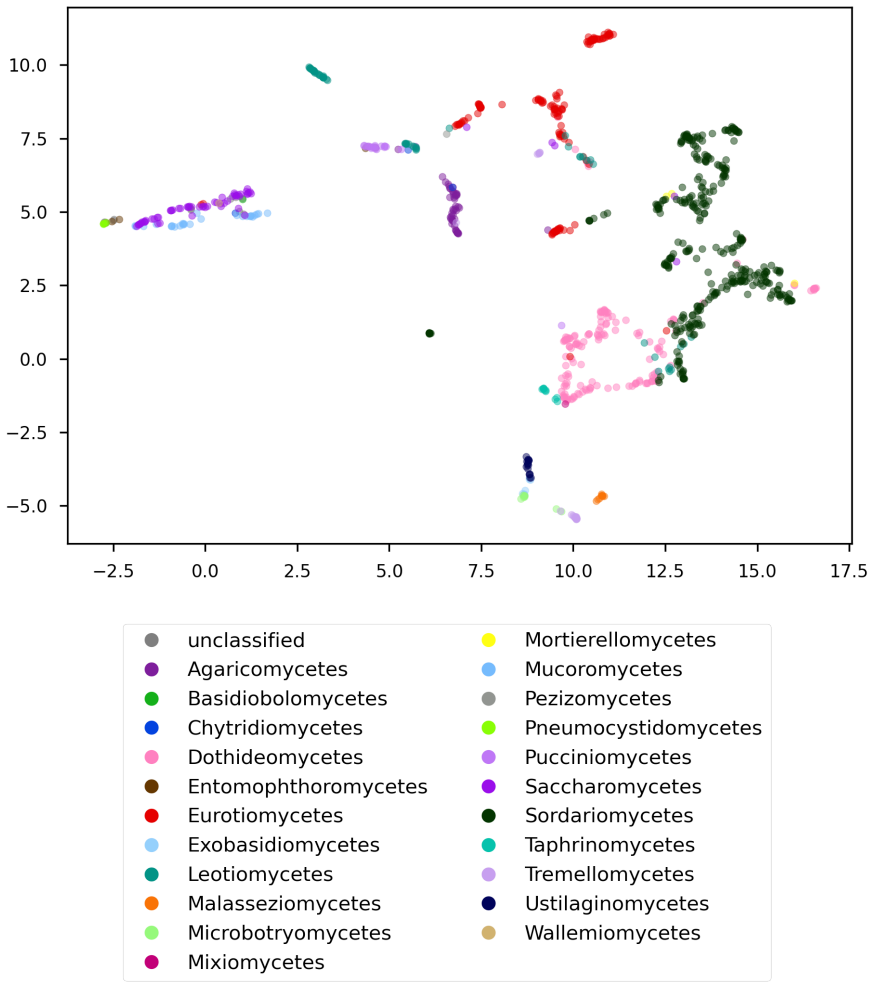


Figure 5.3: UMAP embeddings of the learned genome representations for the core database; taxonomic rank: class. The clusters are related, but not fully reducible to the taxonomic classification of the analyzed species. In general, related species are close to each other in this space, but some taxonomic units are distributed among more than one cluster (e.g. *Eurotiomycetes*, in red), and some clusters contain members of distant taxa (e.g. the left-most cluster).

Table 5.5: Performance of the multi-class classifiers on non-fungal datasets, read pairs. The 4-class classifier has been trained to detect non-pathogenic bacteria and non-human viruses, bacterial pathogens, human-infecting viruses, and human-pathogenic fungi. The 2-class classifiers are the original DeePaC ResNets (Bartoszewicz, Genske, et al., 2021a) for bacteria and viruses accordingly. We use BLAST as shown in Bartoszewicz, Seidel, and Renard (2021a) and Bartoszewicz et al. (2020); since this only returns binary labels, AUPR is not defined. The multi-class models do not assume a purely bacterial or viral sample, so they are more flexible than the 2-class networks – a single classifier can be used for both datasets without retraining, although a minor cost in performance should be expected of a more general approach. For the viral dataset, assignments to the bacterial pathogens class may be assumed to actually reflect bacteriophages infecting those bacteria due to high bacteriophage-host sequence similarity, and reassigned accordingly ('reass.'). The multi-class models and the specialized binary classifiers perform comparably well, even though the latter can only be used in very specific situations. BLAST, using an appropriate reference database and representing the estimated upper bound on performance of homology-based approaches, is outperformed by a significant margin. Acc. – accuracy, Prec. – precision; Rec. – recall, Spec. – specificity, AUPR – area under the PR curve (calculated for the respective positive class).

		Acc.	Prec.	Rec.	Spec.	AUPR
Bacteria	4-class ensemble (ours)	85.1	**84.1**	87.2	**83.0**	88.3
	2-class	**87.3**	83.6	**92.7**	81.8	**89.0**
	BLAST	70.0	**84.1**	86.6	53.5	-
Viruses	4-class ensemble, reass. (ours)	89.6	95.3	**89.3**	90.0	**98.1**
	2-class	**90.3**	94.7	85.5	**95.2**	97.2
	BLAST	80.6	**98.4**	79.1	82.2	-

As shown in Table A.9, integrating the fungal dataset with three bacterial and viral classes indeed does not negatively influence the prediction accuracy. The main challenge in multi-class pathogenic potential prediction seems to be actually differentiating between bacteriophages and the pathogenic bacteria that they infect. The multi-class networks confuse 18–21% of viral non-pathogen reads for pathogenic bacteria. However, their accuracy is still similar to BLAST's, even though BLAST was used with a purely viral reference database, and hence would always classify ambiguous bacterial or viral sequences as viral. This problem is unrelated to the fungal database presented here, and can be circumvented by assuming that spurious assignments to bacteria should be reassigned to bacteriophages in the case of a purely viral dataset (see section Multi-class evaluation). The resulting networks achieve accuracy similar to that of the simple binary classifiers, while still being capable of multi-class predictions for more complex datasets. They also outperform BLAST, which represents the estimated upper limit on the accuracy of pathogen detection by finding the closest taxonomic matches, by a wide margin (Table 5.5).

Table 5.6: Performance on the multi-class dataset, read pairs. The 4-class classifier includes the fungi class along the three viral and bacterial classes included in the 3-class classifier. The best performance for each class is marked in bold. In this setting, the true positive rate corresponds to the rate of correct assignments within a given class. Hence, recall is equal to accuracy for each class. We use the F1 score as an additional measure. As expected, BLAST's predictions are very precise, since when it finds a match, it is usually a relevant one. This does not hold for the non-pathogen class, which could indicate confusion between bacteriophage and bacterial pathogen reads. Our classifier significantly outperforms BLAST in terms of recall and prediction accuracy for all classes. BLAST, representing the estimated upper bound on performance of homology-based approaches, yields no predictions for 12.5% of all read pairs. Acc. – accuracy, F1 – F1 score, Prec. – precision, Rec. – recall, AUPR – area under the precision-recall curve.

		Acc.	F1	Prec.	Rec.	AUPR
All classes	4-class ens. (ours)	**87.6**	**87.7**	87.7	**87.6**	93.4
	BLAST	78.3	84.0	**90.6**	78.3	-
Non-pathogens	4-class ens. (ours)	**77.4**	**78.7**	**80.1**	**77.4**	86.7
	BLAST	66.5	71.6	77.5	66.5	-
Path. bacteria	4-class ens. (ours)	**87.2**	85.1	83.2	**87.2**	90.4
	BLAST	83.8	**87.5**	**91.6**	83.8	-
Human viruses	4-class ens. (ours)	**90.9**	**93.7**	96.7	**90.9**	98.4
	BLAST	78.9	87.9	**99.2**	78.9	-
Fungi	4-class ens. (ours)	**95.0**	**92.9**	90.9	**95.0**	97.9
	BLAST	84.1	88.9	**94.2**	84.1	-

The fungal dataset can be integrated with the other classes without causing any significant performance hits on the full, multi-class dataset as well (Table A.10). Consistently with the other results, performance is lower on the non-pathogen class, since many bacteriophage reads can be confused with pathogenic bacteria. While this issue requires further research, we expect future solutions to remain compatible with our database. The 4-class ensemble achieves the most balanced performance on non-pathogen data, the best recall on fungal reads and is also the most accurate overall, cutting the average error rate by over 40% compared to BLAST (Table 5.6). The results show that using the data collected in our database, we can efficiently extend previously developed pathogenic potential methods to be compatible with a metagenomic setting including fungi, without assuming a purely bacterial or viral sample as it was done previously (Bartoszewicz, Genske, et al., 2021a).

5.5 Discussion

5.5.1 Database of pathogenic fungi and their hosts

Fungal pathogens have been under-studied compared to human-infecting bacteria and viruses, leading to repeated calls for more research in this area (Huseyin et al., 2017; "Stop neglecting fungi", 2017). What is more, a large part of the research effort has been focused on plant pathogens due to their agricultural significance. A subset of them could in principle also have an unreported or undetected ability to infect a human host. An analogous problem also applies to incomplete data regarding pathogenicity towards non-human animals or plants. For this reason, we do not claim that species not listed as potential pathogens are indeed non-pathogens. In our database, we include confirmed labels alongside appropriate sources; in case of lack of evidence, we treat the respective label as missing. It is therefore possible that some of the fungi currently labelled as 'non-human pathogens' would have to be reclassified as the state of science evolves. This may be especially important as it has been suggested that the ongoing climate change will lead to more frequent host-switching events, including expansion of host range to mammals, which are usually relatively resistant to fungal infections (Garcia-Solache & Casadevall, 2010). Even though the very goal of the presented classifiers is to generalize to newly emerging species, large, comprehensive datasets are crucial – often more important than the actual analysis method used. This has been shown before for metagenomic data (Piro et al., 2020) and likely applies to the tasks analyzed here as well. Therefore, extending the database to include more species, as more genomes are sequenced in the future, could facilitate the downstream tasks. To support future extensions, we include all considered species in the database – even those without assigned TaxIDs, genomes, or labels (in the case of screened GenBank genomes). This broadens the scope of the data from 1,455 labelled genomes to over 14,500 records, enabling easy labelling of newly published genomes and minimizing the workload needed for the addition of new, non-redundant records. It is also possible to link the species TaxIDs to taxa below the species level. However, it should be kept in mind that the fungal taxonomy is in constant flux – taxa previously considered variants of a single species may be reclassified into separate species in the future. While automatically curated databases like EID2 (Wardeh et al., 2015) are relatively easy to update and scale, we note that they may be prone to errors introduced by the automated protocol used. Manual curation is not fully error-free either, but we see it as a necessary step to maximize the quality of the collected labels. Both approaches are complementary and may be best suited for different use-cases.

5.5.2 Application to pathogenic potential prediction

We show that both BLAST and ResNet can accurately predict if a fungus is a human pathogen based on its genome. However, the task remains inherently difficult – a DNA sequence by itself, outside of the context of the host, can be probabilistically associated with a certain trait, but the trait itself is only realized in the biological system comprising of both the host and the colonizing microbe as a whole. Therefore, we intentionally include opportunistic pathogens in the 'positive' class of human pathogens. These considerations mirror challenges previously described for other pathogen classes (Bartoszewicz, Genske, et al., 2021a; Bartoszewicz, Seidel, & Renard, 2021a; Bartoszewicz et al., 2020; Deneke et al., 2017).

Given the limited information content of a short NGS read, predicting if a fungus can infect humans directly (and only) from reads is therefore an extremely challenging task. The read-level performance is admittedly low for predicting a fungal host. More expressive neural architectures could possibly help alleviate this problem, but it is also possible that prediction accuracy known for other pathogen classes is simply impossible for fungi, for example due to the higher complexity of eukaryotic genomes with abundant regions of non-coding DNA. On the other hand, multi-class networks detecting fungal, bacterial or viral pathogens yield satisfactory accuracy under some simplifying assumptions. We envision that the networks trained using the database can be applied to predict pathogenic potentials of short metagenomic reads, as previously suggested for purely bacterial and viral datasets (Bartoszewicz, Genske, et al., 2021a). Although we only evaluate them on simulated reads, this has been shown to approximate performance on real datasets very well (Bartoszewicz, Genske, et al., 2021a; Bartoszewicz, Seidel, & Renard, 2021a; Bartoszewicz et al., 2020). Further work could extend the presented multi-class setup to Nanopore reads, as shown for bacterial and viral models (Bartoszewicz, Genske, et al., 2021a), enabling selective sequencing of mixed-pathogen samples.

5.5.3 Data visualization and genome representations

We show that full genomes can be represented by aggregating representations of reads originating from each genome. In addition to that, we observe that coverage as low as 0.08 is enough to correctly classify a species. Taken together, those two facts warrant a view of a species genome as a distribution generating subsequences (i.e. reads) originating from it; such a distribution can also be considered in an abstract representation space. This concept is very similar to that of a k-mer spectrum, where an *empirical* distribution of k-mers is used as a signature of a longer sequence to enable alignment-free comparisons (Zielezinski et al., 2017), including being used as input features for machine

learning approaches as in Deneke et al. (2017). However, k-mer spectra operate in the sequence space only.

Classifiers based on aggregated *representations* are approximately equivalent to classifiers based on aggregated *predictions*, although this relation is modulated by the standard deviation of the respective, genome-specific distribution. A somewhat related effect was reported in the context of competing design choices for neural networks equivariant to DNA reverse-complementarity – models averaging the predictions for both DNA strands were found to be approximately equivalent to models applying a sigmoid transformation to an average of logits (H. Zhou et al., 2022). The probabilistic view of genome representations presented here deserves deeper investigation; this could potentially lead to the development of useful embeddings also for whole, multi-species samples.

Finally, the trained representations allowed us to visualize the taxonomic diversity of the database along its phenotypic landscape. As expected, the apparent fungal host-range signal seems to be related to, but not fully reducible to the fungal taxonomy. It can be captured by finding the closest taxonomic matches, as well as by a neural network, avoiding direct taxonomic assignment for a more assumption-free view of fungal pathogenicity. This suggests that both methods should generalize well to novel species sequenced in the future. At the same time, a stark dataset bias is clearly visible, with many more genomes available for some taxonomic units than for others in the same rank. Future sequencing projects focusing on underrepresented taxa could help build a more fine-grained view of the pathogenicity landscape presented here.

5.5.4 Reuse potential and implications

Although in this work, we focus on using the collected data in a pathogenic potential prediction task, the database can find future applications beyond this particular problem. For example, we imagine that fungal genomes could be scanned for regions associated with their ability to colonize and infect humans, as shown previously for bacteria and viruses (Bartoszewicz, Seidel, & Renard, 2021a). This approach uses sliding windows of the same size as the NGS reads used for training; high-scoring sequences tend to be clustered together and often colocalized with virulence-related genes. Even though our binary fungal host classifiers achieved relatively low per-read accuracy, we speculate that finding contiguous regions of elevated pathogenic potential could be a way to identify potentially relevant genes, also for novel and weakly annotated species. On the other hand, the multitude of genomic features present in fungal genomes, often including intron features and regions without obvious functional annotation, renders the validation of such an approach a challenging project on its own. This could be perhaps facilitated

by focusing exclusively on coding regions, which should in principle carry a stronger phenotype-related signal, at the risk of omitting potentially relevant, non-coding (e.g. regulatory) elements. Building such a gene-based classifier could be a future application of the presented dataset.

We imagine that, depending on annotation quality, genomes collected here could be a valuable resource for functional and comparative genomics of fungi. As a source of curated labels, it could also support applications of proteomics to fungal pathogen research. For example, computational metaproteomics and proteogenomics approaches enable analysis of microbial communities based on mass spectrometry data and can be co-opted for pathogen detection workflows independent of DNA sequencing (Renard et al., 2012; Schiebenhoefer et al., 2019; Schiebenhoefer et al., 2020).

5.5.5 Conclusions

We compiled a comprehensive database of fungal species linked to their host group (human, non-human animal or plant), evidence for their pathogenicity, and publicly available genomes. To showcase the potential uses of the dataset, we benchmark the two most promising approaches to fungal pathogenic potential prediction: a deep neural network capable of fast inference directly from fungal DNA sequences, and the gold standard in homology-based pathogen identification – BLAST. We also extended our approach to a multi-class scenario, integrating collected fungal data with bacterial and viral datasets. The database, hosted at https://zenodo.org/record/5846345, can be reused for future research on fungal pathogenicity based on both machine learning and other methods of computational biology. The models, read sets and code are available at https://zenodo.org/record/5711877, https://zenodo.org/record/5846397, and https://github.com/dacs-hpi/deepac.

6 Summary and Conclusions

6.1 Summary

The 20th century has brought many unprecedented achievements in the fight against infectious diseases, including the eradication of smallpox with successful public health and vaccination programmes (Fenner, 1982), or the discovery of penicillin (Fleming, 1929) and the antibiotic revolution that followed. However, new challenges arose as well. Many of them, such as the rise of antibiotic resistance (Levy & Marshall, 2004; Neu, 1992; Piddock, 2012; Ventola, 2015), increased human exposition to zoonotic agents (K. E. Jones et al., 2008), and possibly also pathogen emergence facilitated by climate change (Casadevall et al., 2019; Garcia-Solache & Casadevall, 2010; K. E. Jones et al., 2008; J. A. Patz et al., 2005; Vouga & Greub, 2016), have been at least indirectly exacerbated (if not directly caused) by human actions. This is difficult to avoid – the constant evolutionary arms race between the pathogens and their hosts leads to a cycle of adaptation and counter-adaptation, as suggested by the Red Queen hypothesis (Van Valen, 1973). In this view, the competition between species is assumed to be a zero-sum game. If the pathogen becomes more virulent, its fitness may increase at the expense of the host. The host then faces stronger selective pressure to develop improved defence mechanisms; if it becomes resistant, its fitness increases at the expense of the pathogen. Red Queen dynamics have been studied for host-pathogen relationships in particular (Brockhurst et al., 2014; Papkou et al., 2019), including the specific context of mammalian hosts and SARS-related coronaviruses both before and after the discovery of SARS-CoV-2 (H. Guo et al., 2020; Sironi et al., 2015). Hence, more novel pathogens should be expected to emerge, especially given the high pace of viral and microbial evolution (K. E. Jones et al., 2008; Trappe et al., 2016; Vouga & Greub, 2016; M. E. J. Woolhouse & Gowtage-Sequeria, 2005; M. E. J. Woolhouse et al., 2005), and the human actions facilitating host-switching events (K. E. Jones et al., 2008). What is more, many potential pathogens remain undiscovered – over 100,000 new RNA viruses have been recently identified by mining already available data, which is still several orders of magnitude less than the estimated number of distinct viruses on Earth (Edgar et al., 2022). Some of the previously unknown agents may cause deadly outbreaks or possibly

pandemics. The risk that a new pathogen would be accidentally or deliberately produced by human research activities must be accounted for as well, as it could escape the confined laboratory environment or even be used as a bioweapon by a malicious actor. DNA and RNA sequencing are the state-of-the-art techniques for open-view pathogen detection and, as such, are suited particularly well for identification of novel threats. However, the computational analysis of obtained data may be challenging, especially if the relevant genome is not present in the reference database. In this thesis, I presented a collection of deep learning methods predicting pathogenic or infectious potentials associated with unknown agents directly from DNA or RNA sequences, without the need of finding the best taxonomic match.

To this end, I first introduced DeePaC, a python package enabling training and evaluation of various deep neural networks for DNA inputs, especially large datasets of millions of short, read-like sequences. It supports simplified design of multiple architectures, including CNNs and LSTMs as presented in chapter 2, ResNets, ResNeXts (S. Xie et al., 2017), as well as the encoder parts of Transformers (Vaswani et al., 2017) and Performers (Choromanski et al., 2021) for supervised tasks. The command-line interface and the supplied default configurations make the tool accessible for all users without the need to reimplement the code or redesign the architectures. Model definition relies on human-readable config files that facilitate performing hyperparameter sweeps, benchmarking various design choices on multiple machines (possibly in different network environments) and allow defining models invariant to DNA reverse-complementarity with just a simple parameter change, setting DeePaC apart from other deep learning libraries for biological sequence inputs developed independently in parallel (Budach & Marsico, 2018; K. M. Chen et al., 2019; Kopp et al., 2020). In chapter 2, I showed that those architectures indeed outperform the previous state-of-the-art in bacterial pathogenic potential prediction using machine learning (Deneke et al., 2017), as well as homology detection based on matching the input reads to the closest possible reference genomes. What is more, predictions can be easily aggregated across individual mates in a read pair for a reliable boost in prediction accuracy.

However, since novel biological threats can originate from many different pathogen groups, classifiers focused exclusively on bacteria are not enough. Therefore, I used the streamlined model development capabilities of DeePaC to train new networks differentiating between reads originating from human-infecting viruses and reads of non-human viruses. Even though an approach originally designed for bacteria was not guaranteed to work for viruses due to their very different biology, the trained models have indeed proven to outperform all alternatives, including sequence alignment and a kNN classifier independently developed by (Z. Zhang et al., 2019). In contrast to the bacterial dataset,

where CNNs and LSTMs worked comparatively well, CNN was the preferred architecture. Initially suboptimal results for full genomes of novel viral species (as opposed to novel viruses of previously known species, such as e.g. SARS-CoV-2) presented in chapter 3 can be improved by tuning the decision threshold, as shown in chapter 5. At this stage of the work, DeePaC was also extended with a suite of interpretability tools facilitating model debugging and visualizing relevant patterns in input sequences. This enabled a brief analysis of the SARS-CoV-2 virus once it emerged after the models had already been developed. However, benchmarking of interpretability approaches is notoriously difficult, especially since the ground truth is often not available, as in the case of the pathogenic or infectious potential prediction tasks. In the future, the preliminary evaluation of the suite presented here should be scaled-up beyond a small set of selected examples.

As novel pathogen detection is a time-critical application, I focused on predictions for unassembled reads, which can be then extended to contigs or whole genomes. This avoids the potentially challenging assembly step, but in the case of standard Illumina protocols, still requires waiting for the preprocessed reads, which can usually be accessed only after the sequencing run has finished. Previously published tools can perform read mapping (Lindner et al., 2017; Loka et al., 2019), host read filtering (Loka et al., 2018), taxonomic classification (Tausch, Strauch, et al., 2018), and detection of known pathogens (Tausch, Loka, et al., 2018) in real time, as the sequencer is running. However, incomplete read fragments accessible in intermediate sequencing cycles (called *subreads* in this thesis) are up to an order of magnitude shorter than the combined length of mates in a finished read pair. This makes the problem of real-time pathogenic potential prediction much more challenging than its full-read counterpart. To maximize prediction accuracy for the shortest subreads without sacrificing performance on finished reads, I introduced a new data setup and a version of the ResNet architecture invariant to reverse-complementarity. The improved models detect subreads probably originating from bacterial or viral pathogens, which can be used for downstream, in-depth analysis. Although the neural networks outperform sequence alignment by a wide margin also in this context, I propose a more complex pipeline, where the reads most similar to known references are aligned using a specialized read mapper (Loka et al., 2019), unmapped reads are filtered using a previously trained ResNet, and putative pathogen reads are re-analyzed with BLAST. As a result, both known and novel pathogens can be efficiently detected and assigned to their closest taxonomic matches. BLAST re-analysis can also be viewed as an additional level of interpretability, adding an explicit connection between the neural network's predictions and available reference genomes. Alternative ResNets trained on simulated Nanopore data enable predictions for 250bp subreads,

corresponding to about 0.5 s sequencing time. In the future, they could be integrated with tools like ReadBouncer (Ulrich et al., 2022), a recent, more efficient alternative to the original Read Until framework for selective sequencing (Loose et al., 2016). This will enable filtering out reads originating from non-pathogens and focusing on putative pathogen reads. Further, although potential applications for screening of synthetic sequences were already suggested in chapters 2 and 3, the subread-trained ResNets presented in chapter 4 achieve stable accuracy also for sequences below 200bp, making oligonucleotide screening feasible.

On the other hand, those models are still binary classifiers, assuming that the analyzed sample consists of either only bacterial or viral reads. This is not realistic in the clinical or metagenomic context. Distinguishing viral from bacterial reads with one of the available methods (e.g. Auslander et al. (2020), Fang et al. (2019), Miao et al. (2021), Ren et al. (2020), and Tampuu et al. (2019)) could be the first step of a multi-model pipeline, but the development of a single, multi-class network enables detecting potential threats in a single, computationally efficient step. Further, it allows extending the predictive capabilities of the classifiers to a third major pathogen group – fungi. This is necessary, as determining the group a putative novel pathogen belongs to from symptoms alone can be extremely challenging. The first step – compiling the relevant training and testing dataset – was more difficult for fungi than for bacteria and viruses. Available data was scarce and often not machine-readable. Therefore, a new database of fungal pathogens, their host groups and reference genomes had to be manually curated, as described in chapter 5. The model trained to differentiate novel human-pathogenic fungi from plant, and non-human animal pathogens is much faster than BLAST and works similarly well; both methods are highly accurate on whole genomes but underperform on isolated reads. A possible reason is that some specific aspects of fungal biology (perhaps shared with other eukaryotes, e.g. long stretches on non-coding DNA) make inference from reads too difficult. On the other hand, future advancements could improve read-level predictions to a more satisfactory level, just as DeePaC models outperformed previous machine learning approaches on bacterial and viral data. Importantly, in patient-derived samples, only human-colonizing fungi can be expected to be present. Relying on this assumption, I present classifiers detecting human-infecting fungi, viruses and bacteria, and differentiating them from non-human viruses (e.g. bacteriophages) and non-pathogenic, commensal bacteria. Finally, genome embeddings based on average per-read activations enable visualization of the dataset structure and offer additional insight into the inner workings of the fungal host range classifier. As expected, related genomes cluster together, but the clusters are not fully reducible to the underlying taxonomy and seem to also be affected by the ability to infect humans conserved in some of the taxa. The developed representations support

the view of a genome as a distribution of reads, which can be considered not only in the raw sequence space, but also in the latent space of trained, intermediate features (and presumably, any given space of results of some transformation of the input sequences). A similar perspective has been implicitly used already by PaPrBaG (Deneke et al., 2017) and the models described in chapters 2 and 3, where genome-level predictions were derived from average per-read outputs.

In summary, I presented a collection of related deep learning approaches for detection of novel bacterial, viral and fungal pathogens, either after or during sequencing, with a possible application also to fully synthetic sequences. Neural networks excel where traditional methods based on finding taxonomic matches often fail – when the relevant reference genome is missing, as in the case of encountering a novel biological threat. They scale well from read fragments as short as 50bp to full genomes, with accuracy consistently rising as more input information becomes available. In practice, a combination of standard tools based on *k*-mers or sequence alignment with the deep learning models introduced here (and an optional assembly step, if feasible) would most probably work best, offering insights derived from homology detection and taxonomic classification, while also detecting threats impossible to find with traditional methods.

6.2 Outlook

Although the main objectives of this work have been completed, several aspects remain that should be improved upon by future research. H. Zhou et al. (2022) have recently benchmarked a wide range of strategies for ensuring equivariance to reverse-complementarity in deep neural networks for regulatory genomics. They also describe a class of 'post-hoc' conjoined models – 'standard' models trained using a dataset augmented with reverse-complement sequences, for which the outputs are averaged across the forward and reverse-complement orientations of each input at inference time. Surprisingly, post-hoc conjoined CNNs seem to often perform slightly better than reverse-complement parameter sharing architectures (called *full* RC-CNNs in chapter 2), which may be due to optimization difficulties rather than overfitting or differences in representational capacity (H. Zhou et al., 2022). Training and evaluating post-hoc conjoined networks (not limited to CNNs) is readily possible using the DeePaC command-line interface, so future projects could easily test if the improvements seen for transcription factor binding site detection and profile prediction tasks apply to raw viral and microbial sequencing data as well.

DeePaC supports also more complex architecture design choices that have not been described in detail here, including ResNet bottlenecks (He et al., 2016), the cardinality

dimension of ResNeXts (S. Xie et al., 2017), and the attention modules of Transformers (Vaswani et al., 2017) and Performers (Choromanski et al., 2021). These have not been comprehensively benchmarked, as preliminary tests suggested that they offered no or only minor improvements in prediction accuracy, at the cost of substantially increased computation time. Dilated convolutions (Yu & Koltun, 2016) are also implemented, although their use in genomics is usually motivated by the need of extending the receptive fields of the trained CNNs (Avsec, Weilert, et al., 2021; Kelley, 2020), a consideration less crucial for relatively short inputs of maximum 250bp used here. Inclusion of some 'modernized' CNN design choices, similar to the modifications introduced in the ConvNeXt architecture (Liu et al., 2022) and the timm library (Wightman et al., 2021), could further improve accuracy. A wide neural architecture search, combined with an appropriate hyperparameter sweep and ideally a nested cross-validation scheme with multiple training runs per fold, would probably identify improved designs. However, this procedure would be very expensive, while the obtained performance boost could still be small.

Another promising avenue is to consider alternative representations of the input sequences, such as FCGR, summarized in a recent review by Löchel and Heider (2021). This would enable using sequences of arbitrary length as inputs, but since existing solutions use one vertex per nucleotide type (A, C, G, T), strategies for proper encoding of ambiguous nucleotides (N) should be investigated first. The potential of representing genomes (or metagenomes) as a distribution of reads in an abstract space should be explored further as well. Pre-trained sequence embeddings can be learned using language models, a class of methods originally developed for natural language processing tasks. This could entail a slight modification of either BERTax (Mock et al., 2021), originally trained on 1.5kb-long eukaryotic, bacterial, archaeal, and viral sequences, or Looking-Glass (Hoarfrost et al., 2020), which was developed for bacterial and archaeal reads. The behaviour of self-attention heads in Transformer models similar to BERTax can be visualized, which would add another dimension to the interpretability suite described in chapter 3. However, the limitations of the methods implemented in DeePaC would still apply.

More specifically, interpreting the predictions for each sample (i.e. read) is not feasible for the use-cases discussed here, focusing on rapid filtering of millions of sequences. To debug a classifier and assess if it is indeed trustworthy, one has to either aggregate relevance scores over hundreds of thousands of input sequences or analyze selected genomes of interest. What is more, the ground truth is largely missing. Rich genome annotations are only available for a subset of the most well-studied species. Even in those cases, only an indirect link between the nucleic acid sequence and the pathogenic

phenotype can be established (e.g. with a Gene Ontology term indicating that a given gene is engaged in virulence). This does not apply exclusively to deep neural networks – decisions of models such as random forests or kNN using k-mer frequencies as inputs are often difficult to explain, as the features themselves do not necessarily have any inherent biological meaning. Interpretability of any machine learning model must be viewed critically, and the risk of delivering misleading or even harmful explanations must be taken into account (see e.g. Lipton (2017) and Rudin (2019)). Alignment algorithms and taxonomic classifiers relying on exact matching of long k-mers are more interpretable (as they indicate easily measurable sequence similarity to the closest taxonomic match) but offer less accurate predictions. Combining homology detection with predictions of a neural network, as shown in chapter 4, helps to circumvent this problem. However, an in-depth, statistically sound investigation of the patterns driving the decisions of the presented ResNets would elucidate the still not fully clear relationship between the pathogenic potential scores and taxonomy. This will help establish more trust in the models, which could then be used to detect regions of interest in unannotated genomes.

Further, while DeePaC-Live can use the models trained here for novel pathogen detection during Illumina sequencing (see chapter 4), an analogous pipeline for ONT sequencing is missing. Integration of the Nanopore-trained ResNets with selective sequencing workflows similar to Read Until (Loose et al., 2016) or ReadBouncer (Ulrich et al., 2022) will require the development of additional software modules connecting the read mapping, pathogenic potential prediction and pore unblocking steps. Efficient real-time mapping has been shown for both base-called reads (H. S. Edwards et al., 2019; Payne et al., 2021) and raw, electrical signals (Kovaka et al., 2021); uncalled squiggles can also be used as input for deep learning models distinguishing human from bacterial DNA (Bao et al., 2021). Neural networks analogous to presented here could be trained to detect novel pathogens directly from Nanopore signals. On the other hand, a more modular design separating the base calling and prediction steps would facilitate building complex and flexible pipelines, where individual tools can be exchanged or tuned for a particular task. Models operating in sequence, rather than electrical signal space, could also be more robust to changes in the sample preparation and sequencing protocols. A benchmark of different design choices should also consider potential differences in detection accuracy between cDNA and direct RNA sequencing of RNA viruses. This is especially important since direct sequencing of even very long, full-length viral genomes has recently become possible, as shown for HCoV-229E, a human coronavirus (Viehweger et al., 2019). Perhaps in the future, DeePaC-like models will be able to use whole viral genomes as input, without the need for assembly. This will likely require further advancements in lowering the Nanopore sequencing error rates,

as well as redesigning the neural architectures to efficiently handle longer inputs, e.g. using dilated convolutions or Transformer blocks (Avsec, Agarwal, et al., 2021). As the nanopore-based, single-molecule protein sequencing technologies develop (Alfaro et al., 2021; Brinkerhoff et al., 2021), methods analogous to the ones presented here could possibly be used also for peptide reads.

