From Department of Laboratory Medicine
Karolinska Institutet, Stockholm, Sweden

MACHINE LEARNING AND DATA-PARALLEL PROCESSING FOR VIRAL METAGENOMICS

Zurab Bzhalava

Karolinska Institutet

Stockholm 2020

Published by Karolinska Institutet.
Printed by Arkitektkopia AB, 2020

ISBN 978-91-7831-708-0

Machine Learning and Data-Parallel Processing for Viral Metagenomics

THESIS FOR DOCTORAL DEGREE (Ph.D.)

The thesis will be defended at Månen 9Q, Alfred Nobels allé 8 (Floor 9), Karolinska Institutet, Campus Fleminsberg, Huddinge.

Friday, April 3, 2020, at 9:00 AM

By

Zurab Bzhalava

Principal Supervisor:

Professor Joakim Dillner
Karolinska Institutet
Department of Laboratory Medicine
Division of Pathology

Co-supervisor(s):

MD PhD Karin Sundström
Karolinska Institutet
Department of Laboratory Medicine
Division of Pathology

Professor Piotr Bała
University of Warsaw
Interdisciplinary Centre for Mathematical and Computational Modelling

Opponent:

Ola Spjuth
Uppsala University
Department of Pharmaceutical Biosciences

Examination Board:

Panagiotis Papapetrou
Stockholm University
Department of Computer and Systems Sciences

Tobias Allander
Karolinska Institutet
Department of Microbiology, Tumor and Cell Biology

Jim Dowling
KTH Royal Institute of Technology
Division of Software and Computer Systems

To my family and friends

ABSTRACT

More than 2 million cancer cases around the world each year are caused by viruses. In addition, there are epidemiological indications that other cancer-associated viruses may also exist. However, the identification of highly divergent and yet unknown viruses in human biospecimens is one of the biggest challenges in bioinformatics. Modern-day Next Generation Sequencing (NGS) technologies can be used to directly sequence biospecimens from clinical cohorts with unprecedented speed and depth. These technologies are able to generate billions of bases with rapidly decreasing cost but current bioinformatics tools are inefficient to effectively process these massive datasets. Thus, the objective of this thesis was to facilitate both the detection of highly divergent viruses among generated sequences as well as large-scale analysis of human metagenomic datasets.

To re-analyze human sample-derived sequences that were classified as being of "unknown" origin by conventional alignment-based methods, we used a methodology based on profile Hidden Markov Models (HMM) which can capture evolutionary changes by using multiple sequence alignments. We thus identified 510 sequences that were classified as distantly related to viruses. Many of these sequences were homologs to large viruses such as Herpesviridae and Mimiviridae but some of them were also related to small circular viruses such as Circoviridae. We found that bioinformatics analysis using viral profile HMM is capable of extending the classification of previously unknown sequences and consequently the detection of viruses in biospecimens from humans.

Different organisms use synonymous codons differently to encode the same amino acids. To investigate whether codon usage bias could predict the presence of virus in metagenomic sequencing data originating from human samples, we trained Random Forest and Artificial Neural Networks based on Relative Synonymous Codon Usage (RSCU) frequency. Our analysis showed that machine learning techniques based on RSCU could identify putative viral sequences with area under the ROC curve of 0.79 and provide important information for taxonomic classification.

For identification of viral genomes among raw metagenomic sequences, we developed the tool ViraMiner, a deep learning-based method which uses Convolutional Neural Networks with two convolutional branches. Using 300 base-pair length sequences, ViraMiner achieved 0.923 area under the ROC curve which is considerably improved performance in comparison with previous machine learning methods for virus sequence classification. The proposed architecture, to the best of our knowledge, is the first deep learning tool which can detect viral genomes on raw metagenomic sequences originating from a variety of human samples.

To enable large-scale analysis of massive metagenomic sequencing data we used Apache Hadoop and Apache Spark to develop ViraPipe, a scalable parallel bioinformatics pipeline for viral metagenomics. Comparing ViraPipe (executed on 23 nodes) with the sequential pipeline (executed on a single node) was 11 times faster in the metagenome analysis. The new distributed workflow contains several standard bioinformatics tools and can scale to terabytes of data by accessing more computer power from the nodes.

To analyze terabytes of RNA-seq data originating from head and neck squamous cell carcinoma samples, we used our parallel bioinformatics pipeline ViraPipe and the most recent version of the HPV sequence database. We detected transcription of HPV viral oncogenes in 92/500 cancers. HPV 16 was the most important HPV type, followed by HPV 33 as the second most common infection. If these cancers are indeed caused by HPV, we estimated that vaccination might prevent about 36 000 head and neck cancer cases in the United States every year.

In conclusion, the work in this thesis improves the prospects for biomedical researchers to classify the sequence contents of ultra-deep datasets, conduct large-scale analysis of metagenome studies, and detect presence of viral genomes in human biospecimens. Hopefully, this work will contribute to our understanding of biodiversity of viruses in humans which in turn can help exploring infectious causes of human disease.

LIST OF SCIENTIFIC PAPERS

This thesis is based on the following publications:

I. BZHALAVA Z, Hultin E, Dillner J. Extension of the viral ecology in humans using viral profile hidden Markov models. Plos ONE. 2018; 13(1):1–12

II. BZHALAVA Z#, Tampuu A#, Bała P, Vicente R, Dillner J. Machine Learning for detection of viral sequences in human metagenomic datasets. BMC Bioinformatics, 2018. 19(1): p. 336

III. Tampuu A#, BZHALAVA Z#, Dillner J, Vicente R. ViraMiner: Deep learning on raw DNA sequences for identifying viral genomes in human samples. PLOS ONE, 2019;14(9): e0222271

IV. Maarala AI, BZHALAVA Z, Dillner J, Heljanko K, Bzhalava D. ViraPipe: scalable parallel pipeline for viral metagenome analysis from next generation sequencing reads. Bioinformatics, Volume 34, Issue 6, 15 March 2018, Pages 928–935

V. BZHALAVA Z, Arroyo Mühr LS and Dillner J. Transcription of Human Papillomavirus Oncogenes in Head and Neck Squamous Cell Carcinomas. Manuscript