One of the important advantages of Nanopore sequencing over the short-read alternatives is that ONT devices are portable enough to be deployed in mobile labs for outbreak surveillance. This has been demonstrated for viral, bacterial and fungal pathogens (Y. Wang et al., 2021), so the Nanopore-trained models presented in chapter 4 should be extended to a multi-class setting introduced in chapter 5. New models could use a wider range of input sequence lengths and ideally also consider other eukaryotic pathogens, formerly classified as '*Protozoa*'. Since protozoans constitute a smaller fraction of human pathogen species than the three major groups discussed in this work (K. E. Jones et al., 2008), preparing training, validation and test datasets will be likely even more challenging and labour-intensive than for fungi. However, this problem could be alleviated using transfer learning if some of the features learned in the context of other pathogen groups are relevant also for protozoan classification. Alternatively, a joint class of eukaryotic pathogens including both human-infecting fungi and *Protozoa* can be considered, as in a clinical sample, only parasitic species can be expected (see chapter 5). An analogous classifier for Illumina data will also be a useful extension of the already trained models.

This highlights the flexibility of deep learning approaches for biological sequences in general and the DeePaC package specifically – many different host-pathogen and sequence-phenotype relationships can be modelled. Prediction of antimicrobial resistance, or at least selective sequencing of resistance-related genes, could be a future application. However, only a minority of reads can be expected to actually convey any relevant information, rendering the problem quite challenging unless assembled contigs or long reads are considered. Further, specialized classifiers trained on selected viral species (Q. Guo et al., 2021; Q. Guo et al., 2020; Mock et al., 2020) offer more precise predictions for the taxon they were trained on, and could possibly even yield mechanistic insights into the biology of the virus in question. Abstract phenotypes can also be predicted beyond the context of the human host. Accurately identifying bacteriophage hosts (M. Li et al., 2020; Tan et al., 2021) and whether they are virulent or temperate (S. Wu, Fang, et al., 2021) would facilitate the development of phage therapeutics. Non-medical applications may be impactful as well. Building simple models predicting mycorrhizal relationships between plants and fungi has been suggested by Lilleskov and Parrent (2007) – with DeePaC, classifiers distinguishing plant symbionts from pathogens from

sequence alone can be evaluated. The same applies to plant growth promoting bacteria, as specialized databases are available and phenotype prediction using BLAST and hidden Markov models has been proposed (S. Patz et al., 2021). Interpretability tools may assist with identifying genes most relevant for the phenotype of interest.

In the long-term perspective, deep neural networks could be used to optimize the target phenotypes and suggest edits or candidate sequences for wet-lab validation. Recognizing parts of the input that are crucial for the final prediction enables sequence design and optimization. This can be done using human insights by combining predictive models with one of the interpretability methods (as suggested for example by Zhao et al. (2021) in the context of plant regulatory genomics). Alternatively, a predictor can be used as an oracle supplying values of an objective function for sequences produced by a generative model (Brookes et al., 2019; Gupta & Kundaje, 2019; Gupta & Zou, 2019; Hawkins-Hooker et al., 2021; Linder et al., 2019, 2020; Riesselman et al., 2018; Schreiber et al., 2020; Shin et al., 2021; Sinai et al., 2018), so that the sequences can be optimized automatically. The latter class of approaches has grown popular in the protein design field, although alternatives based on Bayesian optimization, evolutionary algorithms and reinforcement learning also exist (Sinai & Kelsic, 2020; Sinai et al., 2020; Z. Wu et al., 2021; K. K. Yang et al., 2019). Computer-aided design of whole genomes using deep learning is not currently possible; the search space of genome-length sequences is immense. What is more, models capable of this would have to represent a whole distribution of complex biological systems (that the genomes correspond to) well enough to both predict the phenotype *and* suggest novel, biologically feasible sequences. On the other hand, the work presented here suggests that capturing complex phenotypes (at least in the predictive context) is indeed possible to some extent. If learning-based optimization of whole genomes becomes achievable in the future, deep 'phenotype models' conceptually similar to the ones presented here could facilitate design and engineering of new therapeutic phages, plant growth-promoting bacteria, perhaps even attenuated viruses for vaccine research or eukaryotic genomes. As language models can predict viral mutational escape at the protein level (Hie et al., 2021), genome-level phenotype models could improve preparedness for virus variants that have not been observed yet. However, this would also be a classic example of a dual-use technology – a powerful tool that could save or improve millions of lives, but also a potential threat in the hands of a malicious actor. The models discussed in this thesis do not have such capabilities, but the balance of benefits and risks posed by synthetic biology, especially in the context of its convergence with artificial intelligence research, should be continuously monitored and evaluated. Relevant risk assessment frameworks have been proposed for example by National Academies of Sciences, Engineering, and Medicine

(2018) and O'Brien and Nelson (2020). Quantitative biorisk models such as the one presented by Sandberg and Nelson (2020) are also flexible enough to integrate automated computational design as one of the steps in a risk chain. Finally, approaches such as DeePaC are intended to help mitigate the risks by improving the accuracy of screening against potentially dangerous synthetic sequences.

Deep learning and representation learning for biological sequences are rapidly growing fields, widely adopted since the works of Alipanahi et al. (2015), Kelley et al. (2016), and J. Zhou and Troyanskaya (2015) popularized CNNs for regulatory genomics. Although DeePaC was one of the first published applications in NGS for pathogen genomics, it is only one of many contributions to the larger wave of research employing neural networks to solve a plethora of different problems in bioinformatics and computational biology. In particular, related methods are being used for other specific tasks in viral and microbial bioinformatics, and the COVID-19 pandemic has stimulated this line of research even more. The emergence of the next novel pathogen is most probably just a matter of time; interpretable, learning-based approaches can help us prepare better, respond faster, or – in the context of synthetic biology – possibly even mitigate the risks before they materialize. Reference-free phenotype prediction directly from genomes or their fragments can also find many applications beyond the tasks discussed here.

A Appendix

A.1 Predicting bacterial pathogenic potential with DeePaC

Hardware

Each of the networks was trained on between one and four GPUs (GTX 980, GTX 1080 Ti, or RTX 2080 Ti) depending on hardware availability at a given time point. CUDA 9.0 was used on machines equipped with the GTX-line cards, but the RTX 2080 Ti card required CUDA version 10.0.

Hyperparameter tuning: RC-LSTM

All networks were trained with the Adam optimizer (Kingma & Ba, 2014) using the default parameters and a batch size of 512. We finished the training after a maximum of 15 epochs or earlier if the validation accuracy did not improve for 10 consecutive epochs. We used dropout regularization (Srivastava et al., 2014) with a dropout rate of 0.5 after all recurrent and dense layers, and input dropout in most of the architectures. For the bidirectional RC-LSTMs, we tested the following parameter combinations for both the *full* and the *Siamese* variant:

- sum merge, 1 layer, 128–512 units, input dropout rate: 0, 0.1–0.3
- max merge, 1 layer, 384 units, input dropout rate: 0.1–0.3
- product merge, 1 layer, 384 units, input dropout rate: 0.1–0.3
- sum merge, 2 layers, 384 units, input dropout rate: 0.1–0.3

The input dropout rate was varied with step of 0.05 unless stated otherwise, and the number of units was adjusted with a step of 128. In addition, we trained two traditional LSTMs without RC parameter sharing, with 256–384 units and an input dropout of 0.2.

Hyperparameter tuning: RC-CNN

For the RC-CNN architectures, we used 64–512 units in the convolutional layers and 64–256 units in the dense layers. Six unit number combinations were generated by alternate doubling of one of those values at each step, starting with the number of convolutional units. We tested the following networks with sum merge, filter size of 15 and input dropout rates of 0 and 0.2–0.3 unless stated otherwise:

- *full*-RC, max pooling, 1 conv. and 1 dense layers
- *full*-RC, average pooling, 1 conv. and 1 dense layers
- *full*-RC, max pooling, 2 conv. and 2 dense layers
- *full*- and *siam*-RC, average pooling, 2 conv. and 2 dense layers
- *full*-RC, average pooling, 3 conv. and 3 dense layers, input dropout rate 0.2–0.25

We also applied batch normalization (Ioffe & Szegedy, 2015) to the RC-CNNs with 2 convolutional and 2 dense layers of 512 and 256 units (hereafter: 2x2-XL), and tested changing the filter size to 7 or 11 by training *full* RC-CNNs with the same layout. In addition, we trained two traditional CNNs without RC parameter sharing. One was a 2x2-XL architecture with the input dropput of 0.25, and the other was a 1x1 max pooling network with 256 convolutional and 128 dense units. Finally, we evaluated hybrid *full* and *Siamese* RC-networks with a convolutional layer of 512 units and filter size of 7, 11 or 15, followed by a recurrent layer of 384 units, trained with the input dropout of 0.2. Those networks were trained for just 10 epochs due to high computational cost.

Loss function and training sets

We use binary cross-entropy for all training runs. We tested the effect of adding an L_2 regularization term while training both the *full* and *Siamese* RC-LSTMs with sum merge, 1 layer of 384 units and an input dropout rate of 0.2. We used regularization rates of 10^{-2}, 10^{-3} and 10^{-5}. Next, we tested using an imbalanced training set and weighting the errors by the inverse of the relative class frequency. To this end, we trained *full* and *Siamese* RC-LSTMs with sum merge, 1 layer of 384 units and input dropout rates between 0.1 and 0.3.

A.2 Viral host-range prediction and interpretability

Architectures and training

For viral host-range prediction, we use the architectures previously selected via hyperparameter tuning for the bacterial dataset. We use the sigmoid activation for the output layer and the ReLU activation after the fully-connected (FC) layers and RC-Convolution blocks (A.1a, dashed lines). The RC-LSTM blocks (A.1b, dashed lines) use the default combination of activation functions for the LSTM layers as implemented in the `tf.keras` library. We also use dropout after the input (p=0.25), convolutional, recurrent and FC layers (p=0.5). Padding is handled automatically by Keras so that the input length does not change after convolutions. Note that the nucleotide dimension is the channel dimension (the input shape is sequence length x number of channels), and we use 1D convolutions. Hence, convolutions, padding and pooling have no vertical dimension. The merging FC layers sum representations for corresponding channels in the 'forward' and 'reverse-complement' orientation.

Virus-level models were trained on Tesla P100 GPUs using early stopping with a patience of 10 epochs for a default maximum of 14 epochs (corresponding to maximum 80h wall-time on this GPU). Species-level model were trained on Tesla V100 GPUs; as convergence took longer we allowed a maximum of 160h (corresponding to a triple increase in maximum epochs). A trained model can be used for either single reads or read pairs; in the latter case we run predictions separately for both mates and the average the final outputs. More in-depth description of the reverse-complement architectures can be found in Bartoszewicz et al., 2020. The architectures presented in Figure A.1a and Figure A.1b can be reproduced using the `deepac-vir train -r` and `deepac-vir train -s` commands, respectively; input data can be supplied using the `-T`, `-t`, `-V` and `-v` flags. Config files can be retrieved using `deepac-vir templates`, modified and used with `deepac train -c` to train custom models. More details are available in the user user guide.

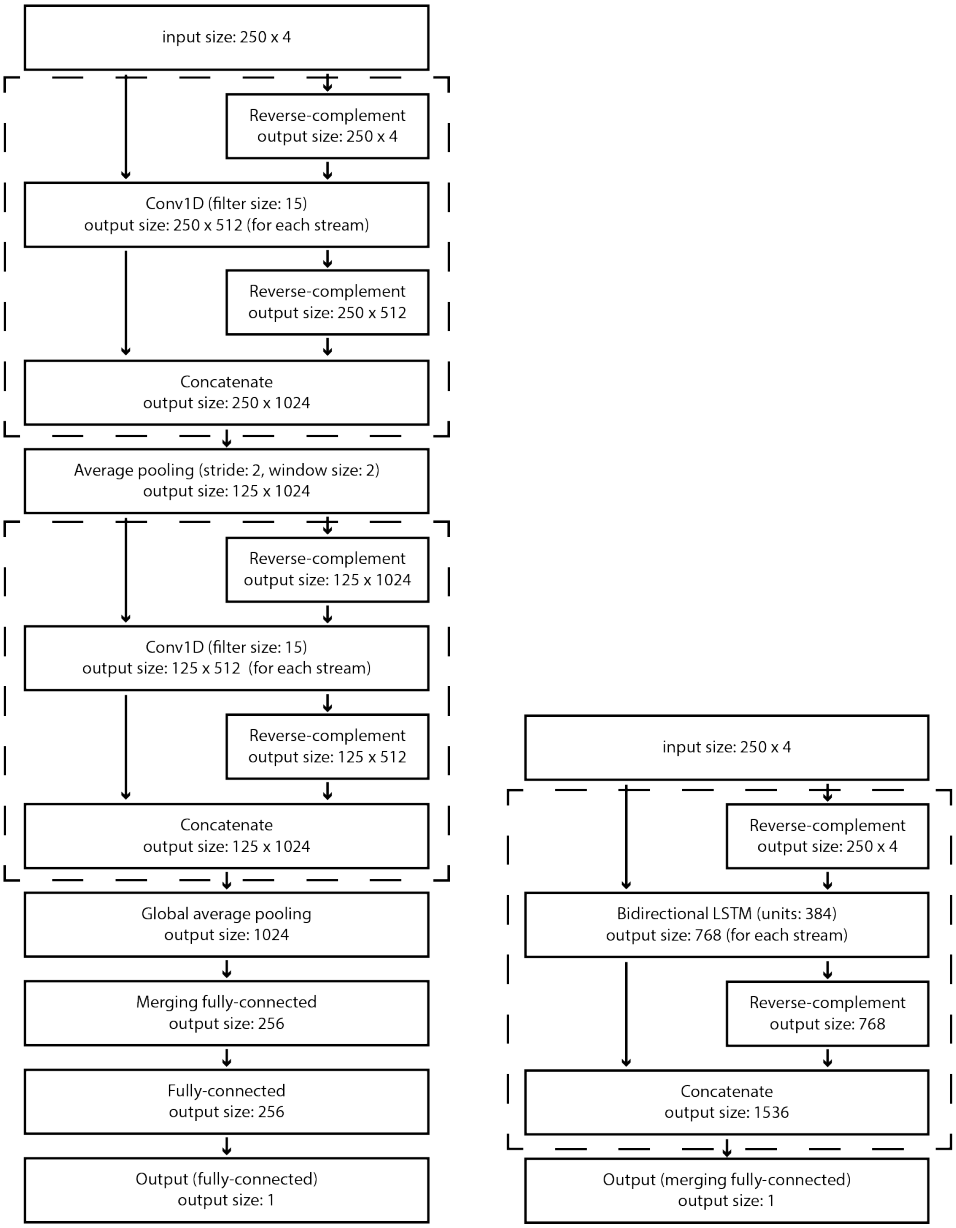


Figure A.1: Simplified plots of the *full* reverse-complement architectures used in this study. We omit the batch dimension for clarity.

Non-negative precision

Proof of Equation 3.5. Let TP be the number of true positives, TN – the number of true negatives, FP – the number of false positives and FN – the number of false negatives. If missing predictions are present, the total number of positives $P = TP + FN + MP$, where MP is the number of missing positives, or those positives for which no prediction was returned. Analogously, $N = TN + FP + MN$, where MN is the number of missing negatives. For a perfectly balanced set, $P = N$ and $TP + FN + MP = TN + FP + MN$. Assuming a balanced set,

$$
\begin{aligned}
TPPV &= \frac{TPR}{TPR + 1 - TNR} \\
&= \frac{\frac{TP}{TP+FN+MP}}{\frac{TP}{TP+FN+MP} + 1 - \frac{TN}{TN+FP+MN}} \\
&= \frac{TP}{TP + TP + FN + MP - \frac{TNTP+TNFN+TNMP}{TN+FP+MN}} \\
&= \frac{TP}{TP + \frac{TPFP+TPMN+FNFP+FNMN+MPFP+MPMN}{TN+FP+MN}} \\
&= \frac{TP}{TP + \frac{TP(FP+MN)+FN(FP+MN)+MP(FP+MN)}{TN+FP+MN}} \\
&= \frac{TP}{TP + \frac{(TP+FN+MP)(FP+MN)}{TN+FP+MN}} \\
&= \frac{TP}{TP + FP + MN} \\
&= \frac{TP}{TP + N - TN}
\end{aligned}
$$

□

Note that due to rounding effects in per-genome coverage calculation, the number of positive and negative reads in the simulated datasets may be slightly different. However, those differences are negligible and orders of magnitude smaller than the size of the dataset ($Prev \approx 0.50$). Therefore, we treat the datasets as perfectly balanced, ignoring the prevalence correction from Equation 3.2.

Table A.1: Classification accuracy depending on the negative class definition, read pairs. Euk. – Eukaryota dataset; Met. – Metazoa dataset, Cho. – Chordata dataset, Str. – Stratified dataset, X-150 – first 150bp of each read in X. Training set in subscript of the model name; test set in column headers. Recall (Rec.) is identical in all cases, as the positive class remains unchanged. Best performance in bold. CNN_{All} achieves best overall accuracy.

	All	Euk.	Met.	Cho.	Str.	Rec.
CNN_{All}	**89.9**	85.9	83.3	78.6	**88.1**	**85.4**
CNN_{Cho}	84.9	84.2	83.6	82.4	84.6	71.7
CNN_{Str}	88.2	**86.4**	**85.1**	**82.7**	87.4	78.8
$CNN_{All\text{-}150}$	89.4	85.5	82.9	78.4	87.7	83.2
$CNN_{Str\text{-}150}$	88.2	86.3	84.9	82.5	87.3	78.3
$LSTM_{All}$	86.4	78.2	74.1	65.5	82.6	83.0
$LSTM_{Cho}$	82.8	81.9	80.8	80.0	82.4	70.6
$LSTM_{Str}$	85.8	82.2	79.6	75.2	84.2	76.3

Table A.2: Classification performance in the fully open-view setting (all virus hosts), single reads. Acc. – accuracy, Prec. – precision, Rec. – recall, Spec. – specificity. BLAST yields no predictions for over 12% of the samples. Best performance in bold.

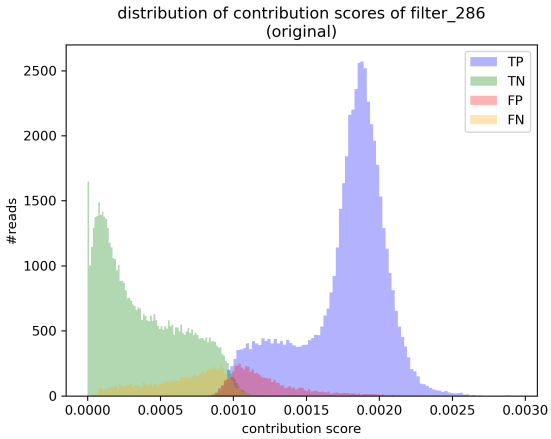
	Acc.	Prec.	Rec.	Spec.
CNN_{All} (ours)	**87.8**	89.9	**85.2**	**90.4**
$LSTM_{All}$ (ours)	84.7	86.0	82.8	86.5
kNN	75.5	76.3	73.9	77.1
BLAST	78.4	**98.3**	79.2	77.6

Table A.3: Classification performance, novel species of Chordata-infecting viruses. Top: paired reads. BLAST yields predictions for only 56.6% of the pairs. Bottom: whole available genomes or contigs – negative class is the majority class. 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, and the results for genomes can be further improved by retuning the classification threshold (see chapter 5, Table 5.3)).

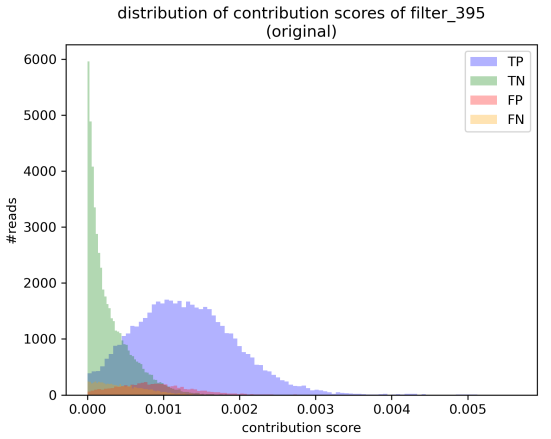
		BAcc.	Prec.	Rec.	Spec.
Read pairs	$CNN_{SP\text{-}Cho}$ (ours)	**58.2**	60.6	47.1	**69.3**
	$CNN_{SP\text{-}All}$ (ours)	56.5	56.3	**57.9**	55.1
	BLAST	37.1	**65.8**	19.1	55.1
Genomes	$CNN_{SP\text{-}Cho}$ (ours)	54.0	42.3	22.9	**85.0**
	$CNN_{SP\text{-}All}$ (ours)	50.3	32.8	**40.6**	60.0
	BLAST (reads)	57.5	**47.9**	35.4	79.5
	BLAST (genome)	**58.0**	49.3	38.5	77.5
Contigs	BLAST (contigs)	51.6	42.2	37.1	66.2

Table A.4: Gene ranking for *S. aureus* (top 3 out of 870). hupB is indirectly engaged in virulence. Our method detects functionally relevant genes using the original DeePaC RC-CNN model.

Rank	Gene	Score	Biological Process
1	sarR	0.644	**Virulence**
2	hupB	0.642	DNA condensation
3	sspB	0.637	**Virulence**

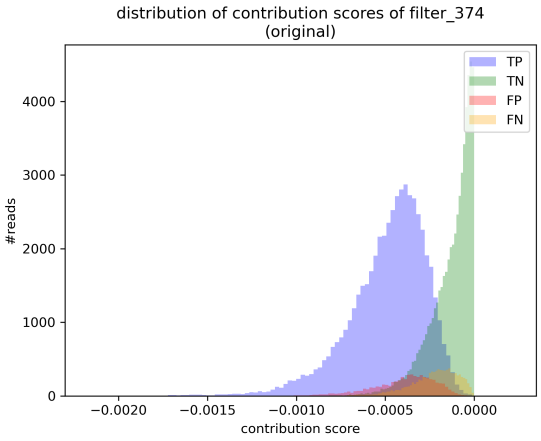


(a)

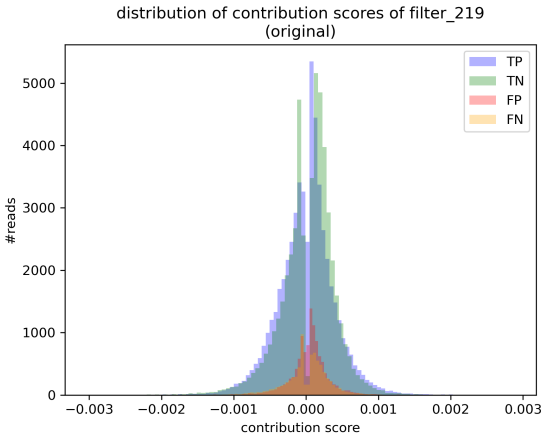


(b)

Figure A.2: Contribution histograms of selected filters. A.2a: The 'ambiguity detector' with the highest mean contribution score. A.2b-A.2d: Filters presented in Figure 3.1. A.2b: Second-highest mean contribution score. Contributions are higher for the positive class.



(c)



(d)

Figure A.2: (continued) Contribution histograms of selected filters. A.2b-A.2d: Filters presented in Figure 3.1. A.2c: Lowest mean contribution score. Positive class reads are 'penalized' with stronger negative contributions than the negative class reads. A.2d: Oscillating filter from the original, bacterial DeePaC RC-CNN. Symmetric contribution distribution suggests that the contributions of the filter are context-dependent, although positive on average.

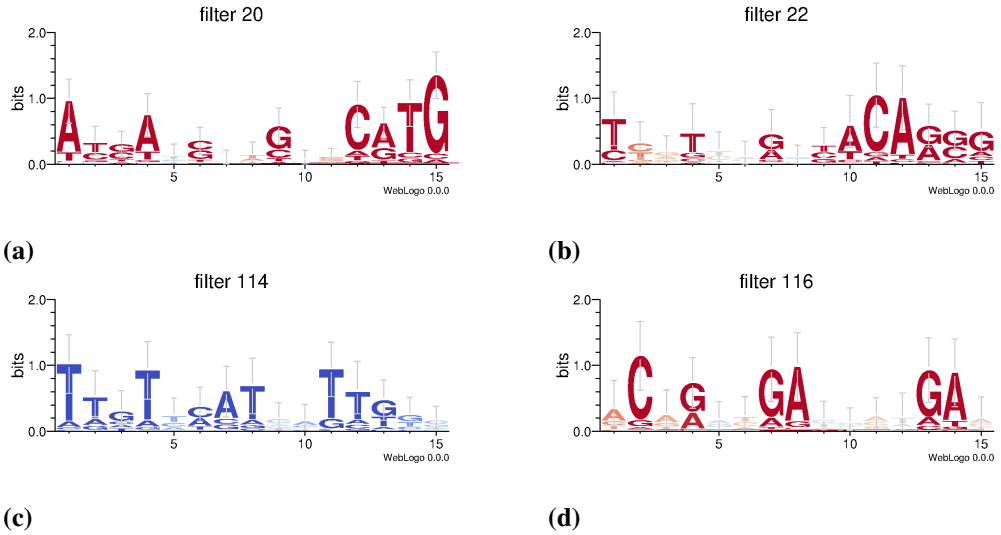
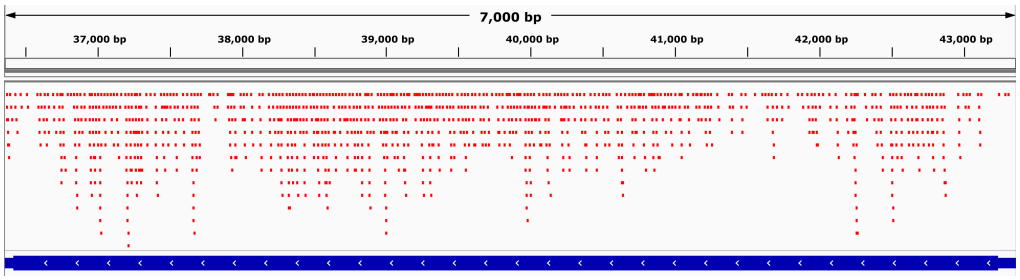
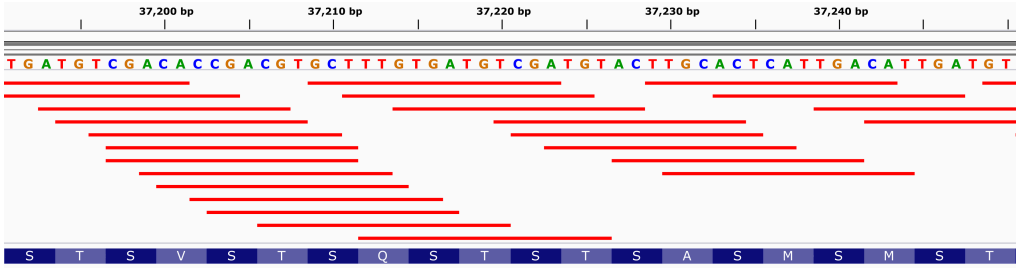


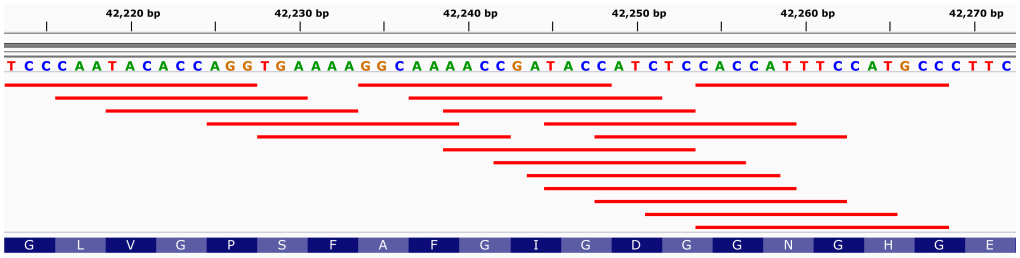
Figure A.3: Example oscillating filters with a codon-like structure, extracted from the CNN_{All} model. Consensus sequences: AYSANSNNGNNCATG (A.3a), TYNTNNRNNACARSG (A.3b), TTGTNMATNNTTKNN (A.3c), ACNRNNGANNNNGAN (A.3d).



(a)



(b)



(c)

Figure A.4: Activations of filter 219 (original DeePaC RC-CNN) in the sraP gene of *S. aureus* subsp. *aureus* 21200. Top, in red: filter hits. Bottom, in blue: reference sequence annotation (contig 19). A.4a: sraP, a serine-rich adhesin for platelets, the top hit gene in the enrichment analysis for filter 219. Serine-rich repeat regions cover over 71% of the sequence. A.4b: An example stretch of filter 219 activations in a serine-rich repeat region. The filter seems to detect the serine repeats. A.4c: An example stretch of filter 219 activations in a non-repeat region. The filter seems to detect a local glycine repeat.

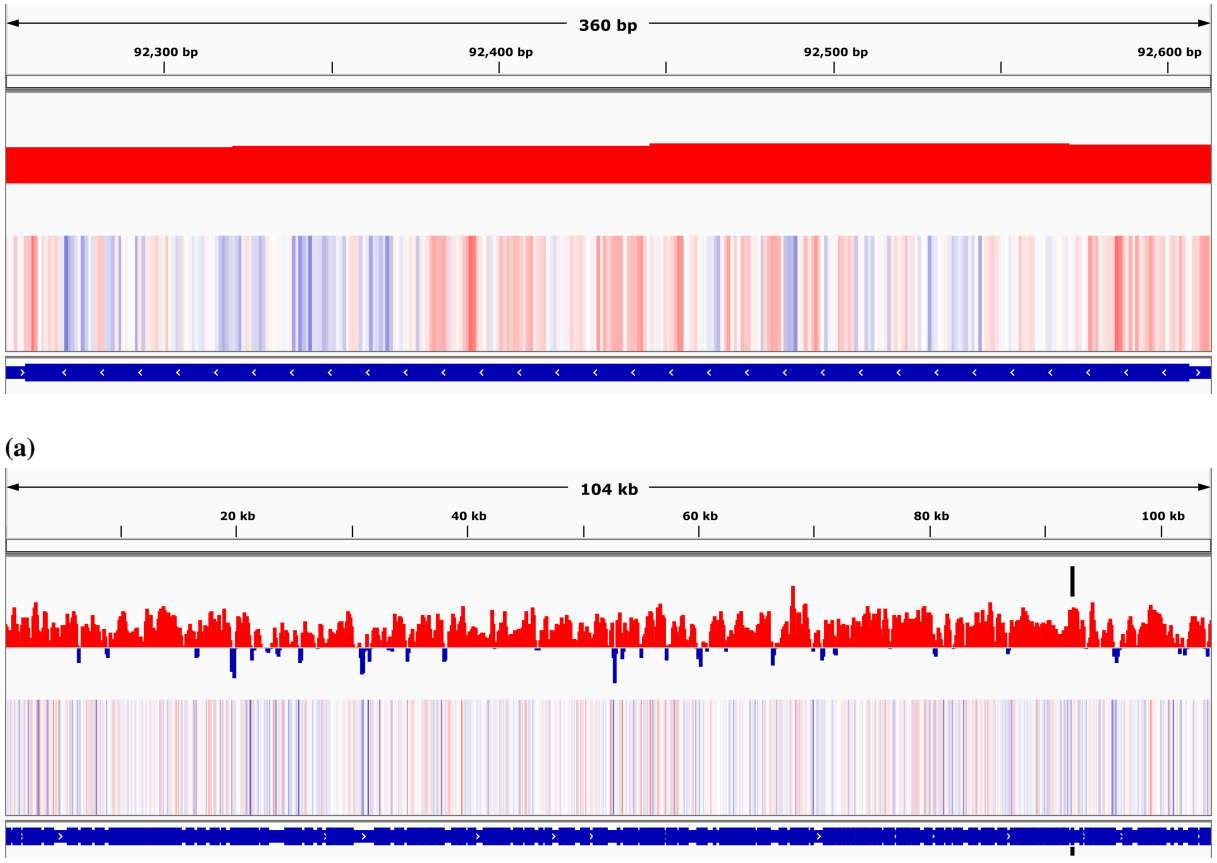


Figure A.5: GWPA of *S. aureus* subsp. *aureus* 21200. Top: score predicted by the original DeePaC RC-CNN. Heatmap: nucleotide contributions. Bottom, in blue: reference sequence annotation. Negative contributions marked with blue, positive contributions in red. A.5a: sarR gene, with the highest average pathogenicity score. A.5b: Contig 18, with sarR location marked with black bars. Neighbouring higher peaks belong to putative genes lacking ground truth annotation.

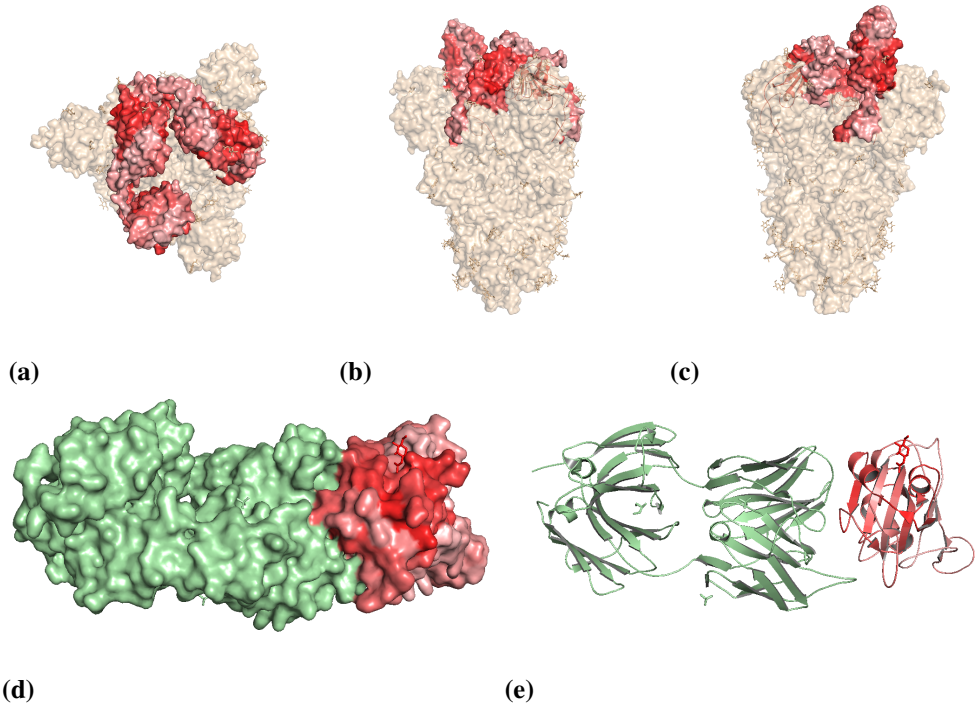


Figure A.6: Average nucleotide contribution scores per residue. This figure is analogous to Figure 3.3, which presents the average predicted pathogenic potential. A.6a-A.6c: Top and side view of the SARS-CoV-2 spike glycoprotein. Three receptor-binding domains (RBDs) are colored according to the per-residue average contribution score of the corresponding genomic sequence. Since all contributions are positive, the color gradient is dominated by different shades of red. One of the domains is in the 'up' conformation. Darker red regions corresponding to the peak in Figure 3.2c are located in the core-RBD subdomain. A.6d: RBD in complex with a SARS-neutralizing antibody CR3022 (green). The darker red region covers over 70% of the CR3022 epitope, but spans also to the neighboring fragments, including the N343 glucosaminylation site (carbohydrate in red stick representation). This is a part of the epitope of another neutralizing antibody, S309. A.6e: Cartoon representation of Figure A.6d. The red region is centered on two exposed α-helices surrounding the core β-sheet (lower score, light red).

A.3 Detecting novel pathogens in real time with DeePaC-Live

In this study, we use modified ResNet architectures presented in Table A.6 and Figure A.7. The reverse-complement convolution blocks (RC-Conv) output feature maps in the forward and reverse-complement orientations, represented as small arrows between blocks (the 'forward' and 'reverse-complement' streams). The skip connections (larger arrows) maintain invariance to reverse-complementarity by either simply summing the outputs of two layers (as in a standard ResNet) or by applying a size-1 RC-convolution if the layer dimensions do not match (similarly to how a standard ResNet uses standard size-1 convolutions). Note that in contrast to a standard ResNet, all convolutional layers are 1D-convolutions, since the inputs are one-hot encoded nucleotide sequences. The first dashed rectangle corresponds to the conv2 stage in Table A.6, and the second dashed rectangle represents stages conv3-conv5. The architecture presented here can be reproduced using the `deepac train -c config.ini` command, where `config.ini` is one of the config files available at https://doi.org/10.5281/zenodo.4456008 along the trained models. `deepac` is installed automatically with `deepac-live`. The necessary training and validation datasets (see Table A.5) are hosted at https://doi.org/10.5281/zenodo.4456857.

Table A.5: A summary of the datasets used in this study. Simulated datasets have been prepared based on datasets of Bartoszewicz, Seidel, and Renard (2021a) and Bartoszewicz et al. (2020). Subread test sets with read lengths between 25 and 250 (step of 25) were generated based on the original simulated Illumina test sets; we only list the them once in the table for clarity. All simulated datasets are hosted on Zenodo, and the real datasets are available in the SRA database (see Zenodo IDs and SRA accession numbers in the last column).

content	technology	positive reads	negative reads	ID / accession
Bacteria (train.)	Illumina (sim.)	10M	10M	4456857
Bacteria (val.)	Illumina (sim.)	1.25M	1.25M	4456857
Bacteria (test)	Illumina (sim.)	2x0.625M	2x0.625M	3678563
Viruses (train.)	Illumina (sim.)	10M	10M	4456857
Viruses (val.)	Illumina (sim.)	1.25M	1.25M	4456857
Viruses (test)	Illumina (sim.)	2x0.625M	2x0.625M	4312525
Bacteria (train.)	Nanopore (sim.)	10M	10M	4456857
Bacteria (val.)	Nanopore (sim.)	1.25M	1.25M	4456857
Bacteria (test)	Nanopore (sim.)	1.25M	1.25M	4456857
Viruses (train.)	Nanopore (sim.)	10M	10M	4456857
Viruses (val.)	Nanopore (sim.)	1.25M	1.25M	4456857
Viruses (test)	Nanopore (sim.)	1.25M	1.25M	4456857
S. aureus	Illumina	2x1.1M	0	SRR5110368
SARS-CoV-2	Illumina	2x517.3k	0	SRR11314339
S. aureus	Nanopore	83.4k	0	SRR8776887
SARS-CoV-2	Nanopore	396.4k	0	SRR11140745

Table A.6: ResNet architecture details. Conv1 and first layers of stages conv3-conv5 use a stride of 2, and all other layer use the stride of 1. Stages 2–5 consist of multiple layers with the same filter width and number of filters. Batch normalization is used after all hidden layers. After the convolutions, we use global average pooling and a fully-connected output layer.

stage	ResNet-18	ResNet-34
conv1	filter width:7, filters:64	filter width:7, filters:64
conv2	[filter width:5, filters:64] x 4	[filter width:5, filters:64] x 6
conv3	[filter width:5, filters:128] x 4	[filter width:5, filters:128] x 8
conv4	[filter width:5, filters:256] x 4	[filter width:5, filters:256] x 12
conv5	[filter width:5, filters:512] x 4	[filter width:5, filters:512] x 6
pool	global average pooling	global average pooling
out	1-unit fully-connected	1-unit fully-connected

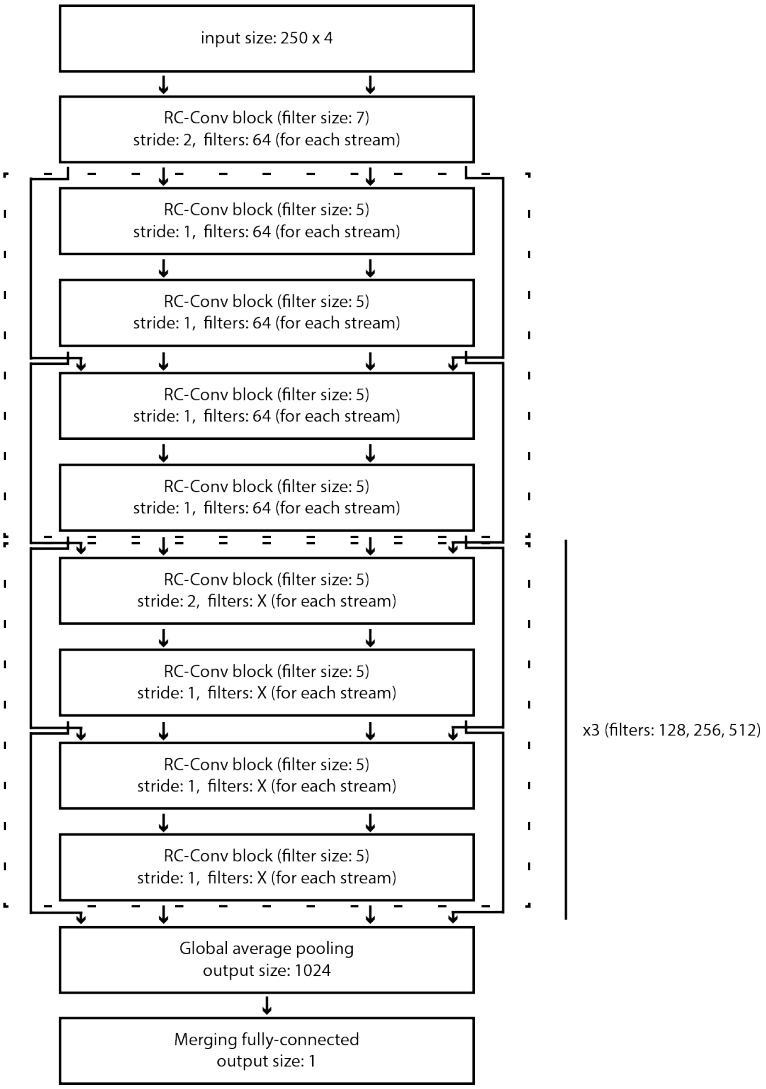


Figure A.7: A simplified visualization of the 18-layer variant of the reverse-complement ResNet used in this study (see Table A.6). We omit the batch dimension for clarity.

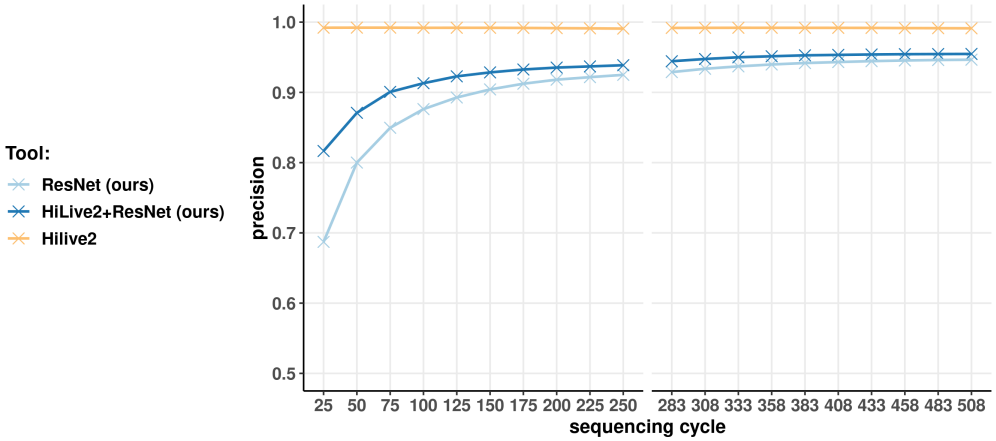


Figure A.8: Precision for the viral dataset. We compared HiLive2's stable performance to the viral hybrid classifier and the ResNet alone, which achieved precision comparable to alignment-based approaches. The hybrid classifier crosses a 90% threshold at cycle 75 (90.1%), while never plunging below 80% even for the earliest cycles. What is more, all of the HiLive2-mapped reads are included in the hybrid classifier's predictions, so no information is lost by employing the extended approach. The high precision resulting from combining the real-time mapper with the deep learning classifier suggests that the associations of reads and a pathogenic phenotype are trustworthy even at the early stages of the sequencing run, getting even more reliable as more information is gained. Precision approaches HiLive2's, especially for the later cycles.

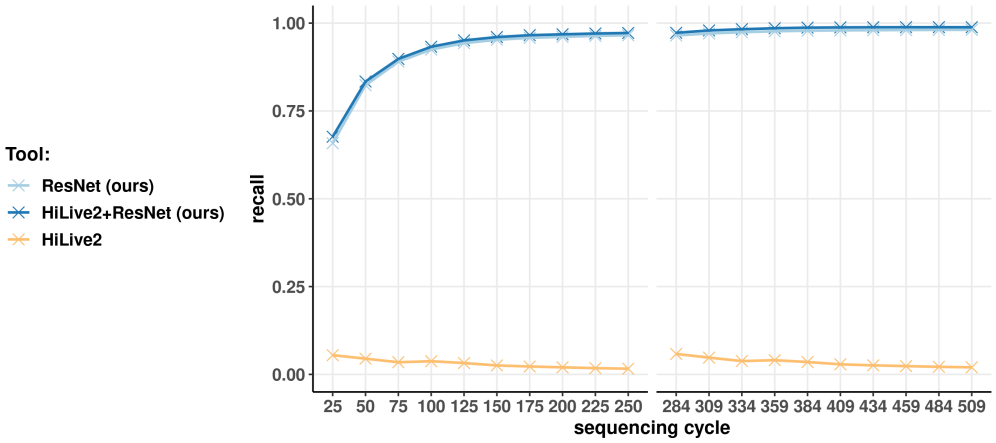


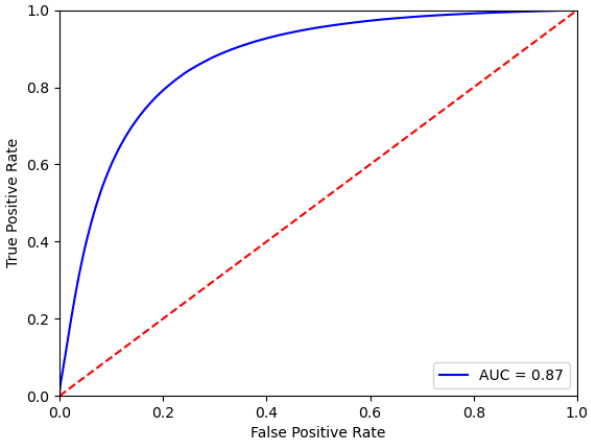
Figure A.9: Recall (true positive rate) for the *S. aureus* sequencing run. As this is a pure pathogen sample, recall is equal to accuracy. A combination of HiLive2 and ResNet correctly identifies 98.8% of the read pairs after the last cycle, and 94.9% on average over the whole run. For this particular species, the performance of the ResNet itself is only marginally worse (98.1% after the last cycle and 94.0% on average).

Table A.7: Inference speed in reads per second and reads per sequencing time of 25 cycles. This is an inherently difficult comparison, as inference with neural networks can be accelerated with GPUs, while other methods cannot. We used a desktop computer equipped with a consumer-grade GPU to benchmark the throughput of deep learning approaches, and a 128-core machine with 500GiB RAM for the other methods. All runtimes were calculated for single 250bp reads. Running PaPrBaG on more than 8 cores was not possible due to high memory consumption (over 400GiB, compared to less than 4GiB for the CNNs, LSTMs and ResNets). BLAST database size influences alignment time. Therefore, we present separate runtimes for the bacterial ($BLAST_{Bac}$) and viral ($BLAST_{Vir}$) dataset. Top four models (bold) offer relatively similar performance, crossing the arbitrary threshold of 5000 classified reads per second on a single consumer-grade GPU. CNN (adj.) and LSTM (adj.) differ from the previously published DeePaC versions only by the adjusted inference batch size.

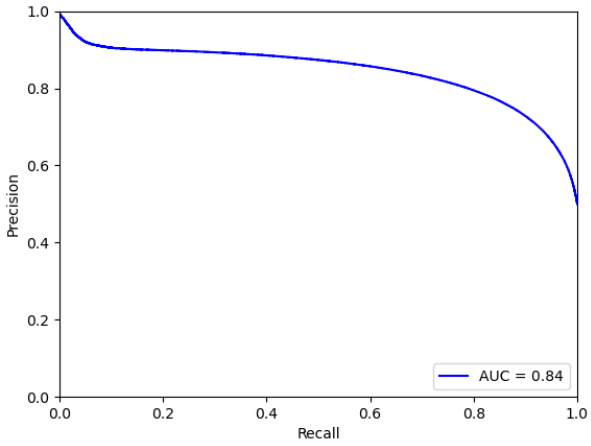
	Device	Reads/s	Reads/25c
CNN (adj.)	1x Nvidia RTX 2080 Ti	**6313**	**63.1M**
LSTM (adj.)	1x Nvidia RTX 2080 Ti	**5896**	**58.9M**
ResNet	1x Nvidia RTX 2080 Ti	**5656**	**56.5M**
DeePaC (CNN)	1x Nvidia RTX 2080 Ti	**5000**	**50.0M**
ResNet-34	1x Nvidia RTX 2080 Ti	2880	28.8M
DeePaC (LSTM)	1x Nvidia RTX 2080 Ti	1855	18.5M
PaPrBaG	8x Intel Xeon E5-4667 v4 @ 2.20GHz	906	9.0M
$BLAST_{Vir}$	100x Intel Xeon E5-4667 v4 @ 2.20GHz	833	8.3M
kNN	100x Intel Xeon E5-4667 v4 @ 2.20Ghz	37	0.3M
$BLAST_{Bac}$	100x Intel Xeon E5-4667 v4 @ 2.20GHz	160	1.6M

Table A.8: Performance on real Nanopore data. Minimap2 was evaluated on both full reads and 250bp subreads. ResNets were trained on Nanopore data with identical species composition as the Illumina data used for DeePaC CNNs and LSTMs. and evaluated on 250bp subreads. Mean sequencing time and maximum sequencing time per read correspond to estimated sequencing times for the mean read length and maximum read length in a dataset (assuming the sequencing speed of 500bp/s).

		Recall	Mean seq. time	Max. seq. time
S. aureus	ResNet	**94.7**	0.5 s/read	0.5 s/read
	minimap2 (250bp)	3.3	0.5 s/read	0.5 s/read
	minimap2 (full)	66.9	15.9 s/read	107.2 s/read
SARS-CoV-2	ResNet	**52.7**	0.5 s/read	0.5 s/read
	minimap2 (250bp)	4.6	0.5 s/read	0.5 s/read
	minimap2 (full)	9.9	1.3 s/read	12.7 s/read

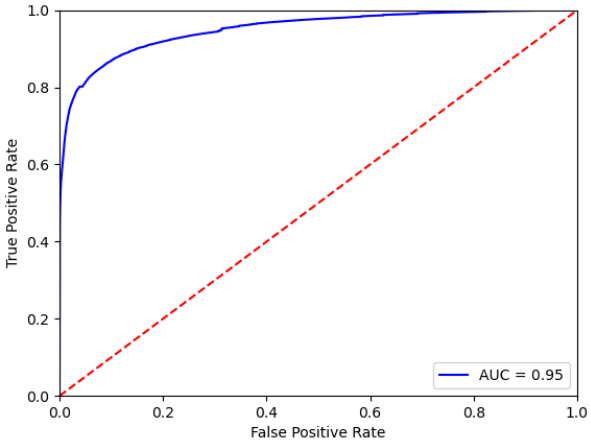


(a) ROC curve, bacteria

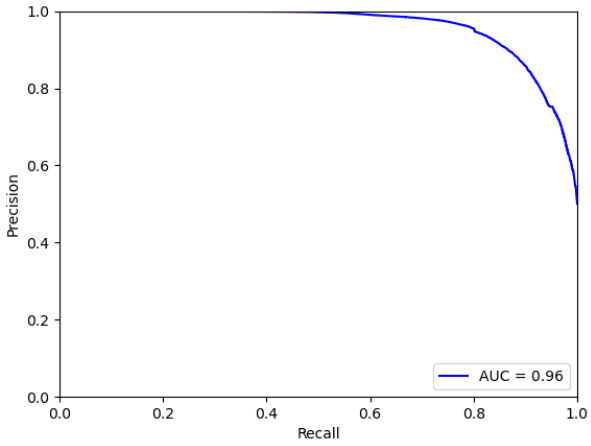


(b) PR curve, bacteria

Figure A.10: ROC and PR curves for Nanopore-trained ResNets evaluated on the bacterial and viral Nanopore test sets. Note that if a user wishes to retune the classification threshold according to custom optimality criteria, we would advise reevaluating the selected threshold on another, separate held-out dataset. Alternatively (if this not possible), one can select the threshold using the validation set (available at https://doi.org/10.5281/zenodo.4456857), and then perform the final evaluation with a fixed threshold on the test set.



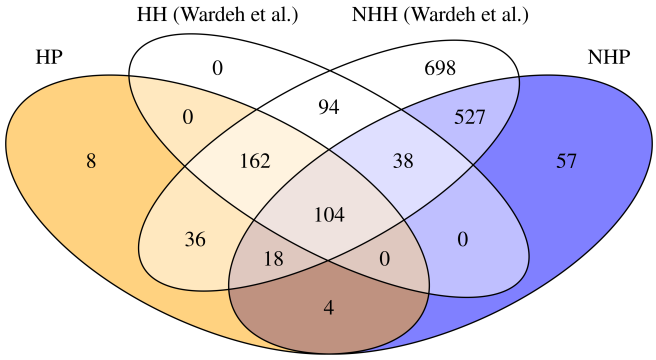
(c) ROC curve, viruses



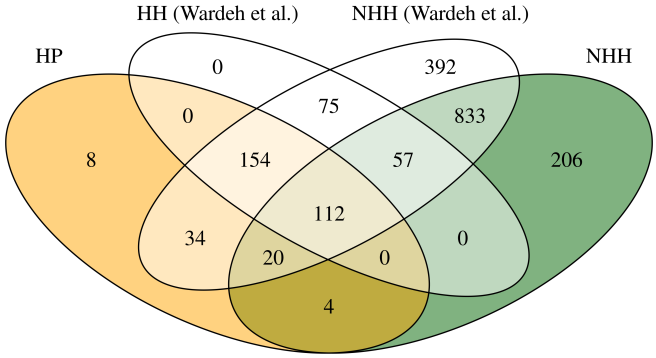
(d) PR curve, viruses

Figure A.10: (continued) ROC and PR curves for Nanopore-trained ResNets evaluated on the bacterial and viral Nanopore test sets. Note that if a user wishes to retune the classification threshold according to custom optimality criteria, we would advise reevaluating the selected threshold on another, separate held-out dataset. Alternatively (if this not possible), one can select the threshold using the validation set (available at https://doi.org/10.5281/zenodo.4456857), and then perform the final evaluation with a fixed threshold on the test set.

A.4 Fungal host prediction and detecting multiple pathogen classes

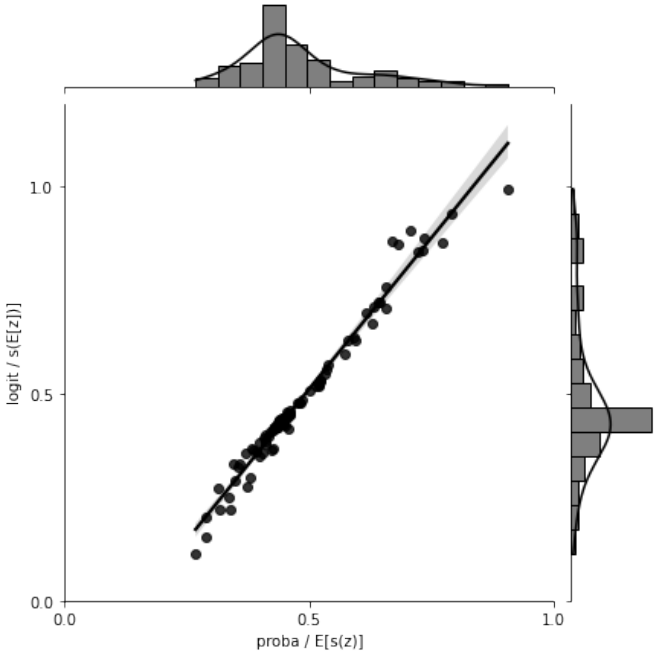


(a) Core database

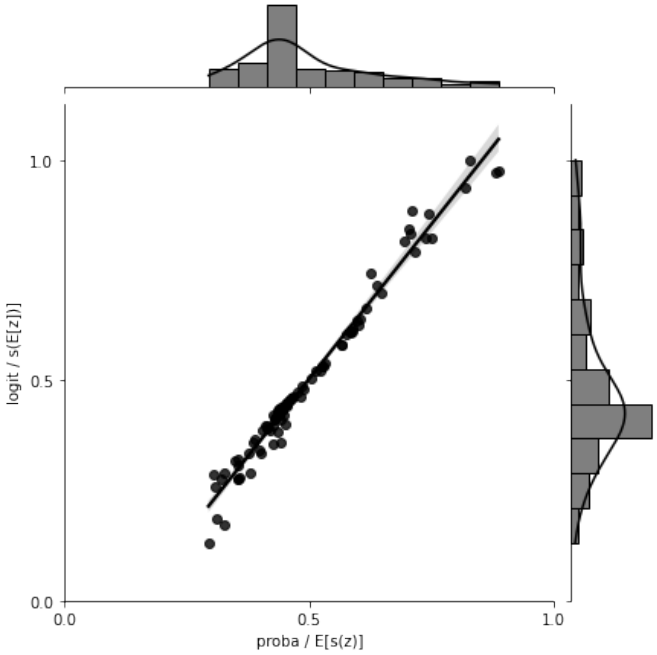


(b) Core database + genomes of plant-associated species

Figure A.11: A Venn diagram of the manually confirmed labels and putative labels from EID2 (Wardeh et al., 2015) for all labelled genomes. HP – human pathogens (manually curated); NHP – non-human pathogens (manually curated); NHH – species with a non-human host (manually curated); HH (Wardeh et al.) – species with a putative human host (EID2); NHH (Wardeh et al.) – species with a putative non-human host (EID2).

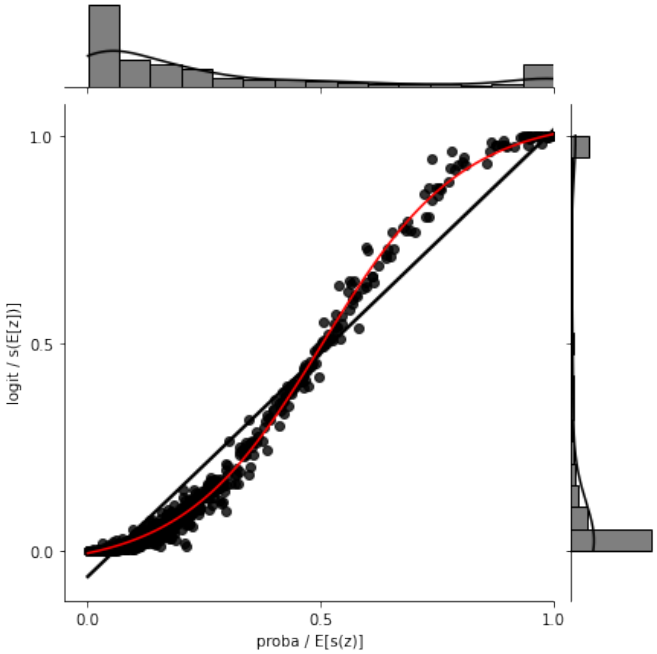


(a) Fungal validation dataset



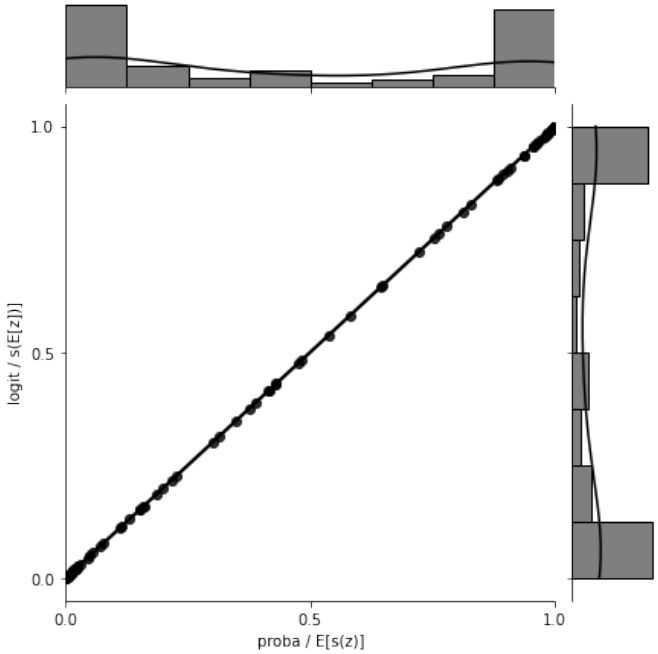
(b) Fungal test dataset

Figure A.12: Comparison of relationships between 'logit'-average and 'proba'-average predictions. Real data.

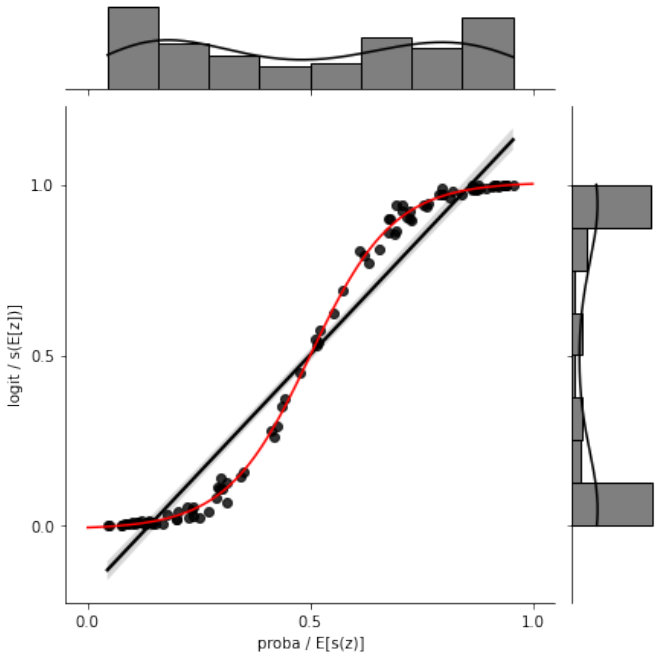


(c) Viral test dataset

Figure A.12: (continued) Comparison of relationships between 'logit'-average and 'proba'-average predictions. Real data.