# Equal contributions

CONTENTS

LIST OF ABBREVIATIONS

HPV	Human Papillomavirus
EBV	Epstein-Barr Virus
HCV	Hepatitis C Virus
KSHV	Kaposi's Sarcoma Herpes Virus
HTLV-1	Human T-cell Lymphotropic virus
HIV-1	Human Immunodeficiency Virus type-1
NGS	Next Generation Sequencing
HBV	Hepatitis B Virus
BCC	Basal Cell Skin
SCC	Squamous Cell Skin
NMSC	Non-Melanoma Skin Cancers
PCR	Polymerase Chain Reaction
SIR	Standardized Incidence Ratio
HNSCC	Head and Neck Squamous Cell Carcinomas
DBG	de Bruijn Graphs
OLC	Overlap Layout Consensus
HDFS	Hadoop Distributed File System
API	Application Programming Interface
RDD	Resilient Distributed Dataset
KDN	k-Disagreeing Neighbors
RF	Random Forest
ANN	Artificial Neural Networks
FFNN	Feed-Forward Neural Networks
CNN	Convolutional Neural Networks
HMM	Hidden Markov Models
TCGA	The Cancer Genome Atlas
FFPE	Formalin-Fixed Paraffin-Embedded
RNA-seq	RNA Sequencing
RSCU	Relative Synonymous Codon Usage
LOEO	Leave-one-experiment-out
AUROC	Area Under Receiver Operating Characteristic
PPV	Predictive Positive Value

1 BACKGROUND

1.1 Tumor viruses

Viruses are abundant and ubiquitous microscopic organisms that lack the ability to replicate ex vivo and are therefore inactive outside of host cells. Once they infect the cells, they can employ the cellular machinery in order to reproduce more virus particles. Their genetic material is composed of a subset of genes in the form of DNA or RNA enclosed in a protective protein coat. Both DNA and RNA viruses have shown abilities to disrupt and engage host cells important regulatory mechanisms which may transform the host cells into cancer [1].

In humans, several viruses, such as human papillomavirus (HPV), Epstein-Barr virus (EBV), hepatitis C (HCV) and hepatitis B (HBV), Kaposi's sarcoma herpesvirus (KSHV), human T-cell lymphotropic virus (HTLV-1), human immunodeficiency virus type-1 (HIV-1) has been linked to human carcinogenesis [1, 2]. While HPV, EBV, HTLV-1, and KSHV are directly associated with cancer development, HCV and HBV are indirectly involved in the cellular transformation through chronic inflammation. HIV-1, on the other hand, increases the chance of cancer by immunosuppression [3]. Even though it might be convenient to think that these viruses belong to one particular group of cancer-associated viruses, they are in fact very different from each other. They represent diverse virus families, have different genomes and life cycles and display different strategies to contribute to tumor development [4]. Having said that, they also share some common characteristics such as their strategy to infect the host cell and persist, instead of killing it as well as their ability to somehow avoid the host immune system, which would otherwise overcome the virus [4].

The International Agency for Research on Cancer has estimated that approximately 2.2 million (15.4%) of 14 million cancers in humans around the world are caused by viral infections [5]. In addition to that, recent epidemiological studies also provided some epidemiological indications that other cancer-related viruses may exist. For instance, increased use of organ transplantation over the last decades has led to the conclusion that not only the virus-associated cancers are increased among immunosuppressed individuals but also some cancers without known viral etiology [6-8]. There is also some evidence that pathogens might be involved in the development of childhood leukemias [9] as well as in autoimmune diseases such as diabetes [10] and multiple sclerosis [11].

To study possible connections between viruses and diseases is therefore very important. The progress, however, has been very slow so far as most studies usually focus only on one infectious agent or one cancer type at a time. In recent years, access to recent Next Generation Sequencing (NGS) technologies has provided

a powerful tool to conduct complex genetic studies in human specimens. With complete sequencing of all microbiological sequences, we can detect and analyze all known and unknown viruses that might be present in human biospecimens. The challenge, however, is that NGS machines generate huge amounts of sequencing datasets that require powerful computational algorithms and resources for processing and detecting the target sequences. If these algorithms are about to improve it is likely that more oncogenic viruses will be revealed, which in turn ultimately could facilitate potential prevention of the diseases.

1.2 Viruses in cancers among immunosuppressed patients

During the last 30 years, studies of immunosuppressed individuals after organ transplantation and patients living with HIV have shown that these patients have a much higher risk of cancer compared to the general population [6, 7, 12]. There are some cancer types such as prostate, breast, corpus uteri, and brain cancer with no clear evidence of increase after immunosuppression, but the majority of cancers are significantly increased in immunosuppressed patients, including cancers with no established viral etiology [7, 12, 13]. This gives rise to a hypothesis that many novel human carcinogenic viruses may yet be discovered in these patients.

One example of a high incidence rate is non-melanoma skin cancers (NMSC), where subsequent tumors among transplant recipients have been documented [14, 15]. Basal cell skin (BCC) cancers include approximately 80% of all NMSC while squamous cell skin (SCC) cancers comprise up to 20% [16]. Although BCC is approximately 5 times more common than SCC in the general population, the incidence ratio is reversed among immunosuppressed individuals with reported 18- to 250 times increase [17, 18]. Viral metagenomic sequencing analysis showed that HPV comprises approximately 95% of total viral reads but there is no agreement which HPV types are the most common [19]. This contrasts the situation of cervical cancers where mostly HPV-16 and HPV-18 are detected [20]. Most of these studies, however, have been conducted using polymerase chain reaction (PCR) systems that are biased to detecting only viral sequences that share high similarity with the used PCR primers. Viral genomes that are different from the a priori defined primers might thus have been entirely missed by these studies [21].

Among the cancers with no known viral etiology, cancer of the lip also shows one of the highest incidence rates after immunosuppression [22]. The oncogenic process of this cancer is casually linked to smoking, exposure to solar UV radiation [23] as well as HPV infection[24] but the evidence is still insufficient and further research is required to confirm the associations.