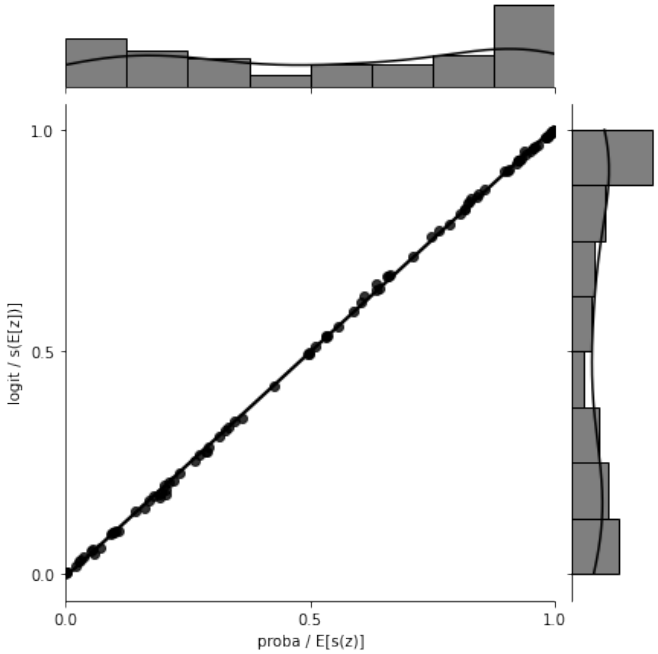


(a) Two classes, well separated, low within-species variances

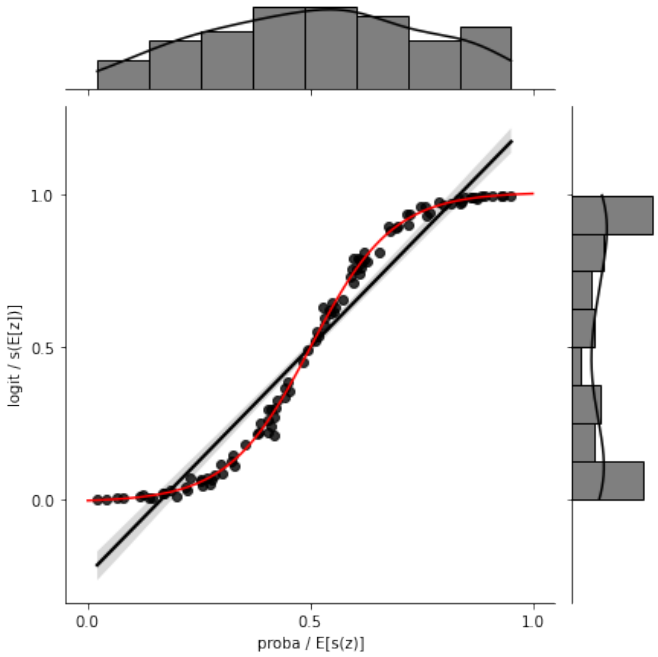


(b) Two classes, well separated, high within-species variances

Figure A.13: Comparison of relationships between 'logit'-average and 'proba'-average predictions. Simulated z values.

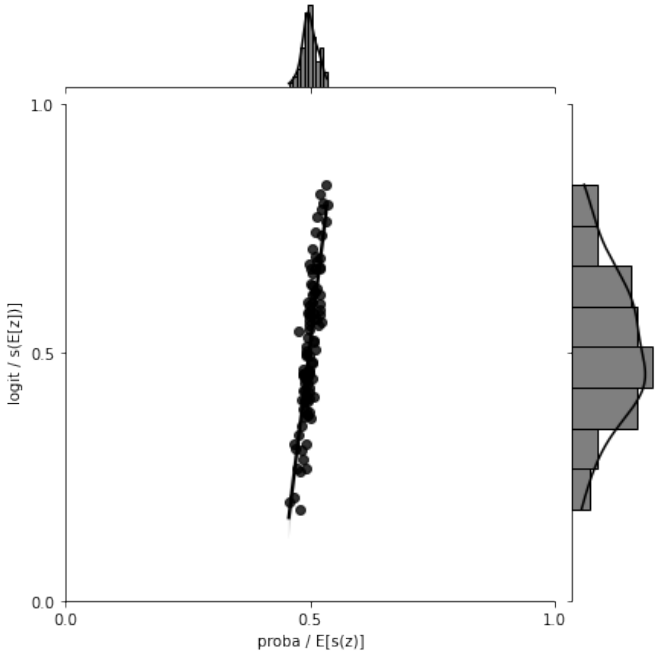


(c) No class distinction, low within-species variances

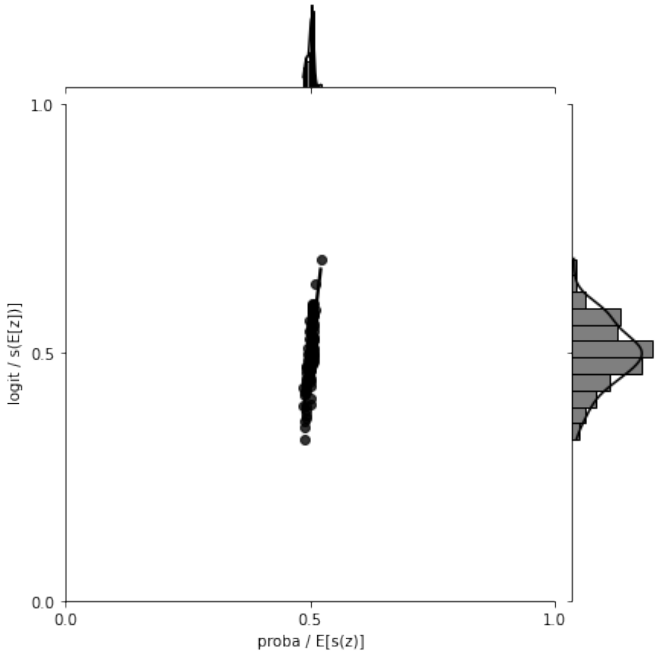


(d) No class distinction, high within-species variances

Figure A.13: (continued) Comparison of relationships between 'logit'-average and 'proba'-average predictions. Simulated z values.



(e) No species signal, extremely high per-read variance



(f) Analogous to Figure A.13e, increased coverage

Figure A.13: (continued) Comparison of relationships between 'logit'-average and 'proba'-average predictions. Simulated z values.

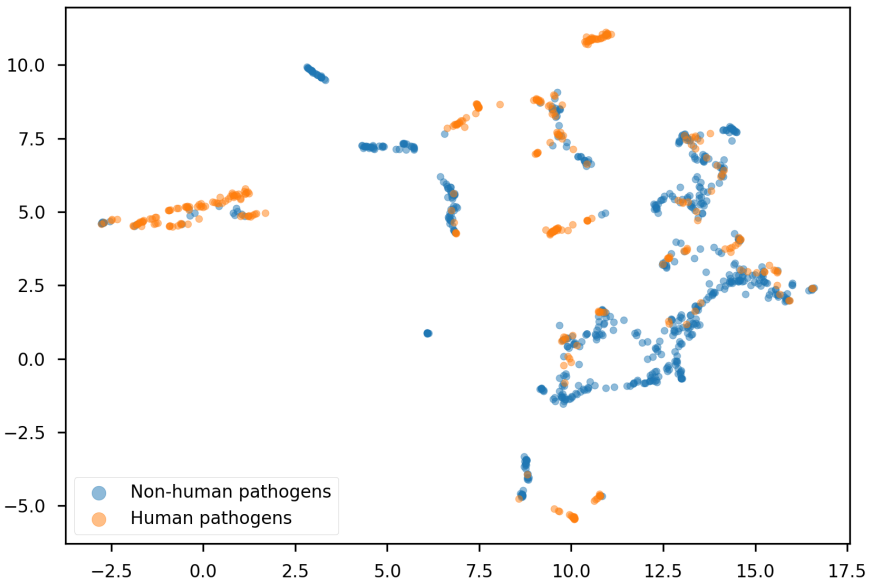


Figure A.14: UMAP embeddings of the learned genome representations for the core database; labels predicted by BLAST. Training species are present in the reference database, so predicting labels for them is easy (99.9% accuracy).

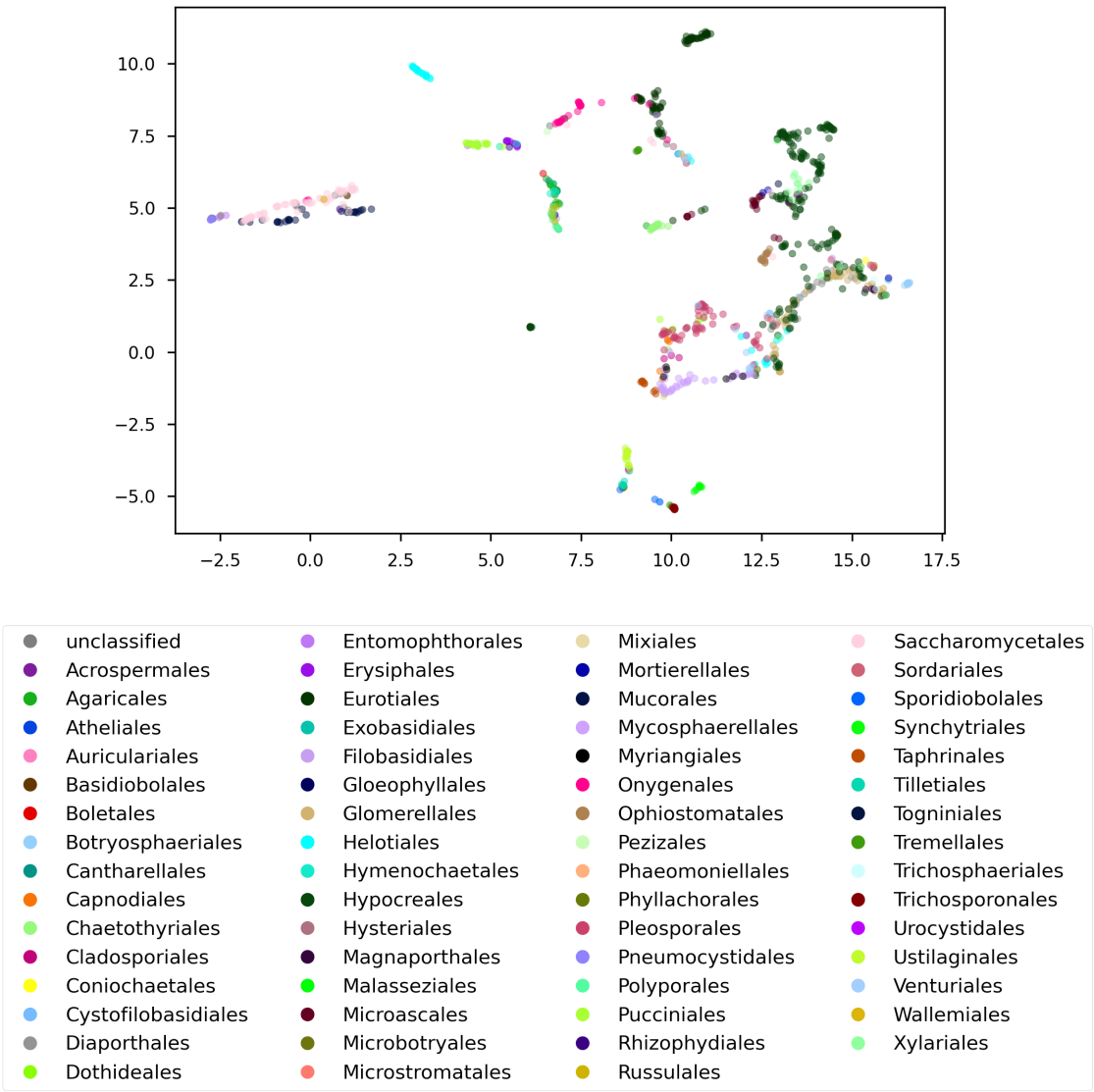


Figure A.15: UMAP embeddings of the learned genome representations for the core database; taxonomic rank: order.

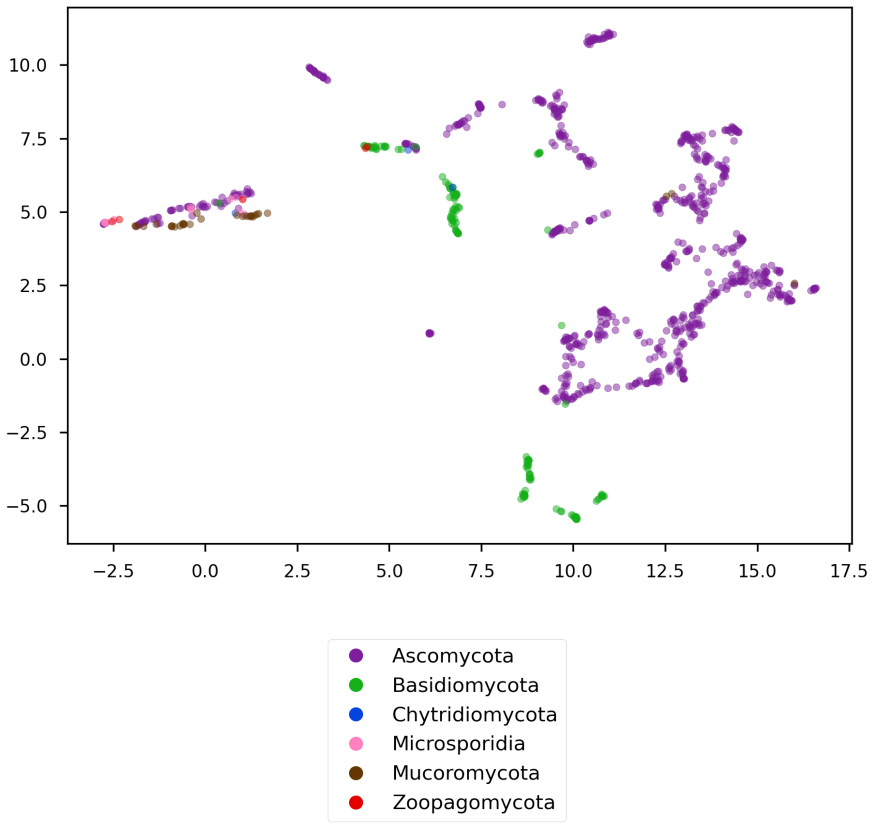


Figure A.16: UMAP embeddings of the learned genome representations for the core database; taxonomic rank: phylum.

Table A.9: Effects of integrating multiple classes on the classification performance on non-fungal datasets, read pairs. The 4-class classifier includes the human-pathogenic fungi class along the three viral and bacterial classes included in the 3-class classifier. The 2-class classifiers are the original DeePaC ResNets (Bartoszewicz, Genske, et al., 2021a) for bacteria and viruses accordingly. For the viral dataset, assignments to the bacterial pathogens class may be assumed to actually reflect bacteriophages infecting those bacteria due to high bacteriophage-host sequence similarity. Classifiers reassigning the bacterial assignments into non-pathogen class based on this assumption are listed as 'reass.'. Integrating fungi into the classifiers does not significantly impact the overall accuracy. Differentiating between non-human viruses and bacteria is more challenging. The 'reassigning' classifiers perform on par with the binary classifiers, and multi-class networks are somewhat less accurate on the bacterial data, still outperforming BLAST by a significant margin (see Table 5.5). Acc. – accuracy, Prec. – precision; Rec. – recall, Spec. – specificity, AUPR – area under the precision-recall curve (calculated for the respective positive class).

		Acc.	Prec.	Rec.	Spec.	AUPR
Bacteria	4-class ensemble	85.1	**84.1**	87.2	**83.0**	88.3
	4-class	83.6	80.9	89.0	78.1	88.6
	3-class	84.0	80.6	89.7	78.2	**90.0**
	2-class	**87.3**	83.6	**92.7**	81.8	89.0
Viruses	4-class ensemble	81.4	94.3	90.9	71.8	**98.1**
	4-class	79.0	93.4	**91.2**	66.7	97.9
	3-class	81.9	93.3	91.1	72.7	97.8
	4-class ensemble, reass.	89.6	**95.3**	89.3	90.0	**98.1**
	4-class, reass.	89.6	94.7	89.6	89.7	97.9
	3-class, reass.	**91.8**	95.1	88.1	**95.4**	97.8
	2-class	90.3	94.7	85.5	95.2	97.2

Table A.10: Effects of integrating multiple classes on the classification performance on the multi-class dataset, read pairs. The 4-class classifier includes the fungi class along the three viral and bacterial classes included in the 3-class classifier. The best performance for each class is marked in bold. In this setting, the true positive rate corresponds to the rate of correct assignments within a given class. Hence, recall is equal to accuracy for each individual class. We use the F1 score as an additional measure. Integrating fungi into the classifiers does significantly impact the overall accuracy for any of the non-fungal classes, and the 4-class ensemble offers a more balanced performance on putative non-pathogen reads than the single-network alternative. The metrics labelled as 'All classes' were calculated for all four classes for the 4-class models, and for the three relevant classes for the 3-class model. Acc. – accuracy, F1 – F1 score, Prec. – precision, Rec. – recall, AUPR – area under the precision-recall curve.

		Acc.	F1	Prec.	Rec.	AUPR
All classes	4-class ensemble	87.6	87.7	87.7	87.6	93.4
	4-class	86.6	86.7	86.8	86.6	92.8
	3-class	85.5	85.6	85.8	85.5	92.4
Non-pathogens	4-class ensemble	**77.4**	**78.7**	80.1	**77.4**	86.7
	4-class	72.5	76.7	81.5	72.5	85.3
	3-class	75.5	78.6	**81.8**	75.5	**87.6**
Path. bacteria	4-class ensemble	87.2	**85.1**	**83.2**	87.2	90.4
	4-class	89.0	83.8	79.1	89.0	90.4
	3-class	**89.7**	84.2	79.3	**89.7**	**91.1**
Human viruses	4-class ensemble	90.9	**93.7**	**96.7**	90.9	98.4
	4-class	**91.2**	93.4	95.7	**91.2**	98.1
	3-class	91.1	93.6	96.3	91.1	**98.5**
Fungi	4-class ensemble	**95.0**	**92.9**	**90.9**	**95.0**	**97.9**
	4-class	93.7	92.2	90.7	93.7	97.4

Bibliography

Ahlgren, N. A., Ren, J., Lu, Y. Y., Fuhrman, J. A., & Sun, F. (2017). Alignment-free d_2^* oligonucleotide frequency dissimilarity measure improves prediction of hosts from metagenomically-derived viral sequences. *Nucleic Acids Research*, *45*(1), 39–53. https://doi.org/10.1093/nar/gkw1002 (cit. on p. 14)

Ahmed, O., Rossi, M., Kovaka, S., Schatz, M. C., Gagie, T., Boucher, C., & Langmead, B. (2021). Pan-genomic matching statistics for targeted nanopore sequencing. *iScience*, *24*(6), 102696. https://doi.org/10.1016/j.isci.2021.102696 (cit. on p. 111)

Ahn, T.-H., Chai, J., & Pan, C. (2015). Sigma: Strain-level inference of genomes from metagenomic analysis for biosurveillance. *Bioinformatics*, *31*(2), 170–177. https://doi.org/10.1093/bioinformatics/btu641 (cit. on pp. 8, 77, 98)

Alfaro, J. A., Bohländer, P., Dai, M., Filius, M., Howard, C. J., van Kooten, X. F., Ohayon, S., Pomorski, A., Schmid, S., Aksimentiev, A., Anslyn, E. V., Bedran, G., Cao, C., Chinappi, M., Coyaud, E., Dekker, C., Dittmar, G., Drachman, N., Eelkema, R., … Joo, C. (2021). The emerging landscape of single-molecule protein sequencing technologies. *Nature Methods*, *18*(6), 604–617. https://doi.org/10.1038/s41592-021-01143-1 (cit. on p. 136)

Alipanahi, B., Delong, A., Weirauch, M. T., & Frey, B. J. (2015). Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning. *Nature Biotechnology*, *33*(8), 831–838. https://doi.org/10.1038/nbt.3300 (cit. on pp. 10, 12, 26, 47, 48, 56, 72, 78, 138)

Alley, E. C., Khimulya, G., Biswas, S., AlQuraishi, M., & Church, G. M. (2019). Unified rational protein engineering with sequence-based deep representation learning. *Nature Methods*, *16*(12), 1315–1322. https://doi.org/10.1038/s41592-019-0598-1 (cit. on pp. 17, 71)

Almeida, B. P. d., Reiter, F., Pagani, M., & Stark, A. (2021). DeepSTARR predicts enhancer activity from DNA sequence and enables the de novo design of enhancers. *bioRxiv*, 2021.10.05.463203. Retrieved December 9, 2021, from https://www.biorxiv.org/content/10.1101/2021.10.05.463203v1 (cit. on p. 12)

Alser, M., Rotman, J., Deshpande, D., Taraszka, K., Shi, H., Baykal, P. I., Yang, H. T., Xue, V., Knyazev, S., Singer, B. D., Balliu, B., Koslicki, D., Skums, P., Zelikovsky, A., Alkan, C., Mutlu, O., & Mangul, S. (2021). Technology dictates algorithms: Recent developments in read alignment. *Genome Biology*, *22*(1), 249. https://doi.org/10.1186/s13059-021-02443-7 (cit. on pp. 7, 100)

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., & Lipman, D. J. (1990). Basic local alignment search tool. *Journal of Molecular Biology*, *215*(3), 403–410. https://doi.org/10.1016/S0022-2836(05)80360-2 (cit. on pp. 7, 25, 48, 77, 98, 100, 108)

Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., Miller, W., & Lipman, D. J. (1997). Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Research*, *25*(17), 3389–3402. https://doi.org/10.1093/nar/25.17.3389 (cit. on p. 7)

Amarasinghe, S. L., Su, S., Dong, X., Zappia, L., Ritchie, M. E., & Gouil, Q. (2020). Opportunities and challenges in long-read sequencing data analysis. *Genome Biology*, *21*(1), 30. https://doi.org/10.1186/s13059-020-1935-5 (cit. on p. 6)

Amgarten, D., Braga, L. P. P., da Silva, A. M., & Setubal, J. C. (2018). MARVEL, a Tool for Prediction of Bacteriophage Sequences in Metagenomic Bins. *Frontiers in Genetics*, *9*, 304. https://doi.org/10.3389/fgene.2018.00304 (cit. on p. 15)

Amores, G. R., Guazzaroni, M.-E., Arruda, L. M., & Silva-Rocha, R. (2016). Recent Progress on Systems and Synthetic Biology Approaches to Engineer Fungi As Microbial Cell Factories. *Current Genomics*, *17*(2), 85–98. https://doi.org/10.2174/1389202917666151116212255 (cit. on p. 100)

Andersen, K. G., Rambaut, A., Lipkin, W. I., Holmes, E. C., & Garry, R. F. (2020). The proximal origin of SARS-CoV-2. *Nature Medicine*, *26*(4), 450–452. https://doi.org/10.1038/s41591-020-0820-9 (cit. on p. 5)

Andrusch, A., Dabrowski, P. W., Klenner, J., Tausch, S. H., Kohl, C., Osman, A. A., Renard, B. Y., & Nitsche, A. (2018). PAIPline: Pathogen identification in metagenomic and clinical next generation sequencing samples. *Bioinformatics*, *34*(17), i715–i721. https://doi.org/10.1093/bioinformatics/bty595 (cit. on pp. 8, 25, 46, 95, 98)

Angermueller, C., Dohan, D., Belanger, D., Deshpande, R., Murphy, K., & Colwell, L. (2019). Model-based reinforcement learning for biological sequence design. Retrieved December 15, 2021, from https://openreview.net/forum?id=HklxbgBKvr (cit. on p. 17)

Angly, F. E., Willner, D., Prieto-Davó, A., Edwards, R. A., Schmieder, R., Vega-Thurber, R., Antonopoulos, D. A., Barott, K., Cottrell, M. T., Desnues, C., Dinsdale, E. A.,

Furlan, M., Haynes, M., Henn, M. R., Hu, Y., Kirchman, D. L., McDole, T., McPherson, J. D., Meyer, F., … Rohwer, F. (2009). The GAAS Metagenomic Tool and Its Estimations of Viral and Microbial Average Genome Size in Four Major Biomes. *PLOS Computational Biology*, *5*(12), e1000593. https://doi.org/10.1371/journal.pcbi.1000593 (cit. on p. 8)

Anishchenko, I., Pellock, S. J., Chidyausiku, T. M., Ramelot, T. A., Ovchinnikov, S., Hao, J., Bafna, K., Norn, C., Kang, A., Bera, A. K., DiMaio, F., Carter, L., Chow, C. M., Montelione, G. T., & Baker, D. (2021). De novo protein design by deep network hallucination. *Nature*, 1–6. https://doi.org/10.1038/s41586-021-04184-w (cit. on p. 17)

Arango-Argoty, G., Garner, E., Pruden, A., Heath, L. S., Vikesland, P., & Zhang, L. (2018). DeepARG: A deep learning approach for predicting antibiotic resistance genes from metagenomic data. *Microbiome*, *6*(1), 23. https://doi.org/10.1186/s40168-018-0401-z (cit. on p. 13)

Aun, E., Brauer, A., Kisand, V., Tenson, T., & Remm, M. (2018). A k-mer-based method for the identification of phenotype-associated genomic biomarkers and predicting phenotypes of sequenced bacteria. *PLOS Computational Biology*, *14*(10), e1006434. https://doi.org/10.1371/journal.pcbi.1006434 (cit. on p. 13)

Auslander, N., Gussow, A. B., Benler, S., Wolf, Y. I., & Koonin, E. V. (2020). Seeker: Alignment-free identification of bacteriophage genomes by deep learning. *Nucleic Acids Research*, *48*(21), e121. https://doi.org/10.1093/nar/gkaa856 (cit. on pp. 15, 132)

Avsec, Ž., Agarwal, V., Visentin, D., Ledsam, J. R., Grabska-Barwinska, A., Taylor, K. R., Assael, Y., Jumper, J., Kohli, P., & Kelley, D. R. (2021). Effective gene expression prediction from sequence by integrating long-range interactions. *Nature Methods*, *18*(10), 1196–1203. https://doi.org/10.1038/s41592-021-01252-x (cit. on pp. 10, 17, 136)

Avsec, Ž., Weilert, M., Shrikumar, A., Krueger, S., Alexandari, A., Dalal, K., Fropf, R., McAnany, C., Gagneur, J., Kundaje, A., & Zeitlinger, J. (2021). Base-resolution models of transcription-factor binding reveal soft motif syntax. *Nature Genetics*, *53*(3), 354–366. https://doi.org/10.1038/s41588-021-00782-6 (cit. on pp. 10, 12, 17, 47, 57, 134)

Babayan, S. A., Orton, R. J., & Streicker, D. G. (2018). Predicting reservoir hosts and arthropod vectors from evolutionary signatures in RNA virus genomes. *Science*, *362*(6414), 577–580. https://doi.org/10.1126/science.aap9072 (cit. on pp. 14, 47)

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., & Samek, W. (2015). On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation. *PLOS ONE*, *10*(7), e0130140. https://doi.org/10.1371/journal.pone.0130140 (cit. on pp. 12, 48)

Balaji, A., Kille, B., Kappell, A. D., Godbold, G. D., Diep, M., Elworth, R. A. L., Qian, Z., Albin, D., Nasko, D. J., Shah, N., Pop, M., Segarra, S., Ternus, K. L., & Treangen, T. J. (2021). SeqScreen: Accurate and Sensitive Functional Screening of Pathogenic Sequences via Ensemble Learning. *bioRxiv*, 2021.05.02.442344. https://doi.org/10.1101/2021.05.02.442344 (cit. on pp. 17, 91, 92, 100)

Bałazy, S., Miętkiewski, R., Tkaczuk, C., Wegensteiner, R., & Wrzosek, M. (2008). Diversity of acaropathogenic fungi in Poland and other European countries. *Experimental and Applied Acarology*, *46*(1), 53–70. https://doi.org/10.1007/s10493-008-9207-1 (cit. on p. 102)

Bao, Y., Wadden, J., Erb-Downward, J. R., Ranjan, P., Zhou, W., McDonald, T. L., Mills, R. E., Boyle, A. P., Dickson, R. P., Blaauw, D., & Welch, J. D. (2021). SquiggleNet: Real-time, direct classification of nanopore signals. *Genome Biology*, *22*(1), 298. https://doi.org/10.1186/s13059-021-02511-y (cit. on p. 135)

Barash, E., Sal-Man, N., Sabato, S., & Ziv-Ukelson, M. (2018). BacPaCS—bacterial pathogenicity classification via sparse-SVM. *Bioinformatics*. https://doi.org/10.1093/bioinformatics/bty928 (cit. on pp. 13, 26, 29, 33, 37, 40, 42, 99)

Barbara, D. J., & Clewes, E. (2003). Plant pathogenic Verticillium species: How many of them are there? *Molecular Plant Pathology*, *4*(4), 297–305. https://doi.org/10.1046/j.1364-3703.2003.00172.x (cit. on p. 103)

Bartoszewicz, J. M., Genske, U., & Renard, B. Y. (2021a). Deep learning-based real-time detection of novel pathogens during sequencing. *Briefings in Bioinformatics*, (bbab269). https://doi.org/10.1093/bib/bbab269 (cit. on pp. 21, 75, 100, 107, 110–112, 120, 122, 123, 125, 168)

Bartoszewicz, J. M., Genske, U., & Renard, B. Y. (2021b). Real-time pathogenicity prediction during genome sequencing of novel viruses and bacteria. *ICLR 2021 Machine Learning for Preventing and Combating Pandemics Workshop* (cit. on p. 21).

Bartoszewicz, J. M., Genske, U., & Renard, B. Y. (2021c). Real-time pathogenicity prediction during genome sequencing of novel viruses and bacteria. *Machine Learning in Computational Biology 2021* (cit. on p. 21).

Bartoszewicz, J. M., Nasri, F., Nowicka, M., & Renard, B. Y. (2022). Detecting DNA of novel fungal pathogens using ResNets and a curated fungi-hosts data collection. *bioRxiv*. https://doi.org/10.1101/2021.11.30.470625 (cit. on pp. 22, 97)

Bartoszewicz, J. M., Seidel, A., & Renard, B. Y. (2021a). Interpretable detection of novel human viruses from genome sequencing data. *NAR Genomics and Bioinformatics*, *3*(lqab004). https://doi.org/10.1093/nargab/lqab004 (cit. on pp. 20, 45, 78–81, 84, 85, 92–95, 99, 100, 105, 107, 108, 110, 112, 114, 115, 122, 125, 126, 153)

Bartoszewicz, J. M., Seidel, A., & Renard, B. Y. (2021b). Interpretable prediction of the infectious potential of novel viruses. *ICLR 2021 AI for Public Health Workshop* (cit. on p. 20).