Interestingly, investigation based on Swedish immunosuppressed cohorts showed that the standardized incidence ratio (SIR) of overall cancer was 3.5 (95% confidence interval: 3,4–3.7) among transplant recipients compared to the general population. More specifically, the increase was particularly significant in cancer of the kidney (SIR=5.8), thyroid (SIR=4.9), NMSC (44.7), lip (SIR=41.5), and larynx (SIR=3) [13]. Investigation of these cancers and their infectious etiology is thus a high priority.

1.3 Human Papillomavirus

Human papillomaviruses have a double-stranded circular DNA genome originating from the Papillomaviridae family. Papillomaviruses are a diverse group of viruses and can infect most mammals and birds. Their isolates were usually described as "types". As the number of identified types increased over time it became necessary to have a taxonomic classification within the family [25]. HPVs are classified based on the nucleotide sequence of the L1 open reading frame which codes the major capsid protein in the genome. If two HPV genomes share less than 60% similarity between their L1 region of the genome they subsequently belong to two different genera. Sharing 60%–70% genome homology means that those viruses are in the same genus. Novel HPV types, however, share less than 90% similarity to other types [25].

To this date, there are 223 officially established HPV types that have been cloned, sequenced and have an approved identifier number at the International HPV Reference Center (https://www.hpvcenter.se/). HPV types such as HPV-16, 18, 31, 33, 35, 39, 45, 52, 58, 59 are established as oncogenic out of which HPV16 and 18 are responsible for the majority of HPV-related cancers [2].

HPV genome is divided into 8 regions (L1, L2, E1, E2, E4, E5, E6, and E7) from which E6 and E7 are viral oncogenes accounting for malignant transformation (Figure 1). These two genes are capable to bind and inactivate the tumor suppressor proteins p53 and pRb which ultimately leads to malignant transformation of the host cells [26, 27].

Besides cervical cancer, HPVs can cause other genital tumors such as anal (over 90%), penile (over 60%), vaginal (75%) and vulvar cancers (70%)[28]. Oncogenic HPVs are also detected in some portions of head and neck squamous cell carcinomas (HNSCCs). Systematic review of 148 studies that used PCR for detecting HPV DNA showed 29.5% (95% confidence interval: 25.5–33.6) HPV positivity in HNSCCs [29]. However, methodologies and results differed from country to country. In comparison, in study V included in this thesis that was conducted based on RNA sequencing data, we identified viral HPV oncogenic transcriptions in 92/500 samples from HNSCCs.

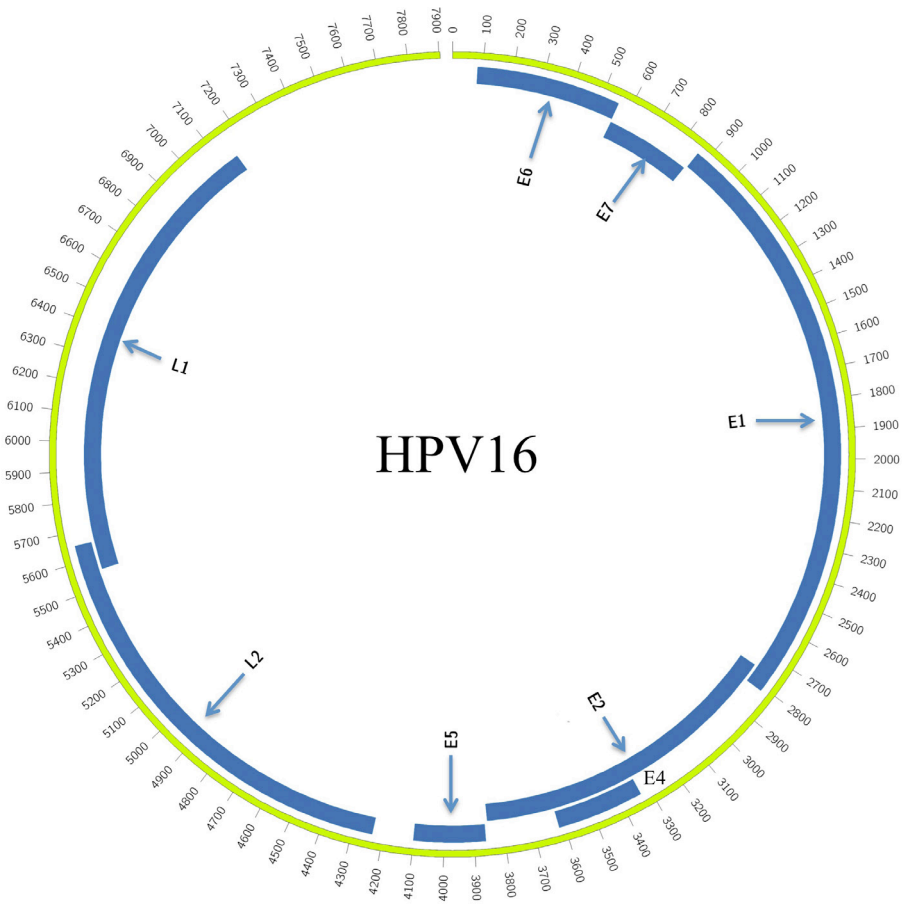

Figure 1. Circular genome of HPV16 with its eight coding genes. *Adapted with permission of Dr. Davit Bzhalava*

1.4 Prophylactic cancer vaccines

One of the most important functions of the immune system is to constantly monitor the body for the intrusions of pathogens, the balance of flora or transformed and abnormal cells. This process is called immunosurveillance [30] which could be strengthened through vaccination to avoid the initial infection that otherwise could lead to the development of an oncogenic process. After vaccination, the body can produce antibodies that can bind with the infectious agents and prevent them from infecting other cells [31].

Currently, there are highly effective vaccines available against oncoviruses such as hepatitis B virus (HBV) and human papillomavirus (HPV). These vaccines can provide protection against persistent infection and related invasive cancers. Long-term evaluation of hepatitis B immunization programs in different countries revealed that adults who were offered vaccination had 76% decrease of HBV infection compared to the cohorts for whom vaccine programs were not available[32]. In the case of HPV vaccination programs, a systematic review showed 83% lower prevalence of HPV16 and HPV18 infections among 15-19 old girls and 66% lower prevalence among 20-24 old women compared to the pre-vaccination period [33].