Bartoszewicz, J. M., Seidel, A., Rentzsch, R., & Renard, B. Y. (2020). DeePaC: predicting pathogenic potential of novel DNA with reverse-complement neural networks. *Bioinformatics*, *36*(1), 81–89. https://doi.org/10.1093/bioinformatics/btz541 (cit. on pp. 18, 23, 47–49, 51–54, 61, 63, 65, 70, 71, 78–82, 84, 85, 92–94, 99, 100, 105–110, 112, 113, 115, 122, 125, 141, 153)

Basenko, E. Y., Pulman, J. A., Shanmugasundram, A., Harb, O. S., Crouch, K., Starns, D., Warrenfeltz, S., Aurrecoechea, C., Stoeckert, C. J., Kissinger, J. C., Roos, D. S., & Hertz-Fowler, C. (2018). FungiDB: An Integrated Bioinformatic Resource for Fungi and Oomycetes. *Journal of Fungi*, *4*(1), 39. https://doi.org/10.3390/jof4010039 (cit. on p. 99)

Becnel, J. J., & Andreadis, T. G. (2014). Microsporidia in Insects. *Microsporidia* (pp. 521–570). John Wiley & Sons, Ltd. https://doi.org/10.1002/9781118395264.ch21. (Cit. on p. 102)

Beghini, F., McIver, L. J., Blanco-Míguez, A., Dubois, L., Asnicar, F., Maharjan, S., Mailyan, A., Manghi, P., Scholz, M., Thomas, A. M., Valles-Colomer, M., Weingart, G., Zhang, Y., Zolfo, M., Huttenhower, C., Franzosa, E. A., & Segata, N. (2021). Integrating taxonomic, functional, and strain-level profiling of diverse microbial communities with bioBakery 3 (P. Turnbaugh, E. Franco, & C. T. Brown, Eds.). *eLife*, *10*, e65088. https://doi.org/10.7554/eLife.65088 (cit. on p. 9)

Bergner, L. M., Mollentze, N., Orton, R. J., Tello, C., Broos, A., Biek, R., & Streicker, D. G. (2021). Characterizing and Evaluating the Zoonotic Potential of Novel Viruses Discovered in Vampire Bats. *Viruses*, *13*(2), 252. https://doi.org/10.3390/v13020252 (cit. on p. 99)

Bin Jang, H., Bolduc, B., Zablocki, O., Kuhn, J. H., Roux, S., Adriaenssens, E. M., Brister, J. R., Kropinski, A. M., Krupovic, M., Lavigne, R., Turner, D., & Sullivan, M. B. (2019). Taxonomic assignment of uncultivated prokaryotic virus genomes is enabled by gene-sharing networks. *Nature Biotechnology*, *37*(6), 632–639. https://doi.org/10.1038/s41587-019-0100-8 (cit. on p. 15)

Biswas, S., Khimulya, G., Alley, E. C., Esvelt, K. M., & Church, G. M. (2021). Low-N protein engineering with data-efficient deep learning. *Nature Methods*, *18*(4), 389–396. https://doi.org/10.1038/s41592-021-01100-y (cit. on pp. 17, 71)

Blackwell, M. (2011). The fungi: 1, 2, 3 . . . 5.1 million species? *American Journal of Botany*, *98*(3), 426–438. https://doi.org/10.3732/ajb.1000298 (cit. on p. 98)

Boeckaerts, D., Stock, M., Criel, B., Gerstmans, H., De Baets, B., & Briers, Y. (2021). Predicting bacteriophage hosts based on sequences of annotated receptor-binding proteins. *Scientific Reports*, *11*(1), 1467. https://doi.org/10.1038/s41598-021-81063-4 (cit. on p. 14)

Bombassaro, A., Schneider, G. X., Costa, F. F., Leão, A. C. R., Soley, B. S., Medeiros, F., da Silva, N. M., Lima, B. J. F. S., Castro, R. J. A., Bocca, A. L., Baura, V. A., Balsanelli, E., Pankievicz, V. C. S., Hrysay, N. M. C., Scola, R. H., Moreno, L. F., Azevedo, C. M. P. S., Souza, E. M., Gomes, R. R., … Vicente, V. A. (2020). Genomics and Virulence of Fonsecaea pugnacius, Agent of Disseminated Chromoblastomycosis. *Frontiers in Genetics*, *11*, 822. https://doi.org/10.3389/fgene.2020.00822 (cit. on p. 102)

Brady, A., & Salzberg, S. L. (2009). Phymm and PhymmBL: Metagenomic phylogenetic classification with interpolated Markov models. *Nature Methods*, *6*(9), 673–676. https://doi.org/10.1038/nmeth.1358 (cit. on p. 9)

Breitwieser, F. P., Baker, D. N., & Salzberg, S. L. (2018). KrakenUniq: Confident and fast metagenomics classification using unique k-mer counts. *Genome Biology*, *19*(1), 198. https://doi.org/10.1186/s13059-018-1568-0 (cit. on pp. 9, 77, 98)

Breitwieser, F. P., Lu, J., & Salzberg, S. L. (2017). A review of methods and databases for metagenomic classification and assembly. *Briefings in Bioinformatics*, *20*(4), 1125–1136. https://doi.org/10.1093/bib/bbx120 (cit. on pp. 7, 9, 100)

Brendel, W., & Bethge, M. (2019). Approximating CNNs with bag-of-local-features models works surprisingly well on ImageNet. *International Conference on Learning Representations*. https://openreview.net/forum?id=SkfMWhAqYQ (cit. on pp. 42, 70, 95)

Brierley, L., & Fowler, A. (2021). Predicting the animal hosts of coronaviruses from compositional biases of spike protein and whole genome sequences through machine learning. *PLOS Pathogens*, *17*(4), e1009149. https://doi.org/10.1371/journal.ppat.1009149 (cit. on pp. 14, 99)

Brinkerhoff, H., Kang, A. S. W., Liu, J., Aksimentiev, A., & Dekker, C. (2021). Multiple rereads of single proteins at single–amino acid resolution using nanopores. *Science*. https://doi.org/10.1126/science.abl4381 (cit. on p. 136)

Brockhurst, M. A., Chapman, T., King, K. C., Mank, J. E., Paterson, S., & Hurst, G. D. D. (2014). Running with the Red Queen: The role of biotic conflicts in evolution. *Proceedings of the Royal Society B: Biological Sciences*, *281*(1797), 20141382. https://doi.org/10.1098/rspb.2014.1382 (cit. on p. 129)

Brookes, D., Park, H., & Listgarten, J. (2019). Conditioning by adaptive sampling for robust design. *International Conference on Machine Learning*, 773–782. Retrieved June 25, 2020, from http://proceedings.mlr.press/v97/brookes19a.html (cit. on pp. 17, 71, 137)

Brown, G. D., Denning, D. W., Gow, N. A. R., Levitz, S. M., Netea, M. G., & White, T. C. (2012). Hidden killers: Human fungal infections. *Science Translational Medicine*, *4*(165). https://doi.org/10.1126/scitranslmed.3004404 (cit. on pp. 98, 105)

Brown, R. C., Lunter, G., & Hancock, J. (2018). An equivariant bayesian convolutional network predicts recombination hotspots and accurately resolves binding motifs. *Bioinformatics*. https://doi.org/10.1093/bioinformatics/bty964 (cit. on pp. 10, 26)

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., … Amodei, D. (2020). Language Models are Few-Shot Learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in Neural Information Processing Systems* (pp. 1877–1901). Curran Associates, Inc. https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf. (Cit. on p. 15)

Brunner-Mendoza, C., Reyes-Montes, M. d. R., Moonjely, S., Bidochka, M. J., & Toriello, C. (2019). A review on the genus Metarhizium as an entomopathogenic microbial biocontrol agent with emphasis on its use and utility in Mexico. *Biocontrol Science and Technology*, *29*(1), 83–102. https://doi.org/10.1080/09583157.2018.1531111 (cit. on p. 102)

Bryan, R., T., Pinner, R. W., Gaynes, R. P., Peters, C. J., Aguilar, J. R., & Berkelman, R. L. (1994). Addressing emerging infectious disease threats: A prevention strategy for the United States. Executive summary. *MMWR. Recommendations and reports: Morbidity and mortality weekly report. Recommendations and reports*, *43*(RR-5), 1–18 (cit. on p. 3).

Budach, S., & Marsico, A. (2018). Pysster: Classification of biological sequences by learning sequence and structure motifs with convolutional neural networks.

Bioinformatics, *34*(17), 3035–3037. https://doi.org/10.1093/bioinformatics/bty222 (cit. on pp. 10, 26, 31, 130)

Burgess, D. J. (2017). Synthetic Biology: Building a custom eukaryotic genome de novo. *Nature Reviews Genetics*, *18*(5), 274–274. https://doi.org/10.1038/nrg.2017.30 (cit. on p. 100)

Byrd, A. L., Perez-Rogers, J. F., Manimaran, S., Castro-Nallar, E., Toma, I., McCaffrey, T., Siegel, M., Benson, G., Crandall, K. A., & Johnson, W. E. (2014). Clinical PathoScope: Rapid alignment and filtration for accurate pathogen identification in clinical samples using unassembled sequencing data. *BMC Bioinformatics*, *15*(1), 262. https://doi.org/10.1186/1471-2105-15-262 (cit. on p. 8)

Calistri, A., & Palù, G. (2015). Editorial commentary: Unbiased next-generation sequencing and new pathogen discovery: Undeniable advantages and still-existing drawbacks. *Clinical Infectious Diseases: An Official Publication of the Infectious Diseases Society of America*, *60*(6), 889–891. https://doi.org/10.1093/cid/ciu913 (cit. on pp. 5, 6, 24, 46, 76)

Calvignac-Spencer, S., Schulze, J. M., Zickmann, F., & Renard, B. Y. (2014). Clock Rooting Further Demonstrates that Guinea 2014 EBOV is a Member of the Zaïre Lineage. *PLoS Currents*, *6*, ecurrents.outbreaks.c0e035c86d721668a6ad7353f7f6fe86. https://doi.org/10.1371/currents.outbreaks.c0e035c86d721668a6ad7353f7f6fe86 (cit. on pp. 46, 76)

Camacho, C., Coulouris, G., Avagyan, V., Ma, N., Papadopoulos, J., Bealer, K., & Madden, T. L. (2009). BLAST+: Architecture and applications. *BMC Bioinformatics*, *10*(1), 421. https://doi.org/10.1186/1471-2105-10-421 (cit. on pp. 7, 48, 77, 82, 98, 100, 108)

Carter, S. R., & Friedman, R. M. (2015). DNA synthesis and biosecurity: Lessons learned and options for the future [J. Craig Venter Institute]. Retrieved January 27, 2019, from https://www.jcvi.org/dna-synthesis-and-biosecurity. (Cit. on pp. 5, 24)

Casadevall, A., Kontoyiannis, D. P., & Robert, V. (2019). On the emergence of candida auris: Climate change, azoles, swamps, and birds. *MBio*, *10*(4), e01397–19 (cit. on pp. 4, 98, 129).

Castro-Wallace, S. L., Chiu, C. Y., John, K. K., Stahl, S. E., Rubins, K. H., McIntyre, A. B. R., Dworkin, J. P., Lupisella, M. L., Smith, D. J., Botkin, D. J., Stephenson, T. A., Juul, S., Turner, D. J., Izquierdo, F., Federman, S., Stryke, D., Somasekar, S., Alexander, N., Yu, G., … Burton, A. S. (2017). Nanopore DNA Sequencing and Genome Assembly on the International Space Station. *Scientific Reports*, *7*(1), 18022. https://doi.org/10.1038/s41598-017-18364-0 (cit. on p. 6)

CDC. (2019). *Antibiotic resistance threats in the United States, 2019*. U.S. Department of Health; Human Services, CDC. https://stacks.cdc.gov/view/cdc/82532. (Cit. on pp. 4, 98, 105)

Chandler, D., Davidson, G., Pell, J. K., Ball, B. V., Shaw, K., & Sunderland, K. D. (2000). Fungal Biocontrol of Acari. *Biocontrol Science and Technology*, *10*(4), 357–384. https://doi.org/10.1080/09583150050114972 (cit. on p. 102)

Chen, I.-M. A., Chu, K., Palaniappan, K., Pillay, M., Ratner, A., Huang, J., Huntemann, M., Varghese, N., White, J. R., Seshadri, R., Smirnova, T., Kirton, E., Jungbluth, S. P., Woyke, T., Eloe-Fadrosh, E. A., Ivanova, N. N., & Kyrpides, N. C. (2019). IMG/m v.5.0: An integrated data management and comparative analysis system for microbial genomes and microbiomes. *Nucleic Acids Research*, *47*, D666–D677. https://doi.org/10.1093/nar/gky901 (cit. on pp. 27, 78)

Chen, K. M., Cofer, E. M., Zhou, J., & Troyanskaya, O. G. (2019). Selene: A PyTorch-based deep learning library for sequence data. *Nature Methods*, *16*(4), 315–318. https://doi.org/10.1038/s41592-019-0360-8 (cit. on p. 130)

Chen, L., & Capra, J. A. (2020). Learning and interpreting the gene regulatory grammar in a deep learning framework. *PLOS Computational Biology*, *16*(11), e1008334. https://doi.org/10.1371/journal.pcbi.1008334 (cit. on p. 12)

Chen, M. L., Doddi, A., Royer, J., Freschi, L., Schito, M., Ezewudo, M., Kohane, I. S., Beam, A., & Farhat, M. (2019). Beyond multidrug resistance: Leveraging rare variants with machine and statistical learning models in Mycobacterium tuberculosis resistance prediction. *EBioMedicine*, *43*, 356–369. https://doi.org/10.1016/j.ebiom.2019.04.016 (cit. on p. 13)

Choromanski, K., Likhosherstov, V., Dohan, D., Song, X., Gane, A., Sarlos, T., Hawkins, P., Davis, J., Mohiuddin, A., Kaiser, L., Belanger, D., Colwell, L., & Weller, A. (2021). Rethinking Attention with Performers. *arXiv:2009.14794 [cs, stat]*. Retrieved January 18, 2022, from http://arxiv.org/abs/2009.14794 (cit. on pp. 130, 134)

Chowdhary, A., Agarwal, K., & Meis, J. F. (2016). Filamentous Fungi in Respiratory Infections. What Lies Beyond Aspergillosis and Mucormycosis? *PLOS Pathogens*, *12*(4), e1005491. https://doi.org/10.1371/journal.ppat.1005491 (cit. on pp. 98, 100)

Chowdhary, A., Kathuria, S., Agarwal, K., & Meis, J. F. (2014). Recognizing filamentous basidiomycetes as agents of human disease: A review. *Medical Mycology*, *52*(8), 782–797. https://doi.org/10.1093/mmy/myu047 (cit. on p. 102)

Cissé, O. H., Ma, L., Dekker, J. P., Khil, P. P., Youn, J.-H., Brenchley, J. M., Blair, R., Pahar, B., Chabé, M., Van Rompay, K. K. A., Keesler, R., Sukura, A.,

Hirsch, V., Kutty, G., Liu, Y., Peng, L., Chen, J., Song, J., Weissenbacher-Lang, C., … Kovacs, J. A. (2021). Genomic insights into the host specific adaptation of the Pneumocystis genus. *Communications Biology*, *4*(1), 1–14. https://doi.org/10.1038/s42003-021-01799-7 (cit. on p. 102)

Cohen, T., & Welling, M. (2016). Group equivariant convolutional networks. *International Conference on Machine Learning*, 2990–2999. Retrieved January 31, 2019, from http://proceedings.mlr.press/v48/cohenc16.html (cit. on p. 26)

Colombo, A. L., Padovan, A. C. B., & Chaves, G. M. (2011). Current knowledge of Trichosporon spp. and Trichosporonosis. *Clinical Microbiology Reviews*, *24*(4), 682–700. https://doi.org/10.1128/CMR.00003-11 (cit. on p. 102)

Cosentino, S., Larsen, M. V., Aarestrup, F. M., & Lund, O. (2013). PathogenFinder - distinguishing friend from foe using bacterial whole genome sequence data. *PLOS ONE*, *8*(10), e77302. https://doi.org/10.1371/journal.pone.0077302 (cit. on p. 26)

Costa, M. M., Silva, B. A. A. S., Moreira, G. M., & Pfenning, L. H. (2021). Colletotrichum falcatum and Fusarium species induce symptoms of red rot in sugarcane in Brazil. *Plant Pathology*, *70*(8), 1807–1818. https://doi.org/10.1111/ppa.13423 (cit. on p. 103)

Coutinho, I. B. L., Freire, F. C. O., Lima, C. S., Lima, J. S., Gonçalves, F. J. T., Machado, A. R., Silva, A. M. S., & Cardoso, J. E. (2017). Diversity of genus Lasiodiplodia associated with perennial tropical fruit plants in northeastern Brazil. *Plant Pathology*, *66*(1), 90–104. https://doi.org/10.1111/ppa.12565 (cit. on p. 103)

Crooks, G. E., Hon, G., Chandonia, J.-M., & Brenner, S. E. (2004). WebLogo: A sequence logo generator. *Genome Research*, *14*(6), 1188–1190. https://doi.org/10.1101/gr.849004 (cit. on p. 48)

Dadi, T. H., Renard, B. Y., Wieler, L. H., Semmler, T., & Reinert, K. (2017). SLIMM: Species level identification of microorganisms from metagenomes. *PeerJ*, *5*, e3138. https://doi.org/10.7717/peerj.3138 (cit. on p. 8)

Dai, J., Boeke, J. D., Luo, Z., Jiang, S., & Cai, Y. (2020). Sc3.0: Revamping and minimizing the yeast genome. *Genome Biology*, *21*(1), 205. https://doi.org/10.1186/s13059-020-02130-z (cit. on p. 100)

Dean, R., Van Kan, J. A. L., Pretorius, Z. A., Hammond-Kosack, K. E., Di PIEietro, A., Spanu, P. D., Rudd, J. J., Dickman, M., Kahmann, R., Ellis, J., & Foster, G. D. (2012). The Top 10 fungal pathogens in molecular plant pathology. *Molecular Plant Pathology*, *13*(4), 414–430. https://doi.org/10.1111/j.1364-3703.2011.00783.x (cit. on p. 105)

de Hoog, G., Guarro, J., Gené, J., Ahmed, S., Al-Hatmi, A., Figueras, M., & Vitale, R. (2020). *Atlas of clinical fungi, 4th edition*. Hilversum. (Cit. on pp. 103, 104).

DeLano, W. L. et al. (2002). Pymol: An open-source molecular graphics tool. *CCP4 Newsletter on protein crystallography*, *40*(1), 82–92 (cit. on p. 59).

Dellière, S., Rivero-Menendez, O., Gautier, C., Garcia-Hermoso, D., Alastruey-Izquierdo, A., & Alanio, A. (2020). Emerging mould infections: Get prepared to meet unexpected fungi in your patient. *Medical Mycology*, *58*(2), 156–162. https://doi.org/10.1093/mmy/myz039 (cit. on p. 102)

Deneke, C., Rentzsch, R., & Renard, B. Y. (2017). PaPrBaG: A machine learning approach for the detection of novel pathogens from NGS data. *Scientific Reports*, *7*, 39194. https://doi.org/10.1038/srep39194 (cit. on pp. 7, 9, 11, 13, 17, 18, 24, 26, 27, 32, 35, 37, 40, 41, 51, 54, 61, 77, 78, 82, 84, 93–95, 99, 100, 107, 115, 125, 126, 130, 133)

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186. https://doi.org/10.18653/v1/N19-1423 (cit. on p. 15)

Diggans, J., & Leproust, E. (2019). Next Steps for Access to Safe, Secure DNA Synthesis. *Frontiers in Bioengineering and Biotechnology*, *7*. https://doi.org/10.3389/fbioe.2019.00086 (cit. on pp. 5, 17, 71, 91, 92, 100)

Dilthey, A. T., Meyer, S. A., & Kaasch, A. J. (2020). Ultraplexing: Increasing the efficiency of long-read sequencing for hybrid assembly with k-mer-based multiplexing. *Genome Biology*, *21*(1), 68. https://doi.org/10.1186/s13059-020-01974-9 (cit. on p. 85)

Doehlemann, G., Ökmen, B., Zhu, W., & Sharon, A. (2017). Plant Pathogenic Fungi. *Microbiology Spectrum*, *5*(1). https://doi.org/10.1128/microbiolspec.FUNK-0023-2016 (cit. on p. 103)

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2021). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv:2010.11929 [cs]*. Retrieved December 14, 2021, from http://arxiv.org/abs/2010.11929 (cit. on p. 15)

Drouin, A., Giguère, S., Déraspe, M., Marchand, M., Tyers, M., Loo, V. G., Bourgault, A.-M., Laviolette, F., & Corbeil, J. (2016). Predictive computational phenotyping and biomarker discovery using reference-free genome comparisons. *BMC*

Genomics, *17*(1), 754. https://doi.org/10.1186/s12864-016-2889-6 (cit. on p. 13)

Drouin, A., Letarte, G., Raymond, F., Marchand, M., Corbeil, J., & Laviolette, F. (2019). Interpretable genotype-to-phenotype classifiers with performance guarantees. *Scientific Reports*, *9*(1), 4071. https://doi.org/10.1038/s41598-019-40561-2 (cit. on p. 13)

Edgar, R. C., Taylor, J., Lin, V., Altman, T., Barbera, P., Meleshko, D., Lohr, D., Novakovsky, G., Buchfink, B., Al-Shayeb, B., Banfield, J. F., de la Peña, M., Korobeynikov, A., Chikhi, R., & Babaian, A. (2022). Petabase-scale sequence alignment catalyses viral discovery. *Nature*, 1–6. https://doi.org/10.1038/s41586-021-04332-2 (cit. on p. 129)

Edwards, H. S., Krishnakumar, R., Sinha, A., Bird, S. W., Patel, K. D., & Bartsch, M. S. (2019). Real-Time Selective Sequencing with RUBRIC: Read Until with Basecall and Reference-Informed Criteria. *Scientific Reports*, *9*(1), 11475. https://doi.org/10.1038/s41598-019-47857-3 (cit. on p. 135)

Edwards, R. A., McNair, K., Faust, K., Raes, J., & Dutilh, B. E. (2016). Computational approaches to predict bacteriophage-host relationships. *FEMS microbiology reviews*, *40*(2), 258–272. https://doi.org/10.1093/femsre/fuv048 (cit. on pp. 14, 46, 56)

Eid, F.-E., ElHefnawi, M., & Heath, L. S. (2016). DeNovo: Virus-host sequence-based protein–protein interaction prediction. *Bioinformatics*, *32*(8), 1144–1150. https://doi.org/10.1093/bioinformatics/btv737 (cit. on p. 15)

Elnaggar, A., Heinzinger, M., Dallago, C., Rehawi, G., Wang, Y., Jones, L., Gibbs, T., Feher, T., Angerer, C., Steinegger, M., Bhowmik, D., & Rost, B. (2021). ProtTrans: Towards Cracking the Language of Life's Code Through Self-Supervised Learning. *bioRxiv*, 2020.07.12.199554. https://doi.org/10.1101/2020.07.12.199554 (cit. on p. 15)

Eng, C. L., Tong, J. C., & Tan, T. W. (2014). Predicting host tropism of influenza a virus proteins using random forest. *BMC Medical Genomics*, *7*(3), S1. https://doi.org/10.1186/1755-8794-7-S3-S1 (cit. on pp. 14, 46)

Eraslan, G., Avsec, Ž., Gagneur, J., & Theis, F. J. (2019). Deep learning: New computational modelling techniques for genomics. *Nature Reviews Genetics*, *20*(7), 389–403. https://doi.org/10.1038/s41576-019-0122-6 (cit. on pp. 10, 13, 48, 78)

Evans, H. C., Elliot, S. L., & Hughes, D. P. (2011). Hidden Diversity Behind the Zombie-Ant Fungus Ophiocordyceps unilateralis: Four New Species Described from Carpenter Ants in Minas Gerais, Brazil. *PLOS ONE*, *6*(3), e17024. https://doi.org/10.1371/journal.pone.0017024 (cit. on p. 102)

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*(6), giz066. https://doi.org/10.1093/gigascience/giz066 (cit. on pp. 15, 132)

Farr, D. F., & Rossman, A. Y. (2021). Fungal databases. https://nt.ars-grin.gov/fungaldatabases/. (Cit. on pp. 98, 103)

Feldbauer, R., Schulz, F., Horn, M., & Rattei, T. (2015). Prediction of microbial phenotypes based on comparative genomics. *BMC Bioinformatics*, *16*(14), S1. https://doi.org/10.1186/1471-2105-16-S14-S1 (cit. on p. 13)

Fenner, F. (1982). Global Eradication of Smallpox. *Reviews of Infectious Diseases*, *4*(5), 916–930. https://doi.org/10.1093/clinids/4.5.916 (cit. on p. 129)

Fischer, M., Strauch, B., & Renard, B. Y. (2017). Abundance estimation and differential testing on strain level in metagenomics data. *Bioinformatics (Oxford, England)*, *33*(14), i124–i132. https://doi.org/10.1093/bioinformatics/btx237 (cit. on p. 8)

Fleming, A. (1929). On the Antibacterial Action of Cultures of a Penicillium, with Special Reference to their Use in the Isolation of B. influenzæ. *British journal of experimental pathology*, *10*(3), 226–236. Retrieved December 20, 2021, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2048009/ (cit. on p. 129)

Frank, C., Werber, D., Cramer, J. P., Askar, M., Faber, M., an der Heiden, M., Bernard, H., Fruth, A., Prager, R., Spode, A., Wadl, M., Zoufaly, A., Jordan, S., Kemper, M. J., Follin, P., Müller, L., King, L. A., Rosner, B., Buchholz, U., … HUS Investigation Team. (2011). Epidemic profile of shiga-toxin-producing escherichia coli o104:h4 outbreak in germany. *The New England Journal of Medicine*, *365*(19), 1771–1780. https://doi.org/10.1056/NEJMoa1106483 (cit. on pp. 4, 23, 27, 76)

Franzen, C., & Müller, A. (2001). Microsporidiosis: Human diseases and diagnosis. *Microbes and Infection*, *3*(5), 389–400. https://doi.org/10.1016/s1286-4579(01)01395-8 (cit. on p. 102)

Freitas, T. A. K., Li, P.-E., Scholz, M. B., & Chain, P. S. G. (2015). Accurate read-based metagenome characterization using a hierarchical suite of unique signatures. *Nucleic Acids Research*, *43*(10), e69. https://doi.org/10.1093/nar/gkv180 (cit. on p. 9)

Gałan, W., Bąk, M., & Jakubowska, M. (2019). Host taxon predictor - a tool for predicting taxon of the host of a newly discovered virus. *Scientific Reports*, *9*(1), 3436. https://doi.org/10.1038/s41598-019-39847-2 (cit. on pp. 14, 47, 100)

Galiez, C., Siebert, M., Enault, F., Vincent, J., & Söding, J. (2017). WIsH: Who is the host? predicting prokaryotic hosts from metagenomic phage contigs. *Bioinfor-*

matics, *33*(19), 3113–3114. https://doi.org/10.1093/bioinformatics/btx383 (cit. on p. 14)

Garcia-Solache, M. A., & Casadevall, A. (2010). Global warming will bring new fungal diseases for mammals. *mBio*, *1*(1), e00061–10. https://doi.org/10.1128/mBio.00061-10 (cit. on pp. 4, 124, 129)

Genske, U. (2019). *Convolutional and long short-term memory neural networks for real-time pathogenic potential prediction from NGS reads* (Bachelor's thesis). Free University of Berlin. (Cit. on p. 21).

Gerson, U., Gafni, A., Paz, Z., & Sztejnberg, A. (2008). A tale of three acaropathogenic fungi in Israel: Hirsutella, Meira and Acaromyces. *Experimental and Applied Acarology*, *46*(1), 183–194. https://doi.org/10.1007/s10493-008-9202-6 (cit. on p. 102)

Gligorijević, V., Barot, M., & Bonneau, R. (2018). deepNF: Deep network fusion for protein function prediction. *Bioinformatics*, *34*(22), 3873–3881. https://doi.org/10.1093/bioinformatics/bty440 (cit. on p. 13)

Gligorijević, V., Renfrew, P. D., Kosciolek, T., Leman, J. K., Berenberg, D., Vatanen, T., Chandler, C., Taylor, B. C., Fisk, I. M., Vlamakis, H., Xavier, R. J., Knight, R., Cho, K., & Bonneau, R. (2021). Structure-based protein function prediction using graph convolutional networks. *Nature Communications*, *12*(1), 3168. https://doi.org/10.1038/s41467-021-23303-9 (cit. on p. 13)

Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, 249–256. Retrieved March 3, 2022, from https://proceedings.mlr.press/v9/glorot10a.html (cit. on p. 31)

Goodwin, S., McPherson, J. D., & McCombie, W. R. (2016). Coming of age: Ten years of next-generation sequencing technologies. *Nature Reviews Genetics*, *17*(6), 333–351. https://doi.org/10.1038/nrg.2016.49 (cit. on p. 6)

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., Penzar, D., Perlman, S., Poon, L. L. M., Samborskiy, D. V., Sidorov, I. A., Sola, I., Ziebuhr, J., & Coronaviridae Study Group of the International Committee on Taxonomy of Viruses. (2020). The species Severe acute respiratory syndrome-related coronavirus : Classifying 2019-nCoV and naming it SARS-CoV-2. *Nature Microbiology*, *5*(4), 536–544. https://doi.org/10.1038/s41564-020-0695-z (cit. on p. 52)

Greenside, P., Shimko, T., Fordyce, P., & Kundaje, A. (2018). Discovering epistatic feature interactions from neural network models of regulatory DNA sequences.

Bioinformatics, *34*(17), i629–i637. https://doi.org/10.1093/bioinformatics/bty575 (cit. on pp. 10, 26, 48)

Grimm, S. (2021). Understanding. In E. N. Zalta (Ed.), *The Stanford Encyclopedia of Philosophy* (Summer 2021). Metaphysics Research Lab, Stanford University. Retrieved December 9, 2021, from https://plato.stanford.edu/archives/sum2021/entries/understanding/. (Cit. on p. 11)

Gröschel, M. I., Owens, M., Freschi, L., Vargas, R., Marin, M. G., Phelan, J., Iqbal, Z., Dixit, A., & Farhat, M. R. (2021). GenTB: A user-friendly genome-based predictor for tuberculosis resistance powered by machine learning. *Genome Medicine*, *13*(1), 138. https://doi.org/10.1186/s13073-021-00953-4 (cit. on p. 13)

Grüning, B., Dale, R., Sjödin, A., Chapman, B. A., Rowe, J., Tomkins-Tinch, C. H., Valieris, R., & Köster, J. (2018). Bioconda: Sustainable and comprehensive software distribution for the life sciences. *Nature Methods*, *15*(7), 475–476. https://doi.org/10.1038/s41592-018-0046-7 (cit. on pp. 73, 96)

Guo, H., Hu, B.-J., Yang, X.-L., Zeng, L.-P., Li, B., Ouyang, S., & Shi, Z.-L. (2020). Evolutionary Arms Race between Virus and Host Drives Genetic Diversity in Bat Severe Acute Respiratory Syndrome-Related Coronavirus Spike Genes. *Journal of Virology*. https://doi.org/10.1128/JVI.00902-20 (cit. on p. 129)

Guo, J., Bolduc, B., Zayed, A. A., Varsani, A., Dominguez-Huerta, G., Delmont, T. O., Pratama, A. A., Gazitúa, M. C., Vik, D., Sullivan, M. B., & Roux, S. (2021). VirSorter2: A multi-classifier, expert-guided approach to detect diverse DNA and RNA viruses. *Microbiome*, *9*(1), 37. https://doi.org/10.1186/s40168-020-00990-y (cit. on p. 15)

Guo, Q., Li, M., Wang, C., Guo, J., Jiang, X., Tan, J., Wu, S., Wang, P., Xiao, T., Zhou, M., Fang, Z., Xiao, Y., & Zhu, H. (2021). Predicting Hosts Based on Early SARS-CoV-2 Samples and Analyzing Later World-wide Pandemic in 2020. *bioRxiv*, 2021.03.21.436312. https://doi.org/10.1101/2021.03.21.436312 (cit. on pp. 14, 100, 136)

Guo, Q., Li, M., Wang, C., Wang, P., Fang, Z., Tan, J., Wu, S., Xiao, Y., & Zhu, H. (2020). Host and infectivity prediction of Wuhan 2019 novel coronavirus using deep learning algorithm. *bioRxiv*, 2020.01.21.914044. https://doi.org/10.1101/2020.01.21.914044 (cit. on pp. 14, 78, 136)

Gupta, A., Kapil, R., Dhakan, D. B., & Sharma, V. K. (2014). MP3: A Software Tool for the Prediction of Pathogenic Proteins in Genomic and Metagenomic Data. *PLOS ONE*, *9*(4), e93907. https://doi.org/10.1371/journal.pone.0093907 (cit. on p. 13)

Gupta, A., & Kundaje, A. (2019). Targeted optimization of regulatory DNA sequences with neural editing architectures. *bioRxiv*, 714402. https://doi.org/10.1101/714402 (cit. on pp. 17, 71, 137)

Gupta, A., & Zou, J. (2019). Feedback GAN for DNA optimizes protein functions. *Nature Machine Intelligence*, *1*(2), 105–111. https://doi.org/10.1038/s42256-019-0017-4 (cit. on pp. 17, 71, 137)

Gurung, S., Short, D. P. G., Hu, X., Sandoya, G. V., Hayes, R. J., Koike, S. T., & Subbarao, K. V. (2015). Host Range of Verticillium isaacii and Verticillium klebahnii from Artichoke, Spinach, and Lettuce. *Plant Disease*, *99*(7), 933–938. https://doi.org/10.1094/PDIS-12-14-1307-RE (cit. on p. 103)

Han, B., & Weiss, L. M. (2017). Microsporidia: Obligate Intracellular Pathogens Within the Fungal Kingdom. *Microbiology Spectrum*, *5*(2). https://doi.org/10.1128/microbiolspec.FUNK-0018-2016 (cit. on p. 102)

Hawkins-Hooker, A., Depardieu, F., Baur, S., Couairon, G., Chen, A., & Bikard, D. (2021). Generating functional protein variants with variational autoencoders. *PLOS Computational Biology*, *17*(2), e1008736. https://doi.org/10.1371/journal.pcbi.1008736 (cit. on p. 137)

Hawksworth, D. L. (2001). The magnitude of fungal diversity: The 1.5 million species estimate revisited. *Mycological Research*, *105*(12), 1422–1432. https://doi.org/10.1017/S0953756201004725 (cit. on p. 98)

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778. https://doi.org/10.1109/CVPR.2016.90 (cit. on pp. 80, 133)

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. *arXiv:1502.01852 [cs]*. Retrieved January 27, 2019, from http://arxiv.org/abs/1502.01852 (cit. on p. 31)

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., Rimmelzwaan, G. F., Osterhaus, A. D. M. E., & Fouchier, R. A. M. (2012). Airborne Transmission of Influenza A/H5n1 Virus Between Ferrets. *Science*, *336*(6088), 1534–1541. https://doi.org/10.1126/science.1213362 (cit. on pp. 5, 46, 91)

Hie, B., Zhong, E. D., Berger, B., & Bryson, B. (2021). Learning the language of viral evolution and escape. *Science*. https://doi.org/10.1126/science.abd7331 (cit. on pp. 15, 137)

Hoarfrost, A., Aptekmann, A., Farfañuk, G., & Bromberg, Y. (2020). Shedding Light on Microbial Dark Matter with A Universal Language of Life, 2020.12.23.424215.