In order to develop and implement similar vaccination programs against a novel oncovirus, several criteria have to be met: there has to be a direct association between a virus and cancer and there has to be sufficient demand from the public [34]. While the implementation of vaccines against HBV and HPV has been successful and are commercially available at this moment, vaccine developments against HCV and EBV have had limited success so far as none of the vaccine candidates have been effective enough to be approved and licensed for public [34].

Meanwhile, research for the identification of more cancer-associated viruses continues in order to enable the prevention of infections and the related oncogenic process. Perhaps, some of the human viruses that are yet be discovered can also become target of vaccination which would help us to eliminate some specific forms of cancer from the general public.

1.5 Viral metagenomics

The term metagenomics is defined as direct analysis of all genetic material present in a sample [35]. Viral fraction of the human microbiome is referred to as the human virome or viral metagenomics [36]. Even though the human virome is able to seriously impact human health, it usually includes less than 1% of all genomes contained in biospecimens [37]. Research on viral metagenomics therefore requires complete and unbiased sequencing of all genome material from human biospecimens to recover viral-related genomes.

NGS technologies have shown the ability to sequence biospecimens at unprecedented speed allowing great scientific discoveries and new biological applications. The term “next-generation” refers to the new sequencing methods and technologies which emerged after Sanger sequencing methods that dominated the field for several decades since the late 1970s. [38, 39]. These newer technologies offer deep, extremely high-throughput and massively parallel analysis from multiple samples with much lower cost [38]. Compared to Sanger sequencing, NGS technologies generated much shorter reads (number of continuous sequenced nucleotides). The shorter read

lengths are produced by breaking DNA or cDNA samples into smaller pieces and attaching adapters to the ends of the fragments during library preparation [40, 41]. Currently, there are several NGS instruments available such as SOLiD (ABI), Ion Torrent (Life Technologies) and Genome Analyzer System (Illumina). The ability of these instruments to generate vast amounts of sequencing data provides possibilities to conduct large-scale studies on human samples including studies on viruses that are present in cancer samples. NGS technologies are already having a striking impact on the field as they are routinely used for virus detection and discovery in metagenomic samples [10, 42, 43]. However, data storage and analyses of the massive amounts of produced datasets is a significant challenge. It is thus essential to develop advanced bioinformatics tools and algorithms in order to create successful applications for viral diagnostics and research.

1.6 Bioinformatics for viral metagenomics

Even though NGS technologies revolutionized the field because of their high speed and low-cost sequencing abilities, processing of generated short reads by the technology remains one of the biggest challenges in bioinformatics for several reasons. First and foremost, the machines can produce a vast amount of data per sample varying from hundreds of GBs to TBs of sequencing reads and without proper bioinformatics pipelines, just the processing of the data can create a major bottleneck in biomedical studies. Secondly, raw read datasets usually include low-quality bases, possible artifacts produced during library preparation and sequencing bias where particular regions of genomes are better covered or represented by the fragments than others. *Coverage* is particularly important for genome assembly algorithms in order to reconstruct the original genomes from samples. If some parts are over- or underrepresented, incorrect results and conclusions might be inevitable without proper processing of the dataset. After raw reads are preprocessed and assembled into contiguous sequences (contigs), then arises a challenge of taxonomy classification especially when it comes to identifying distant homologs or yet unknown viruses. As conventional taxonomy classification algorithms just align new sequences against public databases, it is almost impossible to detect a novel virus if a similar genome does not exist in the database.

To overview available tools and challenges in the viral metagenomics, the bioinformatics pipeline is divided into two parts: *processing* which includes every step from processing raw reads to assembly algorithms and *taxonomy classification* which deals with annotating assembled contigs (Figure 2).

1.6.1 Processing of NGS dataset

Processing of NGS datasets usually starts with demultiplexing, adaptor removal and quality check of raw reads based on their Phred quality scores, which are widely used quality values of the sequenced nucleobases [44]. Phred score is defined as a value that is logarithmically related to the estimated probabilities of base-calling errors. This implies that if, for instance, a base quality score is 10, one base in one hundred is anticipated to be inaccurate (90% accuracy); a quality score of 40 would imply one inaccurate base in 10 thousand (99.99% accuracy)[44].

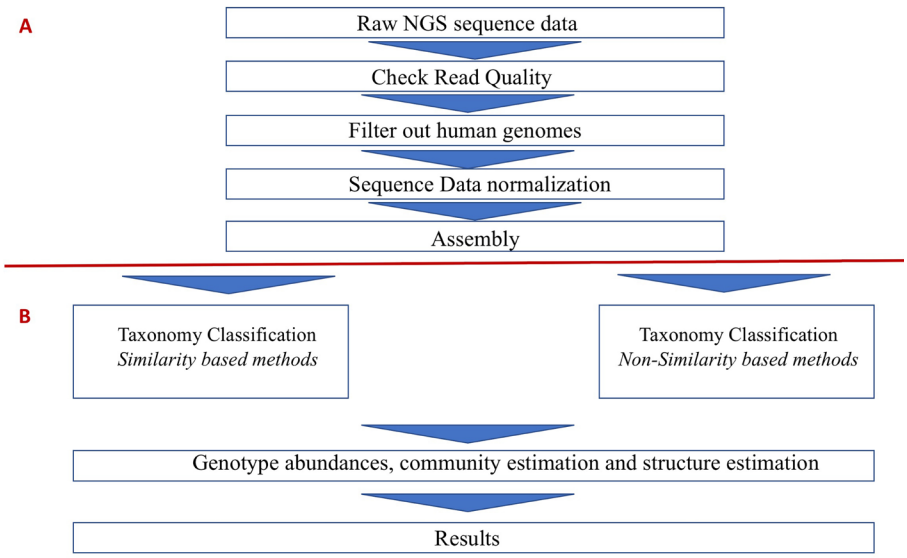

Figure 2. Bioinformatics pipeline to analyze NGS dataset for viral metagenomics. *Part A represents steps in processing of NGS data to make sequences ready for Part B – Taxonomy classification.*

NGS dataset sequenced from human samples usually contain more 70% human-related genomes while viruses are less than 1% [37, 45]. To obtain the dataset that includes virus-related sequences, reads that are not a target of investigation should be filtered out in order to speed up analysis and decrease the chance of assembling erroneous “chimeric” sequences [46].

Table 1. Taxonomy classification of NGS reads (%) from different sample types. Adapted from Bzhalava et al., Unbiased approach for virus detection in skin lesions, in PloS One, 2013; 8:e65953, with permission from the Creative Commons Attribution License

	FFPE Biopsies	Biopsies	Skin swabs	Serum	Water
Human	37.3	99.8	69.1	75	2.8
Bacteria	21.3	0.1	24.2	1	52.2
Virus	0.2	0	0.3	0.1	0
Other	10.2	0	2.2	0.5	15.5
Uknown	30.9	0	4.2	24.4	29.5

The next step in the bioinformatics pipeline is to normalize sequencing data. As mentioned above, NGS can produce sequencing bias which can result from both sample preparation and randomly sampled DNA molecules for sequencing [47]. Obtaining lower abundant DNA/RNA molecules from a sample requires deep sequencing which might as well increase the coverage of higher abundant molecules. For example, sequencing from human samples produces greater coverage of human reads than we usually need for assembly. This is especially true in the case of viral metagenomics where the main purpose is to detect viral genomes. To reduce sample variation and remove highly redundant data, a digital sequence normalization [47] algorithm can be applied to normalize the sequence datasets. The algorithm estimates the distribution of *k*-mer abundance. *k*-mer is a certain length (*k*) of DNA word and more times a specific part of genome is sequenced, higher the *k*-mer abundance is observed from that part [47]. Reads whose estimated coverage is above the threshold can be discarded which would decrease and normalize the average coverage of NGS dataset. Afterwards, normalized datasets are much easier to process for assembly algorithms as they require less computational power because of their significantly reduced size.

The process of merging millions of reads into longer contiguous sequences (contigs) without a reference genome is called *de novo* assembly. Mainly, there are two types of *de novo* assembly algorithms: de Bruijn graphs (DBG) and overlap layout consensus (OLC) assemblers. Algorithms based on de Bruijn split short reads even shorter fragments using *k*-mer approach where nodes are formed with *k*-mers whereas edges are made with overlapping (*k*-1)-mers [48, 49]. With the OLC approach, however, nodes are formed with the reads themselves and edges are the sequences that overlap between these reads [49]. Both approaches have their own advantages and disadvantages but in general, OLC algorithms are more suitable for longer reads while DBG algorithms are more suitable for shorter reads [50].

For all these steps mentioned above, there are highly effective, open source algorithms available. For example, to subtract low quality reads Trimmomatic can be useful for Illumina single-ended or pair-end datasets [51]. To remove redundant, high-coverage reads from NGS dataset, a digital normalization algorithm can be used [47]. To assemble short reads into longer sequences, there are several options available including Trinity [52], SOAPdenovo [53], IDBA-UD[54] and MEGAHIT [55]. However, all these bioinformatics tools belong to traditional sequential algorithms that are computationally inefficient and inflexible to perform large-scale analysis on NGS datasets. These algorithms usually are a mixture of several command line tools that can only be executed on a single computer and the whole process can be extremely time-consuming which can create a major bottleneck in viral metagenomics.

1.6.2 Data-parallel processing

Nowadays, distributed computing frameworks have the potential to accelerate computing speed of metagenomic analyses and meet the requirements of fast-developing metagenomic research. These frameworks are designed to distribute large datasets across multiple cluster nodes and enable several processors to execute the same tasks simultaneously. This empowers reliable, scalable and efficient way of computing in server clusters which brings huge performance advantages compared to executing algorithms on standalone machines.

Apache Hadoop

Apache Hadoop (https://hadoop.apache.org/) is an open source distributed computing framework that can be installed on a commodity Linux cluster to process vast amount of data. Core components of Hadoop include the Hadoop Distributed File System (HDFS) [56], a fault-tolerant distributed file system that allows high throughput of data access and MapReduce, an execution engine that allows programmers to process large datasets in parallel [57]. HDFS stores data by splitting it into smaller blocks and distributes them over the entire cluster. These data blocks are written onto the local disks of each node which enables MapReduce to effectively move the computation where the data is located. MapReduce, in general, divides a big computational program into various independent tasks across many machines where it executes a combination of Map and Reduce functions. The job of Map function is to filter and sort input files whereas Reduce performs aggregation or summary operations at the end. This strategy reduces the network traffic and significantly accelerates the performance of processing large data [58]. Apache Hadoop is recognized as one of the leading technologies for big data solutions and several bioinformatics applications have already been developed on top of Hadoop to deal with massive biological datasets including metagenomic sequencing data.

Cloudbrush [59] is a distributed genome assembler that relies on Hadoop MapReduce programming model and de-Bruijn string graphs. This de-novo assembler also provides an edge-adjustment algorithm to identify and fix structural defects in string graphs.

Halvade [60] was also implemented based on the MapReduce framework to execute tasks simultaneously for parallel variant discovery workflow. It supports both whole genome and whole exome sequencing data and is developed based on GAKT's[61] variant calling pipeline.

For sequence file management, the Hadoop-BAM Java library [62] was developed. The library provides a convenient API for scalable manipulation of BAM (Binary Alignment/Map) files and operates as an integration layer between an application and HDFS where files are stored.

Apache Spark

Apache Spark [63] is an open-source software framework offering high-level Application Programming Interfaces (API) for data processing in parallel. Spark is based on Resilient Distributed Datasets (RDDs) which are collection of objects partitioned across many nodes in a computing cluster. RDDs can be cashed in-memory and reused in parallel operations.

The key difference between Apache Spark and Hadoop MapReduce is that while Spark keeps and process large data in-memory by utilizing RDD abstraction, the MapReduce performs disk-based computations. Comparing the two approaches in terms of speed, Spark achieves approximately 100 times faster performance [64, 65]. Spark supports advanced APIs in several programming languages including Python, JAVA, SCALA and R. It also provides many advanced modules such as GraphX for constructing and computing graphs, Spark SQL for structured data processing and MLlib for machine learning. These frameworks are now widely used for processing of NGS datasets [64].