Retrieved December 8, 2021, from https://www.biorxiv.org/content/10.1101/2020.12.23.424215v2 (cit. on pp. 15, 134)

Hockenberry, A. J., & Wilke, C. O. (2021). BACPHLIP: Predicting bacteriophage lifestyle from conserved protein domains. *bioRxiv*, 2020.05.13.094805. https://doi.org/10.1101/2020.05.13.094805 (cit. on p. 14)

Holmes, E. C., Goldstein, S. A., Rasmussen, A. L., Robertson, D. L., Crits-Christoph, A., Wertheim, J. O., Anthony, S. J., Barclay, W. S., Boni, M. F., Doherty, P. C., Farrar, J., Geoghegan, J. L., Jiang, X., Leibowitz, J. L., Neil, S. J. D., Skern, T., Weiss, S. R., Worobey, M., Andersen, K. G., … Rambaut, A. (2021). The Origins of SARS-CoV-2: A Critical Review. *Cell*, *0*(0). https://doi.org/10.1016/j.cell.2021.08.017 (cit. on pp. 4, 5)

Holtgrewe, M. (2010). Mason – a read simulator for second generation sequencing data. *Technical Report FU Berlin*. Retrieved January 27, 2019, from http://publications.imp.fu-berlin.de/962/ (cit. on pp. 28, 51, 105)

Hong, C., Manimaran, S., Shen, Y., Perez-Rogers, J. F., Byrd, A. L., Castro-Nallar, E., Crandall, K. A., & Johnson, W. E. (2014). PathoScope 2.0: A complete computational framework for strain identification in environmental or clinical sequencing samples. *Microbiome*, *2*(1), 33. https://doi.org/10.1186/2049-2618-2-33 (cit. on pp. 8, 25, 32, 77, 98)

Hu, W., Ran, Y., Zhuang, K., Lama, J., & Zhang, C. (2015). Alternaria arborescens Infection in a Healthy Individual and Literature Review of Cutaneous Alternariosis. *Mycopathologia*, *179*(1), 147–152. https://doi.org/10.1007/s11046-014-9822-9 (cit. on p. 102)

Huseyin, C. E., O'Toole, P. W., Cotter, P. D., & Scanlan, P. D. (2017). Forgotten fungi—the gut mycobiome in human health and disease. *FEMS Microbiology Reviews*, *41*(4), 479–511. https://doi.org/10.1093/femsre/fuw047 (cit. on pp. 98, 124)

Huson, D. H., Auch, A. F., Qi, J., & Schuster, S. C. (2007). MEGAN analysis of metagenomic data. *Genome Research*, *17*(3), 377–386. https://doi.org/10.1101/gr.5969107 (cit. on p. 8)

Imai, M., Watanabe, T., Hatta, M., Das, S. C., Ozawa, M., Shinya, K., Zhong, G., Hanson, A., Katsura, H., Watanabe, S., Li, C., Kawakami, E., Yamada, S., Kiso, M., Suzuki, Y., Maher, E. A., Neumann, G., & Kawaoka, Y. (2012). Experimental adaptation of an influenza H5 HA confers respiratory droplet transmission to a reassortant H5 HA/H1n1 virus in ferrets. *Nature*, *486*(7403), 420–428. https://doi.org/10.1038/nature10831 (cit. on pp. 5, 46, 91)

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv:1502.03167 [cs]*. Retrieved January 27, 2019, from http://arxiv.org/abs/1502.03167 (cit. on pp. 31, 140)

Jančič, S., Nguyen, H. D. T., Frisvad, J. C., Zalar, P., Schroers, H.-J., Seifert, K. A., & Gunde-Cimerman, N. (2015). A Taxonomic Revision of the Wallemia sebi Species Complex. *PLOS ONE*, *10*(5), e0125933. https://doi.org/10.1371/journal.pone.0125933 (cit. on p. 102)

Jha, A., K. Aicher, J., R. Gazzara, M., Singh, D., & Barash, Y. (2020). Enhanced Integrated Gradients: Improving interpretability of deep learning models using splicing codes as a case study. *Genome Biology*, *21*(1), 149. https://doi.org/10.1186/s13059-020-02055-7 (cit. on pp. 12, 48)

Ji, Y., Zhou, Z., Liu, H., & Davuluri, R. V. (2021). DNABERT: Pre-trained Bidirectional Encoder Representations from Transformers model for DNA-language in genome. *Bioinformatics*, *37*(15), 2112–2120. https://doi.org/10.1093/bioinformatics/btab083 (cit. on p. 11)

Jiang, H., An, L., Lin, S. M., Feng, G., & Qiu, Y. (2012). A Statistical Framework for Accurate Taxonomic Assignment of Metagenomic Sequencing Reads. *PLOS ONE*, *7*(10), e46450. https://doi.org/10.1371/journal.pone.0046450 (cit. on p. 8)

Jones, D. R., & Baker, R. H. A. (2007). Introductions of non-native plant pathogens into Great Britain, 1970–2004. *Plant Pathology*, *56*(5), 891–910. https://doi.org/10.1111/j.1365-3059.2007.01619.x (cit. on p. 103)

Jones, K. E., Patel, N. G., Levy, M. A., Storeygard, A., Balk, D., Gittleman, J. L., & Daszak, P. (2008). Global trends in emerging infectious diseases. *Nature*, *451*(7181), 990–993. https://doi.org/10.1038/nature06536 (cit. on pp. 3, 4, 129, 136)

Kapoor, A., Simmonds, P., Lipkin, W. I., Zaidi, S., & Delwart, E. (2010). Use of Nucleotide Composition Analysis To Infer Hosts for Three Novel Picorna-Like Viruses. *Journal of Virology*, *84*(19), 10322–10328. https://doi.org/10.1128/JVI.00601-10 (cit. on p. 14)

Karagöz, M. A., & Nalbantoglu, O. U. (2021). Taxonomic classification of metagenomic sequences from Relative Abundance Index profiles using deep learning. *Biomedical Signal Processing and Control*, *67*, 102539. https://doi.org/10.1016/j.bspc.2021.102539 (cit. on p. 9)

Kelley, D. R. (2020). Cross-species regulatory sequence activity prediction. *PLOS Computational Biology*, *16*(7), e1008050. https://doi.org/10.1371/journal.pcbi.1008050 (cit. on pp. 10, 17, 134)

Kelley, D. R., Reshef, Y. A., Bileschi, M., Belanger, D., McLean, C. Y., & Snoek, J. (2018). Sequential regulatory activity prediction across chromosomes with convolutional neural networks. *Genome Research*, *28*(5), 739–750. https://doi.org/10.1101/gr.227819.117 (cit. on p. 10)

Kelley, D. R., Snoek, J., & Rinn, J. L. (2016). Basset: Learning the regulatory code of the accessible genome with deep convolutional neural networks. *Genome Research*, *26*(7), 990–999. https://doi.org/10.1101/gr.200535.115 (cit. on pp. 10, 26, 47, 138)

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*(1), 90. https://doi.org/10.1186/s40168-020-00867-0 (cit. on p. 15)

Kim, D., Song, L., Breitwieser, F. P., & Salzberg, S. L. (2016). Centrifuge: Rapid and sensitive classification of metagenomic sequences. *Genome Research*. https://doi.org/10.1101/gr.210641.116 (cit. on p. 8)

Kim, S. W., Jung, J. H., Lamsal, K., Kim, Y. S., Min, J. S., & Lee, Y. S. (2012). Antifungal Effects of Silver Nanoparticles (AgNPs) against Various Plant Pathogenic Fungi. *Mycobiology*, *40*(1), 53–58. https://doi.org/10.5941/MYCO.2012.40.1.053 (cit. on p. 103)

Kim, W., Cavinder, B., Proctor, R. H., O'Donnell, K., Townsend, J. P., & Trail, F. (2019). Comparative Genomics and Transcriptomics During Sexual Development Gives Insight Into the Life History of the Cosmopolitan Fungus Fusarium neocosmosporiellum. *Frontiers in Microbiology*, *10*, 1247. https://doi.org/10.3389/fmicb.2019.01247 (cit. on p. 103)

King, A. M. Q., Adams, M. J., Carstens, E. B., & Lefkowitz, E. J. (Eds.). (2012). *Virus Taxonomy: Ninth Report of the International Committee on Taxonomy of Viruses*. Academic Press. (Cit. on pp. 49, 52).

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv:1412.6980 [cs]*. Retrieved January 27, 2019, from http://arxiv.org/abs/1412.6980 (cit. on pp. 31, 139)

Kjærbølling, I., Vesth, T. C., Frisvad, J. C., Nybo, J. L., Theobald, S., Kuo, A., Bowyer, P., Matsuda, Y., Mondo, S., Lyhne, E. K., Kogle, M. E., Clum, A., Lipzen, A., Salamov, A., Ngan, C. Y., Daum, C., Chiniquy, J., Barry, K., LaButti, K., … Andersen, M. R. (2018). Linking secondary metabolites to gene clusters through genome sequencing of six diverse Aspergillus species. *Proceedings of the National Academy of Sciences*, *115*(4), E753–E761. https://doi.org/10.1073/pnas.1715954115 (cit. on p. 102)

Köhler, J. R., Casadevall, A., & Perfect, J. (2015). The Spectrum of Fungi That Infects Humans. *Cold Spring Harbor Perspectives in Medicine*, *5*(1), a019273. https://doi.org/10.1101/cshperspect.a019273 (cit. on p. 102)

Kopp, W., Monti, R., Tamburrini, A., Ohler, U., & Akalin, A. (2020). Deep learning for genomics using Janggu. *Nature Communications*, *11*(1), 3488. https://doi.org/10.1038/s41467-020-17155-y (cit. on p. 130)

Kopp, W., & Schulte-Sasse, R. (2017). Unsupervised learning of DNA sequence features using a convolutional restricted boltzmann machine. *bioRxiv*, *183095*. https://doi.org/10.1101/183095 (cit. on p. 26)

Kovaka, S., Fan, Y., Ni, B., Timp, W., & Schatz, M. C. (2021). Targeted nanopore sequencing by real-time mapping of raw electrical signal with UNCALLED. *Nature Biotechnology*, *39*(4), 431–441. https://doi.org/10.1038/s41587-020-0731-9 (cit. on p. 135)

Kulmanov, M., & Hoehndorf, R. (2020). DeepGOPlus: Improved protein function prediction from sequence. *Bioinformatics*, *36*(2), 422–429. https://doi.org/10.1093/bioinformatics/btz595 (cit. on p. 13)

Kwon-Chung, K. J., Bennett, J. E., Wickes, B. L., Meyer, W., Cuomo, C. A., Wollenburg, K. R., Bicanic, T. A., Castañeda, E., Chang, Y. C., Chen, J., Cogliati, M., Dromer, F., Ellis, D., Filler, S. G., Fisher, M. C., Harrison, T. S., Holland, S. M., Kohno, S., Kronstad, J. W., … Casadevall, A. (2017). The Case for Adopting the "Species Complex" Nomenclature for the Etiologic Agents of Cryptococcosis. *mSphere*, *2*(1), e00357–16. https://doi.org/10.1128/mSphere.00357-16 (cit. on p. 102)

Lanchantin, J., Singh, R., Lin, Z., & Qi, Y. (2016). Deep Motif: Visualizing Genomic Sequence Classifications. *CoRR*, *abs/1605.01133*. http://arxiv.org/abs/1605.01133 (cit. on pp. 12, 48)

Lanchantin, J., Singh, R., Wang, B., & Qi, Y. (2017). Deep motif dashboard: Visualizing and understanding genomic sequences using deep neural networks. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, *22*, 254–265. https://doi.org/10.1142/9789813207813_0025 (cit. on pp. 12, 48)

Lanchantin, J., Weingarten, T., Sekhon, A., Miller, C., & Qi, Y. (2021). Transfer Learning for Predicting Virus-Host Protein Interactions for Novel Virus Sequences. *bioRxiv*, 2020.12.14.422772. Retrieved December 12, 2021, from https://www.biorxiv.org/content/10.1101/2020.12.14.422772v2 (cit. on p. 15)

Langmead, B., & Salzberg, S. L. (2012). Fast gapped-read alignment with bowtie 2. *Nature methods*, *9*(4), 357–359. https://doi.org/10.1038/nmeth.1923 (cit. on pp. 7, 25, 32, 54, 77, 95, 98)

Latorre-Pérez, A., Pascual, J., Porcar, M., & Vilanova, C. (2020). A lab in the field: Applications of real-time, in situ metagenomic sequencing. *Biology Methods & Protocols*, *5*(1), bpaa016. https://doi.org/10.1093/biomethods/bpaa016 (cit. on p. 6)

Lecuit, M., & Eloit, M. (2014). The diagnosis of infectious diseases by whole genome next generation sequencing: A new era is opening. *Frontiers in Cellular and Infection Microbiology*, *4*, 25. https://doi.org/10.3389/fcimb.2014.00025 (cit. on pp. 5, 6, 24, 46, 76)

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539 (cit. on pp. 10, 26)

Lederberg, J., Shope, R. E., & Oaks, S. C. (Eds.). (1992). *Emerging Infections: Microbial Threats to Health in the United States*. The National Academies Press. https://doi.org/10.17226/2008. (Cit. on p. 3)

Leendertz, S. A. J., Gogarten, J. F., Düx, A., Calvignac-Spencer, S., & Leendertz, F. H. (2016). Assessing the evidence supporting fruit bats as the primary reservoirs for ebola viruses. *EcoHealth*, *13*(1), 18–25. https://doi.org/10.1007/s10393-015-1053-0 (cit. on p. 46)

Lees, J. A., Galardini, M., Bentley, S. D., Weiser, J. N., & Corander, J. (2018). Pyseer: A comprehensive tool for microbial pangenome-wide association studies. *Bioinformatics*, *34*(24), 4310–4312. https://doi.org/10.1093/bioinformatics/bty539 (cit. on p. 13)

Lees, J. A., Mai, T. T., Galardini, M., Wheeler, N. E., Horsfield, S. T., Parkhill, J., & Corander, J. (2020). Improved Prediction of Bacterial Genotype-Phenotype Associations Using Interpretable Pangenome-Spanning Regressions. *mBio*, *11*(4), e01344–20. https://doi.org/10.1128/mBio.01344-20 (cit. on p. 13)

Lees, J. A., Vehkala, M., Välimäki, N., Harris, S. R., Chewapreecha, C., Croucher, N. J., Marttinen, P., Davies, M. R., Steer, A. C., Tong, S. Y. C., Honkela, A., Parkhill, J., Bentley, S. D., & Corander, J. (2016). Sequence element enrichment analysis to determine the genetic basis of bacterial phenotypes. *Nature Communications*, *7*(1), 12797. https://doi.org/10.1038/ncomms12797 (cit. on p. 13)

Lefkowitz, E. J., Dempsey, D. M., Hendrickson, R. C., Orton, R. J., Siddell, S. G., & Smith, D. B. (2018). Virus taxonomy: The database of the International Committee on Taxonomy of Viruses (ICTV). *Nucleic Acids Research*, *46*(D1), D708–D717. https://doi.org/10.1093/nar/gkx932 (cit. on pp. 49, 52)

Leite, D. M. C., Brochet, X., Resch, G., Que, Y.-A., Neves, A., & Peña-Reyes, C. (2018). Computational prediction of inter-species relationships through omics

data analysis and machine learning. *BMC Bioinformatics*, *19*(14), 420. https://doi.org/10.1186/s12859-018-2388-7 (cit. on p. 14)

Leite, D. M. C., Lopez, J. F., Brochet, X., Barreto-Sanz, M., Que, Y.-A., Resch, G., & Peña-Reyes, C. (2018). Exploration of multiclass and one-class learning methods for prediction of phage-bacteria interaction at strain level. *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 1818–1825. https://doi.org/10.1109/BIBM.2018.8621433 (cit. on p. 14)

Leonhardt, S., Büttner, E., Gebauer, A. M., Hofrichter, M., & Kellner, H. (2018). Draft Genome Sequence of the Sordariomycete Lecythophora (Coniochaeta) hoffmannii CBS 245.38. *Genome Announcements*, *6*(7), e01510–17. https://doi.org/10.1128/genomeA.01510-17 (cit. on pp. 102, 103)

Levy, S. B., & Marshall, B. (2004). Antibacterial resistance worldwide: Causes, challenges and responses. *Nature Medicine*, *10*(12), S122–S129. https://doi.org/10.1038/nm1145 (cit. on pp. 13, 129)

Li, F. (2016). Structure, Function, and Evolution of Coronavirus Spike Proteins. *Annual Review of Virology*, *3*(1), 237–261. https://doi.org/10.1146/annurev-virology-110615-042301 (cit. on p. 68)

Li, H., & Sun, F. (2018). Comparative studies of alignment, alignment-free and SVM based approaches for predicting the hosts of viruses based on viral sequences. *Scientific Reports*, *8*(1), 10032. https://doi.org/10.1038/s41598-018-28308-x (cit. on pp. 14, 47)

Li, H. (2013). Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. *arXiv:1303.3997 [q-bio]*. Retrieved December 6, 2021, from http://arxiv.org/abs/1303.3997 (cit. on p. 7)

Li, H. (2018). Minimap2: Pairwise alignment for nucleotide sequences. *Bioinformatics*, *34*(18), 3094–3100. https://doi.org/10.1093/bioinformatics/bty191 (cit. on pp. 7, 84, 98)

Li, H., & Durbin, R. (2010). Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics*, *26*(5), 589–595. https://doi.org/10.1093/bioinformatics/btp698 (cit. on pp. 7, 25, 54, 77, 98)

Li, M., Wang, Y., Li, F., Zhao, Y., Liu, M., Zhang, S., Bin, Y., Smith, A. I., Webb, G., Li, J., Song, J., & Xia, J. (2020). A Deep Learning-Based Method for Identification of Bacteriophage-Host Interaction. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1–1. https://doi.org/10.1109/TCBB.2020.3017386 (cit. on pp. 14, 136)

Li, R., Li, L., Xu, Y., & Yang, J. (2021). Machine learning meets omics: Applications and perspectives. *Briefings in Bioinformatics*, bbab460. https://doi.org/10.1093/bib/bbab460 (cit. on p. 13)

Li, X., Zhang, Z., Liang, B., Ye, F., & Gong, W. (2021). A review: Antimicrobial resistance data mining models and prediction methods study for pathogenic bacteria. *The Journal of Antibiotics*, *74*(12), 838–849. https://doi.org/10.1038/s41429-021-00471-w (cit. on p. 13)

Li, Y., Wang, S., Bi, C., Qiu, Z., Li, M., & Gao, X. (2020). DeepSimulator1. 5: A more powerful, quicker and lighter simulator for Nanopore sequencing. *Bioinformatics*, *36*(8), 2578–2580. https://doi.org/10.1093/bioinformatics/btz963 (cit. on p. 84)

Liang, Q., Bible, P. W., Liu, Y., Zou, B., & Wei, L. (2020). DeepMicrobes: Taxonomic classification for metagenomics with deep learning. *NAR Genomics and Bioinformatics*, *2*(1), lqaa009. https://doi.org/10.1093/nargab/lqaa009 (cit. on pp. 9, 10, 15)

Lilleskov, E. A., & Parrent, J. L. (2007). Can we develop general predictive models of mycorrhizal fungal community–environment relationships? *New Phytologist*, *174*(2), 250–256. https://doi.org/10.1111/j.1469-8137.2007.02023.x (cit. on p. 136)

Linder, J., Bogard, N., Rosenberg, A. B., & Seelig, G. (2019). Deep exploration networks for rapid engineering of functional DNA sequences. *bioRxiv*, 864363. https://doi.org/10.1101/864363 (cit. on pp. 17, 71, 137)

Linder, J., Bogard, N., Rosenberg, A. B., & Seelig, G. (2020). A Generative Neural Network for Maximizing Fitness and Diversity of Synthetic DNA and Protein Sequences. *Cell Systems*, *11*(1), 49–62.e16. https://doi.org/10.1016/j.cels.2020.05.007 (cit. on p. 137)

Lindner, M. S., & Renard, B. Y. (2013). Metagenomic abundance estimation and diagnostic testing on species level. *Nucleic Acids Research*, *41*(1), e10. https://doi.org/10.1093/nar/gks803 (cit. on p. 8)

Lindner, M. S., & Renard, B. Y. (2015). Metagenomic profiling of known and unknown microbes with MicrobeGPS. *PLOS ONE*, *10*(2), e0117711. https://doi.org/10.1371/journal.pone.0117711 (cit. on pp. 8, 25)

Lindner, M. S., Strauch, B., Schulze, J. M., Tausch, S. H., Dabrowski, P. W., Nitsche, A., & Renard, B. Y. (2017). HiLive: Real-time mapping of illumina reads while sequencing. *Bioinformatics*, *33*(6), 917–319. https://doi.org/10.1093/bioinformatics/btw659 (cit. on pp. 7, 25, 77, 81, 82, 131)

Lipsitch, M., & Inglesby, T. V. (2014). Moratorium on Research Intended To Create Novel Potential Pandemic Pathogens. *mBio*, *5*(6). https://doi.org/10.1128/mBio.02366-14 (cit. on pp. 5, 46, 91)

Lipton, Z. C. (2017). The Mythos of Model Interpretability. *arXiv:1606.03490 [cs, stat]*. Retrieved February 2, 2022, from http://arxiv.org/abs/1606.03490 (cit. on p. 135)

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., & Xie, S. (2022). A ConvNet for the 2020s. *arXiv:2201.03545 [cs]*. Retrieved February 9, 2022, from http://arxiv.org/abs/2201.03545 (cit. on p. 134)

Liu-Wei, W., Kafkas, Ş., Chen, J., Dimonaco, N. J., Tegnér, J., & Hoehndorf, R. (2021). DeepViral: Prediction of novel virus–host interactions from protein sequences and infectious disease phenotypes. *Bioinformatics*, *37*(17), 2722–2729. https://doi.org/10.1093/bioinformatics/btab147 (cit. on p. 15)

Locey, K. J., & Lennon, J. T. (2016). Scaling laws predict global microbial diversity. *Proceedings of the National Academy of Sciences of the United States of America*, *113*(21), 5970–5975. https://doi.org/10.1073/pnas.1521291113 (cit. on p. 24)

Löchel, H. F., Eger, D., Sperlea, T., & Heider, D. (2020). Deep learning on chaos game representation for proteins. *Bioinformatics*, *36*(1), 272–279. https://doi.org/10.1093/bioinformatics/btz493 (cit. on pp. 10, 47)

Löchel, H. F., & Heider, D. (2021). Chaos game representation and its applications in bioinformatics. *Computational and Structural Biotechnology Journal*, *19*, 6263–6271. https://doi.org/10.1016/j.csbj.2021.11.008 (cit. on pp. 10, 134)

Lockhart, S. R., Etienne, K. A., Vallabhaneni, S., Farooqi, J., Chowdhary, A., Govender, N. P., Colombo, A. L., Calvo, B., Cuomo, C. A., Desjardins, C. A., Berkow, E. L., Castanheira, M., Magobo, R. E., Jabeen, K., Asghar, R. J., Meis, J. F., Jackson, B., Chiller, T., & Litvintseva, A. P. (2017). Simultaneous Emergence of Multidrug-Resistant Candida auris on 3 Continents Confirmed by Whole-Genome Sequencing and Epidemiological Analyses. *Clinical Infectious Diseases: An Official Publication of the Infectious Diseases Society of America*, *64*(2), 134–140. https://doi.org/10.1093/cid/ciw691 (cit. on p. 98)

Loka, T. P., Tausch, S. H., Dabrowski, P. W., Radonic, A., Nitsche, A., & Renard, B. Y. (2018). PriLive: Privacy-preserving real-time filtering for next-generation sequencing. *Bioinformatics*, *34*(14), 2376–2383. https://doi.org/10.1093/bioinformatics/bty128 (cit. on pp. 111, 131)

Loka, T. P., Tausch, S. H., & Renard, B. Y. (2019). Reliable variant calling during runtime of Illumina sequencing. *Scientific Reports*, *9*(1), 1–8. https://doi.org/10.1038/s41598-019-52991-z (cit. on pp. 7, 20, 25, 76, 77, 81, 82, 85, 87, 131)

Loose, M., Malla, S., & Stout, M. (2016). Real-time selective sequencing using nanopore technology. *Nature Methods*, *13*(9), 751–754. https://doi.org/10.1038/nmeth.3930 (cit. on pp. 6, 84, 95, 132, 135)

Lovett, B., & St. Leger, R. J. (2017). The Insect Pathogens. *Microbiology Spectrum*, *5*(2), 5.2.01. https://doi.org/10.1128/microbiolspec.FUNK-0001-2016 (cit. on p. 102)

Lu, J., Breitwieser, F. P., Thielen, P., & Salzberg, S. L. (2017). Bracken: Estimating species abundance in metagenomics data. *PeerJ Computer Science*, *3*, e104. https://doi.org/10.7717/peerj-cs.104 (cit. on p. 9)

Lu, T., Yao, B., & Zhang, C. (2012). DFVF: Database of fungal virulence factors. *Database*, *2012*(bas032). https://doi.org/10.1093/database/bas032 (cit. on pp. 98, 101)

Lundberg, S. M., & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 30* (pp. 4765–4774). Curran Associates, Inc. Retrieved November 4, 2019, from http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf. (Cit. on pp. 11, 13, 48, 56, 57, 72)

Luo, Z., Wang, L., Wang, Y., Zhang, W., Guo, Y., Shen, Y., Jiang, L., Wu, Q., Zhang, C., Cai, Y., & Dai, J. (2018). Identifying and characterizing SCRaMbLEd synthetic yeast using ReSCuES. *Nature Communications*, *9*(1), 1930. https://doi.org/10.1038/s41467-017-00806-y (cit. on p. 100)

Lytras, S., Xia, W., Hughes, J., Jiang, X., & Robertson, D. L. (2021). The animal origin of SARS-CoV-2. *Science*, *373*(6558), 968–970. https://doi.org/10.1126/science.abh0117 (cit. on p. 5)

MacDonald, N. J., & Beiko, R. G. (2010). Efficient learning of microbial genotype–phenotype association rules. *Bioinformatics*, *26*(15), 1834–1840. https://doi.org/10.1093/bioinformatics/btq305 (cit. on p. 13)

Manara, S., Pasolli, E., Dolce, D., Ravenni, N., Campana, S., Armanini, F., Asnicar, F., Mengoni, A., Galli, L., Montagnani, C., Venturini, E., Rota-Stabelli, O., Grandi, G., Taccetti, G., & Segata, N. (2018). Whole-genome epidemiology, characterisation, and phylogenetic reconstruction of staphylococcus aureus strains in a paediatric hospital. *Genome Medicine*, *10*(1), 82. https://doi.org/10.1186/s13073-018-0593-7 (cit. on pp. 29, 79, 84)

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., Gwadz, M., Hurwitz, D. I., Lu, F., Marchler, G. H., Song, J. S., Thanki, N., Wang, Z., Yamashita, R. A., Zhang, D., ... Bryant, S. H. (2017). CDD/SPARCLE: Functional classification of proteins

via subfamily domain architectures. *Nucleic Acids Research*, *45*(D1), D200–D203. https://doi.org/10.1093/nar/gkw1129 (cit. on p. 68)

Martins-Santana, L., Nora, L. C., Sanches-Medeiros, A., Lovate, G. L., Cassiano, M. H. A., & Silva-Rocha, R. (2018). Systems and Synthetic Biology Approaches to Engineer Fungi for Fine Chemical Production. *Frontiers in Bioengineering and Biotechnology*, *6*. https://doi.org/10.3389/fbioe.2018.00117 (cit. on p. 100)

Matejczyk, S., & Michalak, T. (2015). *Solving Influence Maximization Problem Using Methods from Cooperative Game Theory*. Instytut Podstaw Informatyki PAN. (Cit. on p. 57).