ADAM is Apache Spark based distributed processing pipeline for exploring genomic data [66]. It supports a command line interface as well as an application programming interface for processing sequencing datasets on a Spark cluster. ADAM provides various algorithms for genome sequencing including for variant calling, genome file transformation, *k*-mer counting.

The Genome Analysis Toolkit (GAKT) [61] is developed by Broad Institute for variant discovery in high-throughput sequencing data. Some tools from the GATK4 version are developed on Spark enabling large-scale genomic studies by reducing execution time.

There are also Sparkhit [67] and MetaSpark [68] available which can be launched on a Spark cluster and can offer several tools for short read processing. Additionally, some studies have directly integrated existing bioinformatics algorithms into Spark framework instead of re-implementing the same tool in Spark [69, 70].

For assembling NGS reads, Spaler [71] was proposed. It utilizes Spark and Graphx APIs for de Bruijn graph construction. When compared previous assemblers based on message passing interface (MPI), the results showed that the algorithm based on Graphx had better performance regarding scalability and was able to produce similar or better contig quality [64]. However, Spaler source code has not been made publicly available so far.

In general, Spark and Hadoop can offer great means and capabilities to process NGS data with more scalable and flexible way. These computing frameworks could prevent bottlenecks in biomedical studies that are created by huge amount of generated data.

1.6.3 Taxonomy classification

Another great challenge in the bioinformatics workflow is taxonomic classification of NGS data. Usually, the identification of potential viral sequences is accomplished by NCBI BLAST, which compares sequences to reference genomes in its database and estimates how much similarity they share. BLASTn conducts searches on nucleotide level whereas BLASTx and tBLASTx queries against a protein database to detect similarities between sequences.

However, a large portion of the sequences from NGS projects is still labeled as unknown [37, 45]. One of the reasons might be that public databases are incomplete which is especially true for viral sequences.

Identifying novel viruses is particularly a challenging task because of the lack of "marker gene". For example, 16S rRNA and 18S rRNA can be used to detect bacteria or eukaryotic genomes in metagenomic sequences but such an approach is not possible for viral organisms. In addition, it is also very difficult to find similarities among viral species. This was further demonstrated by Soueidan et al. when the authors compared archaea, bacteria, plants, and viral genomes by counting k-Disagreeing Neighbors (KDN) among each species. KDN counts a number of neighbors (k) in a genome that does not share its label. According to the study, highest number of KDNs were found in viruses compared to the other species [72].

Alignment-free taxonomic classification methods can be used to explore and compare genome sequence compositions based on codon or *k*-mer usage. These algorithms can help classifying sequences that are highly divergent or have no

homologs in public databases. In addition, they are computationally less expensive and faster compared to alignment-based methods. Although, they usually classify sequences with lower accuracy and heavily depend on sequence length. Developing alignment-free methods for taxonomic classification is a new area of research and up to this time, there have been a very few such algorithms designed for viral genome classification in metagenomic sequencing datasets [73]. Since current genomic reference databases are incomplete, especially for viral sequences, accurate evaluation of viral genomes in metagenomics is a major challenge. It is, therefore essential to develop sophisticated bioinformatics tools and methods to analyze viral metagenomic datasets and explore the biodiversity of viruses.

1.6.4 Machine learning

Machine learning is a branch of Artificial Intelligence that enables algorithms to learn and build models from previous observations in order to make predictions about new independent data. As machine learning can learn from very complex and noisy datasets, it is increasingly applied in natural sciences including bioinformatics for metagenomic sequencing data [74-79]. Machine learning field represents a wide range of algorithms but for this thesis, we only used *supervised learning* algorithms which involve teaching the model with a collection of data containing correct input-output pairs. Supervised learning can be further divided into classification and regression tasks [80]. In this thesis, Random Forest, Feed-Forward Neural Networks and Convolutional Neural Networks were used to build binary classifiers (virus/non-virus). In one study, we also employed an algorithm based on Hidden Markov Models to identify and classify highly divergent viral sequences into different viral families.

Random Forest

Random Forest (RF) [81] is a collection of many decision trees. Each decision tree starts with the root node which branches into leaf nodes – the point where the tree is not split anymore. The path between the root and leaf node is called a classification rule. Every decision tree in RF is constructed by randomly selected observations and variables which makes each tree a biased classifier as they capture different trends of data. In the final decision of RF, however, the majority of votes from these decision trees determines a classification label [81].

RF is one of the most widely-used algorithms in the machine learning field. It has also become very popular in biomedical sciences mainly because it can achieve high prediction accuracy with a large number of features and relatively few observations. Another reason for its popularity is that it can output information about feature importance for classification with easy interpretability [82, 83]. To put it simply, it is straightforward to interpret which features are most decisive in the classification model. This interpretability and feature importance can be

used to identify important biomarkers [84], risk-associated SNPs in genome-wide association study [85] or to just remove features that are not very informative[86].

Artificial Neural Networks

Artificial Neural Networks (ANN) are computational models initially inspired by biological neural structures in the brain. ANN include different kinds of networks and the simplest among them is Feed-Forward neural networks. It consists of layers of "neurons" also called nodes. Multi-layered neural networks are also referred as Deep Learning. While there are no connections between nodes from the same layer, nodes in neighboring layers have all-to-all connections (edges) with each other. All these edges have weights associated with them [87]. In the beginning, all the weights are randomly assigned but later updated to reduce the error at the output layer. This process is called training. After the network is trained, it takes data points as input, processes it with a mathematical function and outputs result values (Figure 3).

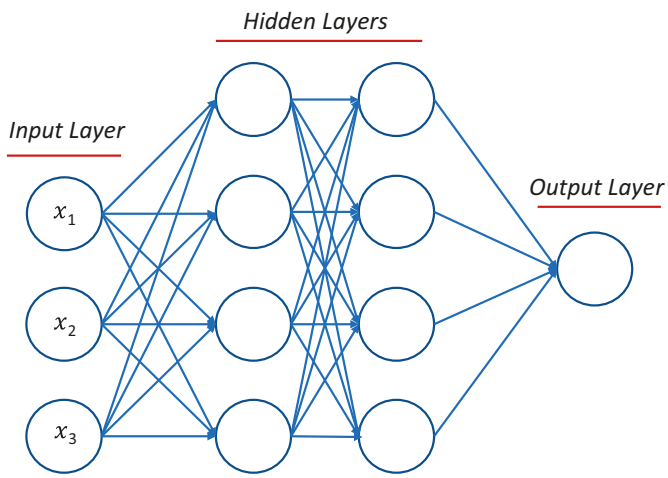

Figure 3. Example of a feed-forward neural network. *It receives data with the input layer, performs computations with the hidden layers and outputs a result value at the output layer.*

Convolutional neural networks (CNN) are another type of ANN that is similar to FFNN [88]. In addition to fully connected layers, CNNs also include convolutional layers as the name suggests. These convolutional layers have an ability to process data as multidimensional arrays which enables the architecture to be very effective in image recognition and classification [88, 89]. CNNs have also proved to be successful in different fields [90, 91] including biological sequence analysis [92, 93]. Considering the algorithm's powerful potential for uncovering highly complex patterns from input raw datasets, these capabilities can also be beneficial in viral metagenomics.

Hidden Markov Models

Hidden Markov Models (HMM) are a probabilistic model that can predict a sequence of unknown variables by capturing hidden information of observable events. HMMs have two stochastic processes: a visible process of observable symbols and an invisible process of hidden states [94]. Because of these properties, HMMs were successfully used for speech recognition before the Deep Learning era [94, 95]. In speech recognition, as the goal is to predict pronounced word from recorded signal, an HMM model would try to discover a sequence of phonemes (sounds of language) which made the actual pronounced sound. Given the example, phonemes are states whereas the uttered word is the observation. According to the model, states can never be directly observed but rather deduced from the observation. This approach can also be useful to model protein or DNA sequences which usually contain smaller substructures often displaying different functions and different statistical properties. HMM can thus offer effective capabilities of prediction and patter recognition for modelling biological sequences including gene modelling [96], base-calling. [97] as well as for modelling for DNA sequencing errors. [97].

2 PRESENT INVESTIGATIONS

2.1 General aim

The primary aim of this thesis was to develop powerful distributed computing and machine learning methodologies and apply them to viral metagenomics, in order to enable large-scale analysis and identification of highly divergent viruses in human metagenomic sequencing data.

2.2 Specific aims

Paper I: To re-analyze all the assembled contiguous sequences (contigs) previously classified as "unknown" by NCBI BLAST, using an algorithm based on Hidden Markov models.

Paper II: To investigate whether training machine learning algorithms such as Random Forest and Artificial Neural Networks based on Relative Synonymous Codon Usage (RSCU) could improve discovery of viral sequences in human metagenomic sequences.

Paper III: To develop a deep learning tool that can identify viral genomes in raw metagenomic sequences from different human samples.

Paper IV: To develop a scalable pipeline on top of Hadoop and Spark which could integrate standard bioinformatics algorithms and would be able to process hundreds of NGS samples in a reasonable time.

Paper V: To estimate what proportion of head and neck cancers may be preventable by HPV vaccination, using all RNA sequencing dataset from head and neck squamous cell carcinomas (HNSCC) from The Cancer Genome Atlas (TCGA) and our scalable pipeline ViraPipe.

3 MATERIALS AND METHODS

3.1 Patient samples and Sequence data

In **Paper I, II, III, IV** we used datasets from metagenomic sequencing projects previously designed to investigate viral sequences in human samples from different patient groups. Total nucleic acids were extracted for the most samples except of formalin-fixed paraffin-embedded (FFPE) biopsies in which case only DNA was extracted. Sample types contained serum, fresh frozen biopsies, FFPE from skin samples and from condyloma and swabs. Illumina machines such as MiSeq, NextSeq, and HiSeq were used to sequence these samples (Table 2).

Table 2. Metadata of human metagenomic projects used in this thesis.

Project ID	Total number of raw reads	Average length of reads	Sample type	Sequencing Platform
2011_G5	585,521,156	250	Serum	MiSeq
2011_2	765,078,022	101	Skin (both fresh frozen and FFPE)	MiSeq
2014_B	463,686,630	150	Prostate secretion	NextSeq
2014_F1	31,784,562	250	Paraffin blank block	MiSeq
2014_G1	46,601,934	250	Serum	MiSeq
2014_G5	1,034,289,514	101	Serum	NextSeq
2014_G6	481,759,557	150	Serum	NextSeq
2014_G7	451,344,599	150	Serum	NextSeq
2014_7	51,719,623	150	Skin (FFPE)	NextSeq
2014_E1	236,299,783	150	Cervical tissue (FFPE)	NextSeq
2015_1	494,183,607	150	Biopsy	NextSeq
2015_F	207,891,764	150	Cervix tissue (FFPE)	NextSeq
2015_F2	336,156,550	150	Cervix tissue (FFPE)	NextSeq
2015_4	415,626,171	150	Serum	NextSeq
2014_A1	22,061,444	250	Cervix tissue (FFPE)	MiSeq
2015_5_LH	37,378,706	180	Saliva	MiSeq
2014_9	1,034,289,514	100	Serum	HiSeq
2014_14	376,961,716	150	Skin swabs	NextSeq
2014_15_SR	29,693,475	150	Serum	MiSeq
2013_1	48,243,885	150	Skin (fresh frozen tissue)	MiSeq
2013_2	34,034,998	250	Skin (fresh frozen tissue)	MiSeq
2014_10	78,080,812	250	Skin (FFPE)	MiSeq
2014_D3	56,224,598	250	Spleen and pancreas tissue	MiSeq
2014_Q1	502,617,745	150	Laryngeal, tonsillar & cervical tissues	NextSeq

In **Paper II**, we also used dataset from the Codon Usage Database (https://www.kazusa.or.jp/codon/) containing complete genes from the NCBI GenBank.