McInnes, L., Healy, J., & Melville, J. (2020). UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv:1802.03426 [cs, stat]*. Retrieved June 28, 2021, from http://arxiv.org/abs/1802.03426 (cit. on p. 110)

McNair, K., Bailey, B. A., & Edwards, R. A. (2012). PHACTS, a computational approach to classifying the lifestyle of phages. *Bioinformatics*, *28*(5), 614–618. https://doi.org/10.1093/bioinformatics/bts014 (cit. on p. 14)

Menzel, P., Ng, K. L., & Krogh, A. (2016). Fast and sensitive taxonomic classification for metagenomics with Kaiju. *Nature Communications*, *7*(1), 11257. https://doi.org/10.1038/ncomms11257 (cit. on p. 8)

Meyer, F., Fritz, A., Deng, Z.-L., Koslicki, D., Gurevich, A., Robertson, G., Alser, M., Antipov, D., Beghini, F., Bertrand, D., Brito, J. J., Brown, C. T., Buchmann, J., Buluç, A., Chen, B., Chikhi, R., Clausen, P. T., Cristian, A., Dabrowski, P. W., … McHardy, A. C. (2021). Critical Assessment of Metagenome Interpretation - the second round of challenges. *bioRxiv*, 2021.07.12.451567. https://doi.org/10.1101/2021.07.12.451567 (cit. on p. 9)

Miao, Y., Liu, F., Hou, T., & Liu, Y. (2021). Virtifier: A deep learning-based identifier for viral sequences from metagenomes. *Bioinformatics*, btab845. https://doi.org/10.1093/bioinformatics/btab845 (cit. on pp. 15, 132)

Mihara, T., Nishimura, Y., Shimizu, Y., Nishiyama, H., Yoshikawa, G., Uehara, H., Hingamp, P., Goto, S., & Ogata, H. (2016). Linking virus genomes with host taxonomy. *Viruses*, *8*(3), 66. https://doi.org/10.3390/v8030066 (cit. on pp. 49, 78)

Milanese, A., Mende, D. R., Paoli, L., Salazar, G., Ruscheweyh, H.-J., Cuenca, M., Hingamp, P., Alves, R., Costea, P. I., Coelho, L. P., Schmidt, T. S. B., Almeida, A., Mitchell, A. L., Finn, R. D., Huerta-Cepas, J., Bork, P., Zeller, G., & Sunagawa, S. (2019). Microbial abundance, activity and population genomic profiling with mOTUs2. *Nature Communications*, *10*(1), 1014. https://doi.org/10.1038/s41467-019-08844-4 (cit. on p. 9)

Miller, N. A., Farrow, E. G., Gibson, M., Willig, L. K., Twist, G., Yoo, B., Marrs, T., Corder, S., Krivohlavek, L., Walter, A., Petrikin, J. E., Saunders, C. J., Thiffault, I., Soden, S. E., Smith, L. D., Dinwiddie, D. L., Herd, S., Cakici, J. A., Catreux, S., … Kingsmore, S. F. (2015). A 26-hour system of highly sensitive whole genome sequencing for emergency management of genetic diseases. *Genome Medicine*, *7*(1), 100. https://doi.org/10.1186/s13073-015-0221-8 (cit. on p. 76)

Mirdita, M., Steinegger, M., Breitwieser, F., Söding, J., & Levy Karin, E. (2021). Fast and sensitive taxonomic assignment to metagenomic contigs. *Bioinformatics*, *37*(18), 3029–3031. https://doi.org/10.1093/bioinformatics/btab184 (cit. on pp. 8, 15)

Mock, F., Kretschmer, F., Kriese, A., Böcker, S., & Marz, M. (2021). BERTax: Taxonomic classification of DNA sequences with Deep Neural Networks. *biorXiv*, 2021.07.09.451778. Retrieved December 10, 2021, from https://www.biorxiv.org/content/10.1101/2021.07.09.451778v1 (cit. on pp. 15, 134)

Mock, F., Viehweger, A., Barth, E., & Marz, M. (2020). VIDHOP, viral host prediction with Deep Learning. *Bioinformatics*. https://doi.org/10.1093/bioinformatics/btaa705 (cit. on pp. 14, 17, 47, 48, 78, 100, 136)

Mollentze, N., Babayan, S. A., & Streicker, D. G. (2021). Identifying and prioritizing potential human-infecting viruses from their genome sequences. *PLOS Biology*, *19*(9), e3001390. https://doi.org/10.1371/journal.pbio.3001390 (cit. on p. 14)

Montavon, G., Binder, A., Lapuschkin, S., Samek, W., & Müller, K.-R. (2019). Layer-Wise Relevance Propagation: An Overview. In W. Samek, G. Montavon, A. Vedaldi, L. K. Hansen, & K.-R. Müller (Eds.), *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning* (pp. 193–209). Springer International Publishing. https://doi.org/10.1007/978-3-030-28954-6_10. (Cit. on p. 12)

Morens, D. M., & Fauci, A. S. (2020). Emerging Pandemic Diseases: How We Got to COVID-19. *Cell*, *182*(5), 1077–1092. https://doi.org/10.1016/j.cell.2020.08.021 (cit. on p. 3)

Moustafa, A., Xie, C., Kirkness, E., Biggs, W., Wong, E., Turpaz, Y., Bloom, K., Delwart, E., Nelson, K. E., Venter, J. C., & Telenti, A. (2017). The blood DNA virome in 8,000 humans. *PLOS Pathogens*, *13*(3), e1006292. https://doi.org/10.1371/journal.ppat.1006292 (cit. on pp. 51, 54, 70)

Naccache, S. N., Federman, S., Veeraraghavan, N., Zaharia, M., Lee, D., Samayoa, E., Bouquet, J., Greninger, A. L., Luk, K.-C., Enge, B., Wadford, D. A., Messenger, S. L., Genrich, G. L., Pellegrino, K., Grard, G., Leroy, E., Schneider, B. S., Fair, J. N., Martínez, M. A., … Chiu, C. Y. (2014). A cloud-compatible bioinformatics pipeline for ultrarapid pathogen identification from next-generation sequencing

of clinical samples. *Genome Research*, *24*(7), 1180–1192. https://doi.org/10.1101/gr.171934.113 (cit. on pp. 8, 98)

Nair, S., Kim, D. S., Perricone, J., & Kundaje, A. (2019). Integrating regulatory DNA sequence and gene expression to predict genome-wide chromatin accessibility across cellular contexts. *Bioinformatics*, *35*(14), i108–i116. https://doi.org/10.1093/bioinformatics/btz352 (cit. on pp. 10, 48)

Nasri, F. (2020). *Predicting pathogenic potential of novel fungal DNA using deep neural networks* (Master's thesis). Free University of Berlin. (Cit. on p. 21).

National Academies of Sciences, Engineering, and Medicine. (2018). *Biodefense in the age of synthetic biology*. The National Academies Press, Washington, DC. https://doi.org/10.17226/24890. (Cit. on pp. 4, 5, 17, 24, 71, 91, 137)

National Research Council. (2010). *Sequence-based classification of select agents: A brighter line*. The National Academies Press, Washington, DC. https://doi.org/10.17226/12970. (Cit. on pp. 5, 24, 71, 77)

Ndow, G., Ambe, J. R., & Tomori, O. (2019). Emerging Infectious Diseases: A Historical and Scientific Review. *Socio-cultural Dimensions of Emerging Infectious Diseases in Africa*, 31–40. https://doi.org/10.1007/978-3-030-17474-3_3 (cit. on p. 3)

Neu, H. C. (1992). The Crisis in Antibiotic Resistance. *Science*, *257*(5073), 1064–1073. https://doi.org/10.1126/science.257.5073.1064 (cit. on pp. 13, 129)

Nielsen, A. A. K., & Voigt, C. A. (2018). Deep learning to predict the lab-of-origin of engineered DNA. *Nature Communications*, *9*(1), 3135. https://doi.org/10.1038/s41467-018-05378-z (cit. on pp. 24, 78)

Nix, R., & Kantarciouglu, M. (2012). Incentive Compatible Privacy-Preserving Distributed Classification. *IEEE Transactions on Dependable and Secure Computing*, *9*(4), 451–462. https://doi.org/10.1109/TDSC.2011.52 (cit. on p. 57)

Noyce, R. S., Lederman, S., & Evans, D. H. (2018). Construction of an infectious horsepox virus vaccine from chemically synthesized DNA fragments. *PLOS ONE*, *13*(1), e0188453. https://doi.org/10.1371/journal.pone.0188453 (cit. on pp. 5, 46, 91)

Nuzzo, J. B., Mullen, L., Snyder, M., Cicero, A., & Inglesby, T. V. (2019). *Preparedness for a High-Impact Respiratory Pathogen Pandemic*. The Johns Hopkins Center for Health Security. https://www.centerforhealthsecurity.org/our-work/pubs_archive/pubs-pdfs/2019/190918-GMPBreport-respiratorypathogen.pdf. (Cit. on pp. 3, 4)

O'Brien, J. T., & Nelson, C. (2020). Assessing the Risks Posed by the Convergence of Artificial Intelligence and Biotechnology. *Health Security*, *18*(3), 219–227. https://doi.org/10.1089/hs.2019.0122 (cit. on pp. 71, 137, 138)

Onimaru, K., Nishimura, O., & Kuraku, S. (2020). Predicting gene regulatory regions with a convolutional neural network for processing double-strand genome sequence information. *PLOS ONE*, *15*(7), e0235748. https://doi.org/10.1371/journal.pone.0235748 (cit. on pp. 10, 26)

Ounit, R., Wanamaker, S., Close, T. J., & Lonardi, S. (2015). CLARK: Fast and accurate classification of metagenomic and genomic sequences using discriminative k-mers. *BMC Genomics*, *16*(1), 236. https://doi.org/10.1186/s12864-015-1419-2 (cit. on p. 9)

Ovchinnikov, S., & Huang, P.-S. (2021). Structure-based protein design with deep learning. *Current Opinion in Chemical Biology*, *65*, 136–144. https://doi.org/10.1016/j.cbpa.2021.08.004 (cit. on p. 17)

Papkou, A., Guzella, T., Yang, W., Koepper, S., Pees, B., Schalkowski, R., Barg, M.-C., Rosenstiel, P. C., Teotónio, H., & Schulenburg, H. (2019). The genomic basis of Red Queen dynamics during rapid reciprocal host–pathogen coevolution. *Proceedings of the National Academy of Sciences*, *116*(3), 923–928. https://doi.org/10.1073/pnas.1810402116 (cit. on p. 129)

Patz, J. A., Campbell-Lendrum, D., Holloway, T., & Foley, J. A. (2005). Impact of regional climate change on human health. *Nature*, *438*(7066), 310–317. https://doi.org/10.1038/nature04188 (cit. on pp. 4, 129)

Patz, S., Gautam, A., Becker, M., Ruppel, S., Rodríguez-Palenzuela, P., & Huson, D. (2021). PLaBAse: A comprehensive web resource for analyzing the plant growth-promoting potential of plant-associated bacteria. *bioRxiv*, 2021.12.13.472471. Retrieved January 29, 2022, from https://www.biorxiv.org/content/10.1101/2021.12.13.472471v1 (cit. on p. 137)

Paulussen, C., Hallsworth, J. E., Álvarez-Pérez, S., Nierman, W. C., Hamill, P. G., Blain, D., Rediers, H., & Lievens, B. (2017). Ecology of aspergillosis: Insights into the pathogenic potency of Aspergillus fumigatus and some other Aspergillus species. *Microbial Biotechnology*, *10*(2), 296–322. https://doi.org/10.1111/1751-7915.12367 (cit. on p. 102)

Payne, A., Holmes, N., Clarke, T., Munro, R., Debebe, B. J., & Loose, M. (2021). Readfish enables targeted nanopore sequencing of gigabase-sized genomes. *Nature Biotechnology*, *39*(4), 442–450. https://doi.org/10.1038/s41587-020-00746-x (cit. on p. 135)

Piddock, L. J. V. (2012). The crisis of no new antibiotics–what is the way forward? *The Lancet. Infectious Diseases*, *12*(3), 249–253. https://doi.org/10.1016/S1473-3099(11)70316-4 (cit. on pp. 13, 129)

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., Guarino, B., Spreafico, R., Cameroni, E., Case, J. B., Chen, R. E., Havenar-Daughton, C., Snell, G., Telenti, A., … Corti, D. (2020). Cross-neutralization of SARS-CoV-2 by a human monoclonal SARS-CoV antibody. *Nature*, 1–10. https://doi.org/10.1038/s41586-020-2349-y (cit. on p. 69)

Piro, V. C., Dadi, T. H., Seiler, E., Reinert, K., & Renard, B. Y. (2020). Ganon: Precise metagenomics classification against large and up-to-date sets of reference sequences. *Bioinformatics*, *36*(Supplement_1), i12–i20. https://doi.org/10.1093/bioinformatics/btaa458 (cit. on pp. 5, 9, 24, 98, 124)

Piro, V. C., Lindner, M. S., & Renard, B. Y. (2016). DUDes: A top-down taxonomic profiler for metagenomics. *Bioinformatics*, *32*(15), 2272–2280. https://doi.org/10.1093/bioinformatics/btw150 (cit. on p. 8)

Piro, V. C., Matschkowski, M., & Renard, B. Y. (2017). Metameta: Integrating metagenome analysis tools to improve taxonomic profiling. *Microbiome*, *5*(1), 1–11 (cit. on p. 98).

Poplin, R., Chang, P.-C., Alexander, D., Schwartz, S., Colthurst, T., Ku, A., Newburger, D., Dijamco, J., Nguyen, N., Afshar, P. T., Gross, S. S., Dorfman, L., McLean, C. Y., & DePristo, M. A. (2018). A universal SNP and small-indel variant caller using deep neural networks. *Nature Biotechnology*, *36*(10), 983–987. https://doi.org/10.1038/nbt.4235 (cit. on pp. 10, 47)

Prakash, P. Y., Irinyi, L., Halliday, C., Chen, S., Robert, V., & Meyer, W. (2017). Online Databases for Taxonomy and Identification of Pathogenic Fungi and Proposal for a Cloud-Based Dynamic Data Network Platform. *Journal of Clinical Microbiology*, *55*(4), 1011–1024. https://doi.org/10.1128/JCM.02084-16 (cit. on p. 102)

Qiang, X.-L., Xu, P., Fang, G., Liu, W.-B., & Kou, Z. (2020). Using the spike protein feature to predict infection risk and monitor the evolutionary dynamic of coronavirus. *Infectious Diseases of Poverty*, *9*(1), 33. https://doi.org/10.1186/s40249-020-00649-8 (cit. on p. 14)

Qin, Q., & Feng, J. (2017). Imputation for transcription factor binding predictions based on deep learning. *PLOS Computational Biology*, *13*(2), e1005403. https://doi.org/10.1371/journal.pcbi.1005403 (cit. on p. 26)

Quang, D., & Xie, X. (2016). DanQ: A hybrid convolutional and recurrent deep neural network for quantifying the function of DNA sequences. *Nucleic Acids Research*, *44*(11), e107–e107. https://doi.org/10.1093/nar/gkw226 (cit. on pp. 10, 26, 47)

Quang, D., & Xie, X. (2019). FactorNet: A deep learning framework for predicting cell type specific transcription factor binding from nucleotide-resolution sequential data. *Methods*, *166*, 40–47. https://doi.org/10.1016/j.ymeth.2019.03.020 (cit. on pp. 10, 26)

Quick, J., Loman, N. J., Duraffour, S., Simpson, J. T., Severi, E., Cowley, L., Bore, J. A., Koundouno, R., Dudas, G., Mikhail, A., Ouédraogo, N., Afrough, B., Bah, A., Baum, J. H., Becker-Ziaja, B., Boettcher, J.-P., Cabeza-Cabrerizo, M., Camino-Sanchez, A., Carter, L. L., … Carroll, M. W. (2016). Real-time, portable genome sequencing for Ebola surveillance. *Nature*, *530*(7589), 228–232. https://doi.org/10.1038/nature16996 (cit. on p. 6)

Raimondi, S., Amaretti, A., Gozzoli, C., Simone, M., Righini, L., Candeliere, F., Brun, P., Ardizzoni, A., Colombari, B., Paulone, S., & et al. (2019). Longitudinal survey of fungi in the human gut: Its profiling, phenotyping, and colonization. *Frontiers in Microbiology*, *10*. https://doi.org/10.3389/fmicb.2019.01575 (cit. on p. 98)

Rang, F. J., Kloosterman, W. P., & de Ridder, J. (2018). From squiggle to basepair: Computational approaches for improving nanopore sequencing read accuracy. *Genome Biology*, *19*(1), 90. https://doi.org/10.1186/s13059-018-1462-9 (cit. on p. 84)

Rehner, S. A., & Buckley, E. (2005). A Beauveria phylogeny inferred from nuclear ITS and EF1-alpha sequences: Evidence for cryptic diversification and links to Cordyceps teleomorphs. *Mycologia*, *97*(1), 84–98. https://doi.org/10.3852/mycologia.97.1.84 (cit. on p. 102)

Relman, D. A. (2020). Opinion: To stop the next pandemic, we need to unravel the origins of COVID-19. *Proceedings of the National Academy of Sciences*, *117*(47), 29246–29248. https://doi.org/10.1073/pnas.2021133117 (cit. on p. 5)

Remita, M. A., Halioui, A., Malick Diouara, A. A., Daigle, B., Kiani, G., & Diallo, A. B. (2017). A machine learning approach for viral genome classification. *BMC bioinformatics*, *18*(1), 208. https://doi.org/10.1186/s12859-017-1602-3 (cit. on p. 15)

Ren, J., Song, K., Deng, C., Ahlgren, N. A., Fuhrman, J. A., Li, Y., Xie, X., Poplin, R., & Sun, F. (2020). Identifying viruses from metagenomic data using deep learning. *Quantitative Biology*, *8*(1), 64–77. https://doi.org/10.1007/s40484-019-0187-4 (cit. on pp. 15, 48, 78, 132)

Renard, B. Y., Xu, B., Kirchner, M., Zickmann, F., Winter, D., Korten, S., Brattig, N. W., Tzur, A., Hamprecht, F. A., & Steen, H. (2012). Overcoming species boundaries in peptide identification with bayesian information criterion-driven error-tolerant peptide search (biceps). *Molecular & cellular proteomics*, *11*(7), M111–014167 (cit. on p. 127).

Rentzsch, R., Deneke, C., Nitsche, A., & Renard, B. Y. (2020). Predicting bacterial virulence factors - evaluation of machine learning and negative data strategies. *Briefings in Bioinformatics*, *21*(5), 1596–1608. https://doi.org/10.1093/bib/bbz076 (cit. on p. 13)

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why Should I Trust You?": Explaining the Predictions of Any Classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144. https://doi.org/10.1145/2939672.2939778 (cit. on p. 12)

Richardson, S. M., Mitchell, L. A., Stracquadanio, G., Yang, K., Dymond, J. S., DiCarlo, J. E., Lee, D., Huang, C. L. V., Chandrasegaran, S., Cai, Y., Boeke, J. D., & Bader, J. S. (2017). Design of a synthetic yeast genome. *Science*, *355*(6329), 1040–1044. https://doi.org/10.1126/science.aaf4557 (cit. on p. 100)

Riedel, S. (2004). Biological warfare and bioterrorism: A historical review. *Proceedings (Baylor University. Medical Center)*, *17*(4), 400–406. Retrieved December 5, 2021, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1200679/ (cit. on p. 4)

Riesselman, A. J., Ingraham, J. B., & Marks, D. S. (2018). Deep generative models of genetic variation capture the effects of mutations. *Nature Methods*, *15*(10), 816–822. https://doi.org/10.1038/s41592-018-0138-4 (cit. on p. 137)

Rizzo, R., Fiannaca, A., La Rosa, M., & Urso, A. (2016). Classification Experiments of DNA Sequences by Using a Deep Neural Network and Chaos Game Representation. *Proceedings of the 17th International Conference on Computer Systems and Technologies 2016*, 222–228. https://doi.org/10.1145/2983468.2983489 (cit. on pp. 10, 47)

Rojas-Carulla, M., Tolstikhin, I., Luque, G., Youngblut, N., Ley, R., & Schölkopf, B. (2019). GeNet: Deep Representations for Metagenomics. *arXiv:1901.11015 [cs, q-bio, stat]*. Retrieved December 10, 2021, from http://arxiv.org/abs/1901.11015 (cit. on pp. 9, 15)

Rosen, G., Garbarine, E., Caseiro, D., Polikar, R., & Sokhansanj, B. (2008). Metagenome fragment classification using n-mer frequency profiles. *Advances in Bioinformatics*. https://doi.org/10.1155/2008/205969 (cit. on pp. 9, 25)

Rosen, G. L., & Lim, T. Y. (2012). NBC update: The addition of viral and fungal databases to the Naïve Bayes classification tool. *BMC Research Notes*, *5*(1), 81. https://doi.org/10.1186/1756-0500-5-81 (cit. on p. 9)

Rosen, G. L., Reichenberger, E. R., & Rosenfeld, A. M. (2011). NBC: The naïve bayes classification tool webserver for taxonomic classification of metagenomic reads. *Bioinformatics*, *27*(1), 127–129. https://doi.org/10.1093/bioinformatics/btq619 (cit. on pp. 9, 25, 32, 77)

Roux, S., Enault, F., Hurwitz, B. L., & Sullivan, M. B. (2015). VirSorter: Mining viral signal from microbial genomic data. *PeerJ*, *3*, e985. https://doi.org/10.7717/peerj.985 (cit. on p. 15)

Roux, S., Hallam, S. J., Woyke, T., & Sullivan, M. B. (2015). Viral dark matter and virus-host interactions resolved from publicly available microbial genomes. *eLife*, *4*. https://doi.org/10.7554/eLife.08490 (cit. on p. 15)

Rozo, M., & Gronvall, G. K. (2015). The Reemergent 1977 H1N1 Strain and the Gain-of-Function Debate. *mBio*, *6*(4), e01013–15. https://doi.org/10.1128/mBio.01013-15 (cit. on p. 4)

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, *1*(5), 206–215. https://doi.org/10.1038/s42256-019-0048-x (cit. on pp. 11, 135)

Sandberg, A., & Nelson, C. (2020). Who Should We Fear More: Biohackers, Disgruntled Postdocs, or Bad Governments? A Simple Risk Chain Model of Biorisk. *Health Security*, *18*(3), 155–163. https://doi.org/10.1089/hs.2019.0115 (cit. on p. 138)

Satoh, K., Makimura, K., Hasumi, Y., Nishiyama, Y., Uchida, K., & Yamaguchi, H. (2009). Candida auris sp. nov., a novel ascomycetous yeast isolated from the external ear canal of an inpatient in a Japanese hospital. *Microbiology and Immunology*, *53*(1), 41–44. https://doi.org/10.1111/j.1348-0421.2008.00083.x (cit. on pp. 98, 105)

Sayers, E. W., Beck, J., Bolton, E. E., Bourexis, D., Brister, J. R., Canese, K., Comeau, D. C., Funk, K., Kim, S., Klimke, W., Marchler-Bauer, A., Landrum, M., Lathrop, S., Lu, Z., Madden, T. L., O'Leary, N., Phan, L., Rangwala, S. H., Schneider, V. A., … Sherry, S. T. (2021). Database resources of the National Center for Biotechnology Information. *Nucleic Acids Research*, *49*(D1), D10–D17. https://doi.org/10.1093/nar/gkaa892 (cit. on pp. 98, 99)

Sayers, E. W., Cavanaugh, M., Clark, K., Pruitt, K. D., Schoch, C. L., Sherry, S. T., & Karsch-Mizrachi, I. (2021). GenBank. *Nucleic Acids Research*, *49*(D1), D92–D96. https://doi.org/10.1093/nar/gkaa1023 (cit. on p. 99)

Schardl, C. L., Young, C. A., Hesse, U., Amyotte, S. G., Andreeva, K., Calie, P. J., Fleetwood, D. J., Haws, D. C., Moore, N., Oeser, B., Panaccione, D. G., Schweri, K. K., Voisey, C. R., Farman, M. L., Jaromczyk, J. W., Roe, B. A., O'Sullivan, D. M., Scott, B., Tudzynski, P., … Zeng, Z. (2013). Plant-Symbiotic Fungi as Chemical Engineers: Multi-Genome Analysis of the Clavicipitaceae Reveals Dynamics of Alkaloid Loci. *PLOS Genetics*, *9*(2), e1003323. https://doi.org/10.1371/journal.pgen.1003323 (cit. on p. 103)

Scheele, B. C., Pasmans, F., Skerratt, L. F., Berger, L., Martel, A., Beukema, W., Acevedo, A. A., Burrowes, P. A., Carvalho, T., Catenazzi, A., De la Riva, I., Fisher, M. C., Flechas, S. V., Foster, C. N., Frías-Álvarez, P., Garner, T. W. J., Gratwicke, B., Guayasamin, J. M., Hirschfeld, M., … Canessa, S. (2019). Amphibian fungal panzootic causes catastrophic and ongoing loss of biodiversity. *Science*, *363*(6434), 1459–1463. https://doi.org/10.1126/science.aav0379 (cit. on p. 105)

Schiebenhoefer, H., Bossche, T. V. D., Fuchs, S., Renard, B. Y., Muth, T., & Martens, L. (2019). Challenges and promise at the interface of metaproteomics and genomics: An overview of recent progress in metaproteogenomic data analysis. *Expert Review of Proteomics*, *16*(5), 375–390. https://doi.org/10.1080/14789450.2019.1609944 (cit. on p. 127)

Schiebenhoefer, H., Schallert, K., Renard, B. Y., Trappe, K., Schmid, E., Benndorf, D., Riedel, K., Muth, T., & Fuchs, S. (2020). A complete and flexible workflow for metaproteomics data analysis based on MetaProteomeAnalyzer and Prophane. *Nature Protocols*, *15*(10), 3212–3239. https://doi.org/10.1038/s41596-020-0368-7 (cit. on p. 127)

Schneider, T. D., & Stephens, R. M. (1990). Sequence logos: A new way to display consensus sequences. *Nucleic Acids Research*, *18*(20), 6097–6100. https://doi.org/10.1093/nar/18.20.6097 (cit. on p. 48)

Schoch, C. L., Ciufo, S., Domrachev, M., Hotton, C. L., Kannan, S., Khovanskaya, R., Leipe, D., Mcveigh, R., O'Neill, K., Robbertse, B., et al. (2020). Ncbi taxonomy: A comprehensive update on curation, resources and tools. *Database*, *2020* (cit. on pp. 99, 102).

Schreiber, J., Lu, Y. Y., & Noble, W. S. (2020). Ledidi: Designing genomic edits that induce functional activity. *bioRxiv*, 2020.05.21.109686. https://doi.org/10.1101/2020.05.21.109686 (cit. on pp. 17, 71, 137)

Schulenburg, H., & Félix, M.-A. (2017). The Natural Biotic Environment of Caenorhabditis elegans. *Genetics*, *206*(1), 55–86. https://doi.org/10.1534/genetics.116.195511 (cit. on p. 102)

Sczyrba, A., Hofmann, P., Belmann, P., Koslicki, D., Janssen, S., Dröge, J., Gregor, I., Majda, S., Fiedler, J., Dahms, E., Bremges, A., Fritz, A., Garrido-Oter, R., Jørgensen, T. S., Shapiro, N., Blood, P. D., Gurevich, A., Bai, Y., Turaev, D., … McHardy, A. C. (2017). Critical Assessment of Metagenome Interpretation—a benchmark of metagenomics software. *Nature Methods*, *14*(11), 1063–1071. https://doi.org/10.1038/nmeth.4458 (cit. on p. 9)

Segata, N., Waldron, L., Ballarini, A., Narasimhan, V., Jousson, O., & Huttenhower, C. (2012). Metagenomic microbial community profiling using unique clade-specific marker genes. *Nature Methods*, *9*(8), 811–814. https://doi.org/10.1038/nmeth.2066 (cit. on p. 9)

Seidel, A. (2018). *Evaluation and interpretation of reverse-complement convolutional neural networks for bacterial pathogenicity prediction* (Master's thesis). Free University of Berlin. (Cit. on pp. 18, 20).

Seyedmousavi, S., Bosco, S. d. M. G., de Hoog, S., Ebel, F., Elad, D., Gomes, R. R., Jacobsen, I. D., Jensen, H. E., Martel, A., Mignon, B., Pasmans, F., Piecková, E., Rodrigues, A. M., Singh, K., Vicente, V. A., Wibbelt, G., Wiederhold, N. P., & Guillot, J. (2018). Fungal infections in animals: A patchwork of different situations. *Medical Mycology*, *56*(suppl_1), 165–187. https://doi.org/10.1093/mmy/myx104 (cit. on p. 102)

Seyedmousavi, S., Guillot, J., Arné, P., de Hoog, G. S., Mouton, J. W., Melchers, W. J. G., & Verweij, P. E. (2015). Aspergillus and aspergilloses in wild and domestic animals: A global health concern with parallels to human disease. *Medical Mycology*, *53*(8), 765–797. https://doi.org/10.1093/mmy/myv067 (cit. on p. 102)

Shang, J., Jiang, J., & Sun, Y. (2021). Bacteriophage classification for assembled contigs using graph convolutional network. *Bioinformatics*, *37*(Supplement_1), i25–i33. https://doi.org/10.1093/bioinformatics/btab293 (cit. on p. 15)

Shang, J., & Sun, Y. (2020). CHEER: hierarCHical taxonomic classification for viral mEtagEnomic data via deep leaRning. *bioRxiv*, 2020.03.26.009001. https://doi.org/10.1101/2020.03.26.009001 (cit. on p. 15)

Shang, J., Tang, X., Guo, R., & Sun, Y. (2022). Accurate identification of bacteriophages from metagenomic data using Transformer. *arXiv:2201.04778 [q-bio]*. Retrieved January 26, 2022, from http://arxiv.org/abs/2201.04778 (cit. on p. 15)

Shang, Y., Feng, P., & Wang, C. (2015). Fungi That Infect Insects: Altering Host Behavior and Beyond. *PLOS Pathogens*, *11*(8), e1005037. https://doi.org/10.1371/journal.ppat.1005037 (cit. on p. 102)

Shin, J.-E., Riesselman, A. J., Kollasch, A. W., McMahon, C., Simon, E., Sander, C., Manglik, A., Kruse, A. C., & Marks, D. S. (2021). Protein design and variant prediction using autoregressive generative models. *Nature Communications*, *12*(1), 2403. https://doi.org/10.1038/s41467-021-22732-w (cit. on p. 137)

Shrikumar, A., Greenside, P., & Kundaje, A. (2017a). Learning Important Features Through Propagating Activation Differences. In D. Precup & Y. W. Teh (Eds.), *Proceedings of the 34th International Conference on Machine Learning* (pp. 3145–3153). PMLR. http://proceedings.mlr.press/v70/shrikumar17a.html. (Cit. on pp. 11, 12, 48, 56–58, 72)

Shrikumar, A., Greenside, P., & Kundaje, A. (2017b). Reverse-complement parameter sharing improves deep learning models for genomics. *bioRxiv*, *103663*. https://doi.org/10.1101/103663 (cit. on pp. 10, 18, 20, 26, 30)

Shrikumar, A., Tian, K., Shcherbina, A., Avsec, Ž., Banerjee, A., Sharmin, M., Nair, S., & Kundaje, A. (2019). TF-MoDISco v0.4.2.2-alpha: Technical Note. *arXiv:1811.00416 [cs, q-bio, stat]*. Retrieved November 4, 2019, from http://arxiv.org/abs/1811.00416 (cit. on p. 48)

Simmonds, P., & Aiewsakun, P. (2018). Virus classification – where do you draw the line? *Archives of Virology*, *163*(8), 2037–2046. https://doi.org/10.1007/s00705-018-3938-z (cit. on p. 52)

Simonyan, K., Vedaldi, A., & Zisserman, A. (2014). Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Workshop Track Proceedings*. http://arxiv.org/abs/1312.6034 (cit. on p. 12)

Sinai, S., Kelsic, E., Church, G. M., & Nowak, M. A. (2018). Variational auto-encoding of protein sequences. *arXiv:1712.03346 [cs, q-bio]*. Retrieved May 26, 2021, from http://arxiv.org/abs/1712.03346 (cit. on p. 137)

Sinai, S., & Kelsic, E. D. (2020). A primer on model-guided exploration of fitness landscapes for biological sequence design. *arXiv:2010.10614 [cs, q-bio]*. Retrieved November 4, 2020, from http://arxiv.org/abs/2010.10614 (cit. on p. 137)

Sinai, S., Wang, R., Whatley, A., Slocum, S., Locane, E., & Kelsic, E. D. (2020). AdaLead: A simple and robust adaptive greedy search algorithm for sequence design. *arXiv:2010.02141 [cs, math, q-bio]*. Retrieved December 15, 2021, from http://arxiv.org/abs/2010.02141 (cit. on pp. 17, 137)

Sironi, M., Cagliani, R., Forni, D., & Clerici, M. (2015). Evolutionary insights into host–pathogen interactions from mammalian sequence data. *Nature Reviews Genetics*, *16*(4), 224–236. https://doi.org/10.1038/nrg3905 (cit. on p. 129)

Skamnioti, P., & Gurr, S. J. (2009). Against the grain: Safeguarding rice from rice blast disease. *Trends in Biotechnology*, *27*(3), 141–150. https://doi.org/10.1016/j.tibtech.2008.12.002 (cit. on p. 105)

Smith, J. M. (2006). Fungal Pathogens of Nonhuman Animals. *eLS*. American Cancer Society. https://doi.org/10.1038/npg.els.0004235. (Cit. on p. 102)

Sobel, J. D. (2007). Vulvovaginal candidosis. *The Lancet*, *369*(9577), 1961–1971. https://doi.org/10.1016/S0140-6736(07)60917-9 (cit. on p. 98)

Sohn, M. B., An, L., Pookhao, N., & Li, Q. (2014). Accurate genome relative abundance estimation for closely related species in a metagenomic sample. *BMC Bioinformatics*, *15*(1), 242. https://doi.org/10.1186/1471-2105-15-242 (cit. on p. 8)

Solis-Reyes, S., Avino, M., Poon, A., & Kari, L. (2018). An open-source k-mer based machine learning tool for fast and accurate subtyping of HIV-1 genomes. *PLOS ONE*, *13*(11), e0206409. https://doi.org/10.1371/journal.pone.0206409 (cit. on p. 15)

Spivak, E. S., & Hanson, K. E. (2018). Candida auris: An emerging fungal pathogen. *Journal of clinical microbiology*, *56*(2) (cit. on p. 98).

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*, 1929–1958. Retrieved January 27, 2019, from http://jmlr.org/papers/v15/srivastava14a.html (cit. on pp. 31, 139)

St. Leger, R. J., & Wang, J. B. (2020). Metarhizium: Jack of all trades, master of many. *Open Biology*, *10*(12), 200307. https://doi.org/10.1098/rsob.200307 (cit. on p. 102)

Starr, T. N., Greaney, A. J., Hilton, S. K., Ellis, D., Crawford, K. H. D., Dingens, A. S., Navarro, M. J., Bowen, J. E., Tortorici, M. A., Walls, A. C., King, N. P., Veesler, D., & Bloom, J. D. (2020). Deep Mutational Scanning of SARS-CoV-2 Receptor Binding Domain Reveals Constraints on Folding and ACE2 Binding. *Cell*, *182*(5), 1295–1310.e20. https://doi.org/10.1016/j.cell.2020.08.012 (cit. on p. 68)

Steinegger, M., & Söding, J. (2017). MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature Biotechnology*, *35*(11), 1026–1028. https://doi.org/10.1038/nbt.3988 (cit. on pp. 8, 15)

Stojkova, P., Spidlova, P., & Stulik, J. (2019). Nucleoid-associated protein hu: A lilliputian in gene regulation of bacterial virulence. *Frontiers in Cellular and Infection Microbiology*, *9*, 159. https://doi.org/10.3389/fcimb.2019.00159 (cit. on p. 68)

Stop neglecting fungi. (2017). *Nature Microbiology*, *2*(8), 1–2. https://doi.org/10.1038/nmicrobiol.2017.120 (cit. on pp. 98, 124)

Storkey, A. (2009). When training and test sets are different: Characterising learning transfer. In C. S. S. Lawrence (Ed.), *Dataset shift in machine learning* (pp. 3–28). MIT Press. http://mitpress.mit.edu/catalog/item/default.asp?ttype=2&tid=11755. (Cit. on p. 16)

Stringer, J. R., Beard, C. B., Miller, R. F., & Wakefield, A. E. (2002). A New Name for Pneumocystis from Humans and New Perspectives on the Host-Pathogen Relationship. *Emerging Infectious Diseases*, *8*(9), 891–896. https://doi.org/10.3201/eid0809.020096 (cit. on p. 99)

Stukenbrock, E. H., Quaedvlieg, W., Javan-Nikhah, M., Zala, M., Crous, P. W., & McDonald, B. A. (2012). Zymoseptoria ardabiliae and Z. pseudotritici, two progenitor species of the septoria tritici leaf blotch fungus Z. tritici (synonym: Mycosphaerella graminicola). *Mycologia*, *104*(6), 1397–1407. https://doi.org/10.3852/11-374 (cit. on p. 103)

Sun, Z., Huang, S., Zhang, M., Zhu, Q., Haiminen, N., Carrieri, A. P., Vázquez-Baeza, Y., Parida, L., Kim, H.-C., Knight, R., & Liu, Y.-Y. (2021). Challenges in benchmarking metagenomic profilers. *Nature Methods*, *18*(6), 618–626. https://doi.org/10.1038/s41592-021-01141-3 (cit. on p. 9)

Sunagawa, S., Mende, D. R., Zeller, G., Izquierdo-Carrasco, F., Berger, S. A., Kultima, J. R., Coelho, L. P., Arumugam, M., Tap, J., Nielsen, H. B., Rasmussen, S., Brunak, S., Pedersen, O., Guarner, F., de Vos, W. M., Wang, J., Li, J., Doré, J., Ehrlich, S. D., … Bork, P. (2013). Metagenomic species profiling using universal phylogenetic marker genes. *Nature Methods*, *10*(12), 1196–1199. https://doi.org/10.1038/nmeth.2693 (cit. on p. 9)

Sundararajan, M., Taly, A., & Yan, Q. (2016). Gradients of Counterfactuals. *CoRR*, *abs/1611.02639*. http://arxiv.org/abs/1611.02639 (cit. on pp. 12, 48)

Sung, G.-H., Hywel-Jones, N. L., Sung, J.-M., Luangsa-ard, J. J., Shrestha, B., & Spatafora, J. W. (2007). Phylogenetic classification of Cordyceps and the clavicipitaceous fungi. *Studies in Mycology*, *57*, 5–59. https://doi.org/10.3114/sim.2007.57.01 (cit. on p. 103)

Szymanski, E., & Calvert, J. (2018). Designing with living systems in the synthetic yeast project. *Nature Communications*, *9*(1), 2950. https://doi.org/10.1038/s41467-018-05332-z (cit. on p. 100)

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*(9), e0222271. https://doi.org/10.1371/journal.pone.0222271 (cit. on pp. 15, 48, 132)

Tamura, M., & D'haeseleer, P. (2008). Microbial genotype–phenotype mapping by class association rule mining. *Bioinformatics*, *24*(13), 1523–1529. https://doi.org/10.1093/bioinformatics/btn210 (cit. on p. 13)

Tan, J., Fang, Z., Wu, S., Guo, Q., Jiang, X., & Zhu, H. (2021). Identify phage hosts from metaviromic short reads based on deep learning and Markov chain model. *bioRxiv*, 2021.03.01.433351. https://doi.org/10.1101/2021.03.01.433351 (cit. on pp. 14, 136)

Tang, Q., Song, Y., Shi, M., Cheng, Y., Zhang, W., & Xia, X.-Q. (2015). Inferring the hosts of coronavirus using dual statistical models based on nucleotide composition. *Scientific Reports*, *5*(1), 17155. https://doi.org/10.1038/srep17155 (cit. on pp. 14, 100)

Tausch, S. H., Loka, T. P., Schulze, J. M., Andrusch, A., Klenner, J., Dabrowski, P. W., Lindner, M. S., Nitsche, A., & Renard, B. Y. (2018). PathoLive - Real time pathogen identification from metagenomic Illumina datasets. *bioRxiv*, 402370. https://doi.org/10.1101/402370 (cit. on pp. 7, 8, 95, 131)

Tausch, S. H., Strauch, B., Andrusch, A., Loka, T. P., Lindner, M. S., Nitsche, A., & Renard, B. Y. (2018). LiveKraken—-real-time metagenomic classification of illumina data. *Bioinformatics*, *34*(21), 3750–3752. https://doi.org/10.1093/bioinformatics/bty433 (cit. on pp. 9, 87, 131)

Taylor, D. L., Hollingsworth, T. N., McFarland, J. W., Lennon, N. J., Nusbaum, C., & Ruess, R. W. (2014). A first comprehensive census of fungi in soil reveals both hyperdiversity and fine-scale niche partitioning. *Ecological Monographs*, *84*(1), 3–20. https://doi.org/10.1890/12-1693.1 (cit. on p. 98)

Taylor, L. H., Latham, S. M., & Woolhouse, M. E. (2001). Risk factors for human disease emergence. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, *356*(1411), 983–989. https://doi.org/10.1098/rstb.2001.0888 (cit. on pp. 3, 102)

Teixeira, M. M., Moreno, L. F., Stielow, B. J., Muszewska, A., Hainaut, M., Gonzaga, L., Abouelleil, A., Patané, J. S. L., Priest, M., Souza, R., Young, S., Ferreira, K. S., Zeng, Q., da Cunha, M. M. L., Gladki, A., Barker, B., Vicente, V. A., de Souza, E. M., Almeida, S., … de Hoog, G. S. (2017). Exploring the genomic diversity of black yeasts and relatives (Chaetothyriales, Ascomycota). *Studies in Mycology*, *86*, 1–28. https://doi.org/10.1016/j.simyco.2017.01.001 (cit. on pp. 102, 103)

Thiel, V. (2018). Synthetic viruses-Anything new? *PLoS pathogens*, *14*(10), e1007019. https://doi.org/10.1371/journal.ppat.1007019 (cit. on pp. 46, 91)

Thorvaldsdóttir, H., Robinson, J. T., & Mesirov, J. P. (2013). Integrative Genomics Viewer (IGV): High-performance genomics data visualization and exploration. *Briefings in Bioinformatics*, *14*(2), 178–192. https://doi.org/10.1093/bib/bbs017 (cit. on p. 59)

Trappe, K., Marschall, T., & Renard, B. Y. (2016). Detecting horizontal gene transfer by mapping sequencing reads across species boundaries. *Bioinformatics*, *32*(17), i595–i604. https://doi.org/10.1093/bioinformatics/btw423 (cit. on pp. 4, 46, 76, 129)

Truong, D. T., Franzosa, E. A., Tickle, T. L., Scholz, M., Weingart, G., Pasolli, E., Tett, A., Huttenhower, C., & Segata, N. (2015). MetaPhlAn2 for enhanced metagenomic taxonomic profiling. *Nature Methods*, *12*(10), 902–903. https://doi.org/10.1038/nmeth.3589 (cit. on pp. 9, 25)

Ulrich, J.-U., Lutfi, A., Rutzen, K., & Renard, B. Y. (2022). ReadBouncer: Precise and Scalable Adaptive Sampling for Nanopore Sequencing. *bioRxiv*, 2022.02.01.478636. Retrieved February 2, 2022, from https://www.biorxiv.org/content/10.1101/2022.02.01.478636v1 (cit. on pp. 6, 132, 135)

Vacher, C., Piou, D., & Desprez-Loustau, M.-L. (2008). Architecture of an Antagonistic Tree/Fungus Network: The Asymmetric Influence of Past Evolutionary History. *PLOS ONE*, *3*(3), e1740. https://doi.org/10.1371/journal.pone.0001740 (cit. on p. 103)

van der Geest, L., Elliot, S., Breeuwer, J., & Beerling, E. (2000). Diseases of Mites. *Experimental & Applied Acarology*, *24*(7), 497–560. https://doi.org/10.1023/A:1026518418163 (cit. on p. 103)

Van Regenmortel, M. H. V. (2018). Chapter One - The Species Problem in Virology. In M. Kielian, T. C. Mettenleiter, & M. J. Roossinck (Eds.), *Advances in Virus Research* (pp. 1–18). Academic Press. https://doi.org/10.1016/bs.aivir.2017.10.008. (Cit. on p. 52)

Van Valen, L. (1973). A new evolutionary law. *Evolutionary Theory*, (1), 1–30 (cit. on p. 129).

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is All you Need. *Advances in Neural Information Processing Systems*, *30*. Retrieved December 14, 2021, from https://papers.nips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html (cit. on pp. 15, 130, 134)

Ventola, C. L. (2015). The Antibiotic Resistance Crisis. *Pharmacy and Therapeutics*, *40*(4), 277–283. Retrieved December 12, 2021, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4378521/ (cit. on pp. 13, 129)

Viehweger, A., Krautwurst, S., Lamkiewicz, K., Madhugiri, R., Ziebuhr, J., Hölzer, M., & Marz, M. (2019). Direct RNA nanopore sequencing of full-length coronavirus genomes provides novel insights into structural variants and enables modification analysis. *Genome Research*, *29*(9), 1545–1554. https://doi.org/10.1101/gr.247064.118 (cit. on p. 135)

Villarroel, J., Kleinheinz, K. A., Jurtz, V. I., Zschach, H., Lund, O., Nielsen, M., & Larsen, M. V. (2016). HostPhinder: A phage host prediction tool. *Viruses*, *8*(5). https://doi.org/10.3390/v8050116 (cit. on p. 14)

Villegas-Morcillo, A., Makrodimitris, S., van Ham, R. C. H. J., Gomez, A. M., Sanchez, V., & Reinders, M. J. T. (2021). Unsupervised protein embeddings outperform hand-crafted sequence and structure features at predicting molecular function. *Bioinformatics*, *37*(2), 162–170. https://doi.org/10.1093/bioinformatics/btaa701 (cit. on p. 13)

Voigt, B., Fischer, O., Krumnow, C., Herta, C., & Dabrowski, P. W. (2021). NGS read classification using AI. *PLOS ONE*, *16*(12), e0261548. https://doi.org/10.1371/journal.pone.0261548 (cit. on p. 16)

von Meijenfeldt, F. A. B., Arkhipova, K., Cambuy, D. D., Coutinho, F. H., & Dutilh, B. E. (2019). Robust taxonomic classification of uncharted microbial sequences and bins with CAT and BAT. *Genome Biology*, *20*(1), 217. https://doi.org/10.1186/s13059-019-1817-x (cit. on p. 8)

Vouga, M., & Greub, G. (2016). Emerging bacterial pathogens: The past and beyond. *Clinical Microbiology and Infection*, *22*(1), 12–21. https://doi.org/10.1016/j.cmi.2015.10.010 (cit. on pp. 4, 46, 76, 129)

Wang, J., Lisanza, S., Juergens, D., Tischer, D., Anishchenko, I., Baek, M., Watson, J. L., Chun, J. H., Milles, L. F., Dauparas, J., Expòsit, M., Yang, W., Saragovi, A., Ovchinnikov, S., & Baker, D. (2021). Deep learning methods for designing proteins scaffolding functional sites. *bioRxiv*, 2021.11.10.468128. Retrieved December 15, 2021, from https://www.biorxiv.org/content/10.1101/2021.11.10.468128v2 (cit. on p. 17)

Wang, W., Ren, J., Tang, K., Dart, E., Ignacio-Espinoza, J. C., Fuhrman, J. A., Braun, J., Sun, F., & Ahlgren, N. A. (2020). A network-based integrated framework for predicting virus–prokaryote interactions. *NAR Genomics and Bioinformatics*, *2*(lqaa044). https://doi.org/10.1093/nargab/lqaa044 (cit. on p. 14)

Wang, Y., Zhao, Y., Bollas, A., Wang, Y., & Au, K. F. (2021). Nanopore sequencing technology, bioinformatics and applications. *Nature Biotechnology*, *39*(11), 1348–1365. https://doi.org/10.1038/s41587-021-01108-x (cit. on p. 136)

Wang, Z., Li, S., You, R., Zhu, S., Zhou, X. J., & Sun, F. (2021). ARG-SHINE: Improve antibiotic resistance class prediction by integrating sequence homology, functional information and deep convolutional neural network. *NAR Genomics and Bioinformatics*, *3*(3), lqab066. https://doi.org/10.1093/nargab/lqab066 (cit. on p. 13)

Wardeh, M., Baylis, M., & Blagrove, M. S. C. (2021). Predicting mammalian hosts in which novel coronaviruses can be generated. *Nature Communications*, *12*(1), 780. https://doi.org/10.1038/s41467-021-21034-5 (cit. on pp. 14, 100)

Wardeh, M., Risley, C., McIntyre, M. K., Setzkorn, C., & Baylis, M. (2015). Database of host-pathogen and related species interactions, and their global distribution. *Scientific Data*, *2*(1), 150049. https://doi.org/10.1038/sdata.2015.49 (cit. on pp. 99, 103, 124, 159)

Wattam, A. R., Davis, J. J., Assaf, R., Boisvert, S., Brettin, T., Bun, C., Conrad, N., Dietrich, E. M., Disz, T., Gabbard, J. L., Gerdes, S., Henry, C. S., Kenyon, R. W., Machi, D., Mao, C., Nordberg, E. K., Olsen, G. J., Murphy-Olson, D. E., Olson, R., … Stevens, R. L. (2017). Improvements to PATRIC, the all-bacterial bioinformatics database and analysis resource center. *Nucleic Acids Research*, *45*, D535–D542. https://doi.org/10.1093/nar/gkw1017 (cit. on pp. 26, 29)

Weimann, A., Mooren, K., Frank, J., Pope, P. B., Bremges, A., & McHardy, A. C. (2016). From Genomes to Phenotypes: Traitar, the Microbial Trait Analyzer. *mSystems*, *1*(6), e00101–16. https://doi.org/10.1128/mSystems.00101-16 (cit. on p. 13)

Whalen, S., Schreiber, J., Noble, W. S., & Pollard, K. S. (2021). Navigating the pitfalls of applying machine learning in genomics. *Nature Reviews Genetics*, 1–13. https://doi.org/10.1038/s41576-021-00434-9 (cit. on p. 16)

Wibbelt, G., Tausch, S. H., Dabrowski, P. W., Kershaw, O., Nitsche, A., & Schrick, L. (2017). Berlin Squirrelpox Virus, a New Poxvirus in Red Squirrels, Berlin, Germany. *Emerging Infectious Diseases*, *23*(10), 1726–1729. https://doi.org/10.3201/eid2310.171008 (cit. on p. 4)

Wightman, R., Touvron, H., & Jégou, H. (2021). ResNet strikes back: An improved training procedure in timm. *arXiv:2110.00476 [cs]*. Retrieved February 9, 2022, from http://arxiv.org/abs/2110.00476 (cit. on p. 134)

Willis, A. (2016). Extrapolating abundance curves has no predictive power for estimating microbial biodiversity. *Proceedings of the National Academy of Sciences*, *113*(35), E5096–E5096. https://doi.org/10.1073/pnas.1608281113 (cit. on p. 24)

Wood, D. E., Lu, J., & Langmead, B. (2019). Improved metagenomic analysis with Kraken 2. *Genome Biology*, *20*(1), 257. https://doi.org/10.1186/s13059-019-1891-0 (cit. on pp. 9, 98, 111)

Wood, D. E., & Salzberg, S. L. (2014). Kraken: Ultrafast metagenomic sequence classification using exact alignments. *Genome Biology*, *15*(3), R46. https://doi.org/10.1186/gb-2014-15-3-r46 (cit. on pp. 9, 25, 32, 54, 77, 95)

Woolhouse, M. E. J., Dye, C., Taylor, L. H., Latham, S. M., & woolhouse, M. E. (2001). Risk factors for human disease emergence. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, *356*(1411), 983–989. https://doi.org/10.1098/rstb.2001.0888 (cit. on p. 3)

Woolhouse, M., & Gaunt, E. (2007). Ecological origins of novel human pathogens. *Critical Reviews in Microbiology*, *33*(4), 231–242. https://doi.org/10.1080/10408410701647560 (cit. on pp. 3, 4)

Woolhouse, M. E. J., & Gowtage-Sequeria, S. (2005). Host range and emerging and reemerging pathogens. *Emerging Infectious Diseases*, *11*(12), 1842–1847. https://doi.org/10.3201/eid1112.050997 (cit. on pp. 3, 129)

Woolhouse, M. E. J., Haydon, D. T., & Antia, R. (2005). Emerging pathogens: The epidemiology and evolution of species jumps. *Trends in Ecology & Evolution*, *20*(5), 238–244. https://doi.org/10.1016/j.tree.2005.02.009 (cit. on pp. 3, 129)

Woolhouse, M. E., Brierley, L., McCaffery, C., & Lycett, S. (2016). Assessing the Epidemic Potential of RNA and DNA Viruses. *Emerging Infectious Diseases*, *22*(12), 2037–2044. https://doi.org/10.3201/eid2212.160123 (cit. on p. 4)

Wrapp, D., Wang, N., Corbett, K. S., Goldsmith, J. A., Hsieh, C.-L., Abiona, O., Graham, B. S., & McLellan, J. S. (2020). Cryo-EM structure of the 2019-nCoV spike in the prefusion conformation. *Science*, *367*(6483), 1260–1263. https://doi.org/10.1126/science.abb2507 (cit. on p. 68)

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., Yuan, M.-L., Zhang, Y.-L., Dai, F.-H., Liu, Y., Wang, Q.-M., Zheng, J.-J., Xu, L., Holmes, E. C., & Zhang, Y.-Z. (2020). A new coronavirus associated with human respiratory disease in China. *Nature*, *579*(7798), 265–269. https://doi.org/10.1038/s41586-020-2008-3 (cit. on p. 49)

Wu, S., Toews, M. D., Castrillo, L. A., Barman, A. K., Cottrell, T. E., & Shapiro-Ilan, D. I. (2021). Identification and Virulence of Cordyceps javanica Strain wf GA17 Isolated From a Natural Fungal Population in Sweetpotato Whiteflies (Hemiptera: Aleyrodidae). *Environmental Entomology*, *50*(5), 1127–1136. https://doi.org/10.1093/ee/nvab061 (cit. on p. 103)

Wu, S., Fang, Z., Tan, J., Li, M., Wang, C., Guo, Q., Xu, C., Jiang, X., & Zhu, H. (2021). DeePhage: Distinguishing virulent and temperate phage-derived sequences in metavirome data with a deep learning approach. *GigaScience*, *10*(9), giab056. https://doi.org/10.1093/gigascience/giab056 (cit. on pp. 14, 136)

Wu, Z., Johnston, K. E., Arnold, F. H., & Yang, K. K. (2021). Protein sequence design with deep generative models. *arXiv:2104.04457 [cs, q-bio, stat]*. Retrieved April 12, 2021, from http://arxiv.org/abs/2104.04457 (cit. on p. 137)

Xia, L. C., Cram, J. A., Chen, T., Fuhrman, J. A., & Sun, F. (2011). Accurate Genome Relative Abundance Estimation Based on Shotgun Metagenomic Reads. *PLOS ONE*, *6*(12), e27992. https://doi.org/10.1371/journal.pone.0027992 (cit. on p. 8)

Xie, R., Li, J., Wang, J., Dai, W., Leier, A., Marquez-Lago, T. T., Akutsu, T., Lithgow, T., Song, J., & Zhang, Y. (2021). DeepVF: A deep learning-based hybrid framework for identifying virulence factors using the stacking strategy. *Briefings in Bioinformatics*, *22*(3), bbaa125. https://doi.org/10.1093/bib/bbaa125 (cit. on p. 13)

Xie, S., Girshick, R., Dollár, P., Tu, Z., & He, K. (2017). Aggregated Residual Transformations for Deep Neural Networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5987–5995. https://doi.org/10.1109/CVPR.2017.634 (cit. on pp. 130, 134)

Xu, B., Tan, Z., Li, K., Jiang, T., & Peng, Y. (2017). Predicting the host of influenza viruses based on the word vector. *PeerJ*, *5*, e3579. https://doi.org/10.7717/peerj.3579 (cit. on pp. 14, 47)

Yang, K. K., Wu, Z., & Arnold, F. H. (2019). Machine-learning-guided directed evolution for protein engineering. *Nature Methods*, *16*(8), 687–694. https://doi.org/10.1038/s41592-019-0496-6 (cit. on p. 137)

Yang, X., Yang, S., Li, Q., Wuchty, S., & Zhang, Z. (2019). Prediction of human-virus protein-protein interactions through a sequence embedding-based machine learning method. *Computational and Structural Biotechnology Journal*, *18*, 153–161. https://doi.org/10.1016/j.csbj.2019.12.005 (cit. on p. 15)

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., Gong, Q., Mei, Y., Xue, T., Zhang, J.-R., Chen, Y., & Zhou, C.-Z. (2014). Structural Insights into SraP-Mediated Staphylococcus aureus Adhesion to Host Cells. *PLOS Pathogens*, *10*(6), e1004169. https://doi.org/10.1371/journal.ppat.1004169 (cit. on p. 65)

Ye, S. H., Siddle, K. J., Park, D. J., & Sabeti, P. C. (2019). Benchmarking Metagenomics Tools for Taxonomic Classification. *Cell*, *178*(4), 779–794. https://doi.org/10.1016/j.cell.2019.07.010 (cit. on p. 100)

You, R., Yao, S., Mamitsuka, H., & Zhu, S. (2021). DeepGraphGO: Graph neural network for large-scale, multispecies protein function prediction. *Bioinformatics*, *37*(Supplement_1), i262–i271. https://doi.org/10.1093/bioinformatics/btab270 (cit. on p. 13)

Yu, F., & Koltun, V. (2016). Multi-Scale Context Aggregation by Dilated Convolutions. In Y. Bengio & Y. LeCun (Eds.), *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*. http://arxiv.org/abs/1511.07122. (Cit. on p. 134)

Yuan, M., Wu, N. C., Zhu, X., Lee, C.-C. D., So, R. T. Y., Lv, H., Mok, C. K. P., & Wilson, I. A. (2020). A highly conserved cryptic epitope in the receptor binding domains of SARS-CoV-2 and SARS-CoV. *Science*, *368*(6491), 630–633. https://doi.org/10.1126/science.abb7269 (cit. on p. 68)

Zeng, H., Edwards, M. D., Liu, G., & Gifford, D. K. (2016). Convolutional neural network architectures for predicting DNA–protein binding. *Bioinformatics*, *32*(12), i121–i127. https://doi.org/10.1093/bioinformatics/btw255 (cit. on pp. 10, 26, 47)

Zhang, K., Schölkopf, B., Muandet, K., & Wang, Z. (2013). Domain adaptation under target and conditional shift. *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, III–819–III–827 (cit. on p. 16).

Zhang, M., Yang, L., Ren, J., Ahlgren, N. A., Fuhrman, J. A., & Sun, F. (2017). Prediction of virus-host infectious association by supervised learning methods. *BMC Bioinformatics*, *18*(3), 60. https://doi.org/10.1186/s12859-017-1473-7 (cit. on p. 14)

Zhang, Z., Cai, Z., Tan, Z., Lu, C., Jiang, T., Zhang, G., & Peng, Y. (2019). Rapid identification of human-infecting viruses. *Transboundary and Emerging Diseases*, *66*(6), 2517–2522. https://doi.org/10.1111/tbed.13314 (cit. on pp. 14, 17, 47–49, 51, 52, 54, 60, 62, 70, 77, 82, 94, 100, 130)

Zhao, H., Tu, Z., Liu, Y., Zong, Z., Li, J., Liu, H., Xiong, F., Zhan, J., Hu, X., & Xie, W. (2021). PlantDeepSEA, a deep learning-based web service to predict the regulatory effects of genomic variants in plants. *Nucleic Acids Research*, *49*(W1), W523–W529. https://doi.org/10.1093/nar/gkab383 (cit. on p. 137)

Zheng, T., Li, J., Ni, Y., Kang, K., Misiakou, M.-A., Imamovic, L., Chow, B. K. C., Rode, A. A., Bytzer, P., Sommer, M., & Panagiotou, G. (2019). Mining, analyzing, and integrating viral signals from metagenomic data. *Microbiome*, *7*(1), 42. https://doi.org/10.1186/s40168-019-0657-y (cit. on p. 15)

Zhou, H., Shrikumar, A., & Kundaje, A. (2022). Towards a Better Understanding of Reverse-Complement Equivariance for Deep Learning Models in Genomics. *Proceedings of the 16th Machine Learning in Computational Biology meeting*, 1–33. Retrieved January 25, 2022, from https://proceedings.mlr.press/v165/zhou22a.html (cit. on pp. 10, 126, 133)

Zhou, J., & Troyanskaya, O. G. (2015). Predicting effects of noncoding variants with deep learning–based sequence model. *Nature Methods*, *12*(10), 931–934. https://doi.org/10.1038/nmeth.3547 (cit. on pp. 10, 12, 26, 47, 78, 138)

Zhou, P., Yang, X.-L., Wang, X.-G., Hu, B., Zhang, L., Zhang, W., Si, H.-R., Zhu, Y., Li, B., Huang, C.-L., Chen, H.-D., Chen, J., Luo, Y., Guo, H., Jiang, R.-D., Liu, M.-Q., Chen, Y., Shen, X.-R., Wang, X., … Shi, Z.-L. (2020). A pneumonia outbreak associated with a new coronavirus of probable bat origin. *Nature*, *579*(7798), 270–273. https://doi.org/10.1038/s41586-020-2012-7 (cit. on pp. 5, 76)

Zhou, X., Park, B., Choi, D., & Han, K. (2018). A generalized approach to predicting protein-protein interactions between virus and host. *BMC Genomics*, *19*(6), 568. https://doi.org/10.1186/s12864-018-4924-2 (cit. on p. 15)

Zielezinski, A., Barylski, J., & Karlowski, W. M. (2021). Taxonomy-aware, sequence similarity ranking reliably predicts phage–host relationships. *BMC Biology*, *19*(1), 223. https://doi.org/10.1186/s12915-021-01146-6 (cit. on pp. 14, 111)

Zielezinski, A., Deorowicz, S., & Gudyś, A. (2021). PHIST: Fast and accurate prediction of prokaryotic hosts from metagenomic viral sequences. *Bioinformatics*, btab837. https://doi.org/10.1093/bioinformatics/btab837 (cit. on pp. 14, 111)

Zielezinski, A., Vinga, S., Almeida, J., & Karlowski, W. M. (2017). Alignment-free sequence comparison: Benefits, applications, and tools. *Genome Biology*, *18*(1), 186. https://doi.org/10.1186/s13059-017-1319-7 (cit. on p. 125)

Zusammenfassung

Das regelmäßige Auftreten neuer Krankheitserreger ist eine der größten Bedrohungen für die globale Gesundheit. DNA- und RNA-Sequenzierung ermöglichen den Nachweis neuer Viren und Mikroben, aber die Standardansätze für die computergestützte Analyse von Sequenzierungsdaten beruhen auf vordefinierten Listen bekannter Erreger. Neue Pathogene, deren Genome stark von den verfügbaren Referenzen abweichen, bleiben schwer zu erkennen.

Dieses Problem kann durch das Training von Klassifikatoren gemildert werden, die vorhersagen, ob ein bestimmter Sequenzierungs-Read von einem möglicherweise neuen Krankheitserreger stammt. In dieser Arbeit zeige ich, dass tiefe neuronale Netze, die invariant gegenüber der DNA-Rückwärtskomplementarität sind, Alternativen, die auf anderen Algorithmen des maschinellen Lernens und der Homologieerkennung durch Sequenzabgleich basieren, deutlich übertreffen. Dies gilt sowohl für Bakterien als auch für Viren. Ich stelle neue Methoden vor, die eine Analyse und Visualisierung der gelernten Muster sowie die Identifizierung von Sequenzen, Genen und genomischen Regionen mit hohem pathogenen Potenzial ermöglichen. Modifizierte ResNet-Architekturen in Kombination mit Echtzeit-Alignierungen von kurzen Reads können sowohl bekannte als auch neuartige Bedrohungen bei laufendem Sequenziervorgang genau erkennen. Ähnliche Modelle funktionieren auch für kurze Fragmente langer Reads, die nur 0,5 s Sequenzierungszeit entsprechen. Anschließend beschreibe ich eine manuell kuratierte Datenbank mit Genomen pathogener Pilze, welche die Erkennung neuartiger Bedrohungen sowohl durch maschinelles Lernen als auch durch alternative Ansätze erleichtert. Ich verwende die erlernten numerischen Repräsentationen der Genome in der Datenbank, um die Beziehung zwischen der Taxonomie und dem pathogenen Phänotyp zu visualisieren. Schließlich setze ich die entwickelten neuronalen Architekturen ein, um Reads zu klassifizieren, die aus Mischungen verschiedener neuartiger Bakterien, Viren und Pilze stammen.

Die vorgestellten Methoden sind in den Paketen DeePaC und DeePaC-Live implementiert. Sie können leicht für das Training, die Bewertung und den Einsatz von tiefen neuronalen Netzen für DNA- und RNA-Sequenzen wiederverwendet werden. Obwohl der Schwerpunkt auf der Identifizierung neu auftretender Krankheitserreger anhand von Sequenzierungsdaten liegt, könnten die vorgestellten Ansätze auch für das Screening synthetischer Sequenzen und die Erkennung manipulierter Bedrohungen verwendet werden. Die trainierten Netze sind in der Lage, abstrakte und komplexe Eigenschaften direkt aus Sequenzen vorherzusagen, ohne dabei auf enge taxonomische Übereinstimmungen angewiesen zu sein. In Zukunft könnten ähnliche "Phänotypmodelle" viele alternative Anwendungen in der Schnelldiagnostik, der öffentlichen Gesundheit und der synthetischen Biologie finden.

Selbstständigkeitserklärung

Name: Bartoszewicz
Vorname: Jakub Maciej

Ich erkläre gegenüber der Freien Universität Berlin, dass ich die vorliegende Dissertation selbstständig und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel angefertigt habe. Die vorliegende Arbeit ist frei von Plagiaten. Alle Ausführungen, die wörtlich oder inhaltlich aus anderen Schriften entnommen sind, habe ich als solche kenntlich gemacht. Diese Dissertation wurde in gleicher oder ähnlicher Form noch in keinem früheren Promotionsverfahren eingereicht.

Mit einer Prüfung meiner Arbeit durch ein Plagiatsprüfungsprogramm erkläre ich mich einverstanden.

Jakub Maciej Bartoszewicz, Berlin, 9. März 2022