In **Paper V**, we used the dataset obtained from The Cancer Genome Atlas (TCGA) portal where more than 2.5 petabytes of genomic, transcriptomic, epigenomic and proteomic data are publicly available (http://cancergenome.nih.gov/). We obtained all RNA sequencing (RNA-seq) datasets from all primary tumor samples from patients belonging to the Head and Neck Squamous Cell Carcinoma project (TCGA-HNSC). The total amount of downloaded patient files included 500 bam files (each bam file corresponding to each patient) where the number of male patients were 367, while females were 133. In total, files consisted of 3.7 terabytes of data.

3.2 Methodology

3.2.1 Study I

In this study, we used HMMER3 [98] to re-analyze sequences that were labeled as unknown by NCBI-BLAST. The algorithm implements profile Hidden Markov Models to detect distant relatives in sequence databases. As for the reference database, we obtained profile viral HMMs constructed from all the virally annotated proteins in RefSeq [99]. Profile HMMs capture the evolutionary changes that might have happened in a set of homologs sequences using multiple sequence alignment. HMM uses a position-specific scoring system to determine how conserved each amino acid is, and which deletions and insertions might have occurred in the genomes.

Sequences were evaluated and ordered based on E-value. In this study, sequences were classified as viral if the calculated E-value for one of their genes was less than 1e-5. In case a sequence had hit with multiple virus families, then the hit with the lowest E-value was selected.

Before assigning the assembled contigs into different taxonomic groups, the reads were quality filtered based on Phred quality scores. Quality checked reads were then mapped against human, bacterial, plant and phage genomes and highly similar reads were removed from the analysis. Then, the reads were normalized and assembled by using de novo assembly algorithms.

In order to evaluate how accurately HMM-based pipeline can classify sequences, we obtained simNGS and simLibrary (http://www.ebi.ac.uk/goldman-srv/simNGS/) software to generate simulated NGS reads. The tools were used with default parameters. In the simulation we included human, bacterial, plant and viral genomes. The simulated reads were then subject to the same viral pipeline as the main NGS dataset.

3.2.2 Study II

In this work, we used Random Forest and Feed-Forward neural networks to build binary virus/non-virus classifiers. We built two machine learning models during this study: GenBank model based on genes extracted from GenBank and metagenomic model based on sequences obtained from different metagenomic experiments. For the metagenomic model, sequences were labeled by two algorithms: firstly, we used PCJ-BLAST[100] with the most recent nt database and secondly, we applied HMMER3 for the sequences that were classified as being unknown origin by PCJ-BLAST. Results from both algorithms were combined for machine learning purposes.

3.2.2.1 Relative Synonymous Codon Usage

To extract features for the machine learning algorithms we counted Relative Synonymous Codon Usage frequency (RSCU) from the dataset [101]. Synonymous codons encoding the same amino acids are identical at the protein level but they are not used randomly. Different organisms choose synonymous codons selectively. Therefore, we expected that extracting RSCU values from different genomes for machine learning purposes would output effective classification results. For a given assembled contig, RSCU value for each codon was calculated as indicated in the following formula:

$$f_{ij} = \frac{x_{ij}}{\frac{1}{n_i}\sum_{j=1}^{n_i} x_{ij}}$$

Where x_{ij} represents the number of occurrences of the *j*th codon coding for *i*th amino acid, which is encoded by total number of n_i synonymous codons. In other words, this formula divides the observed number of occurrences by the expected if the usage of the codons was uniformly distributed [101].

3.2.2.2 Training the models

To train the model based on the metagenomic sequences and to provide as accurate estimation of the performance as possible we applied *leave-one-experiment-out cross-validation* (LOEO) approach. With this methodology, the machine learning algorithms were trained on 18 metagenomic projects and tested on the remaining 19. This process was repeated 19 times so that each time a different metagenomic project was used as the test set. Using this approach, the algorithms were tested on strictly unseen dataset.

In metagenomic data where viral sequences usually contain less than 1%, a classification model which always predicts a sequence as non-virus will get 99% accuracy but such a classifier would, of course, be useless. With the huge class imbalance between virus and non-virus data points within the datasets, we needed

to measure *precision* (fraction of predicted positives which was actually correct) and *recall* (fraction of actual positives which was predicted correctly) for virus class separately as the main goal of the study was to separate and identify viral genomes from other sequences. When calculating *precision* and *recall*, however, the classification threshold is typically set at 0.5 which means that if p(virus) > 0.5, the model would label a sequence as a virus. If the purpose of the classification was to obtain as few false positives as possible, we could increase the classification threshold in which case we would get higher *precision* but lower *recall*. Conversely, lowering the threshold would result in lower *precision* but higher *recall*.

The area under ROC (AUROC) curve is constructed by visualizing true positive rate against false positive rate at different thresholds. Considering that the AUROC curve can summarize the model's performance at all possible thresholds, we used the curve as the main metric to evaluate the machine learning models designed in this study as well as in Study III.

Afterwards, we used micro and macro-average evaluation measures [102] to combine the results from different testing sets. In micro-averaged measure, a testing set that provides more validation samples has a bigger impact on the results whereas in macro-averaged measure, the number of samples is ignored, and the results are simply averaged.

3.2.3 Study III

Here, we used Convolutional Neural Networks to build a deep learning tool, ViraMiner for detecting viral genomes among raw metagenomic sequences from human samples. The architecture of ViraMiner has two convolutional branches in order to capture different types of information from sequences (Figure 4).

To design the input training dataset for CNN, each contig was split into equal 300 bp fragments which were labeled as the initial contig. After the split, the remaining nucleotides at the end of the contigs were removed and not included in the training set. For instance, 920 bp contig would be divided into three 300 bp sequences and the remaining 20 nucleotides would be removed from the analysis. Moreover, sequences that included at least one N letter (unknown) were not included in the training set. We also designed an input dataset with longer, 500bp sequences with the same approach but the initial analysis indicated that the 300bp dataset could achieve considerably better accuracy. Consequently, we decided to continue the study with only 300 bp sequences.

The model receives raw sequences in a one-hot encoded form and processes them with two different convolutional branches. The frequency branch is followed by a global average pooling that outputs average values after each convolutional filter.

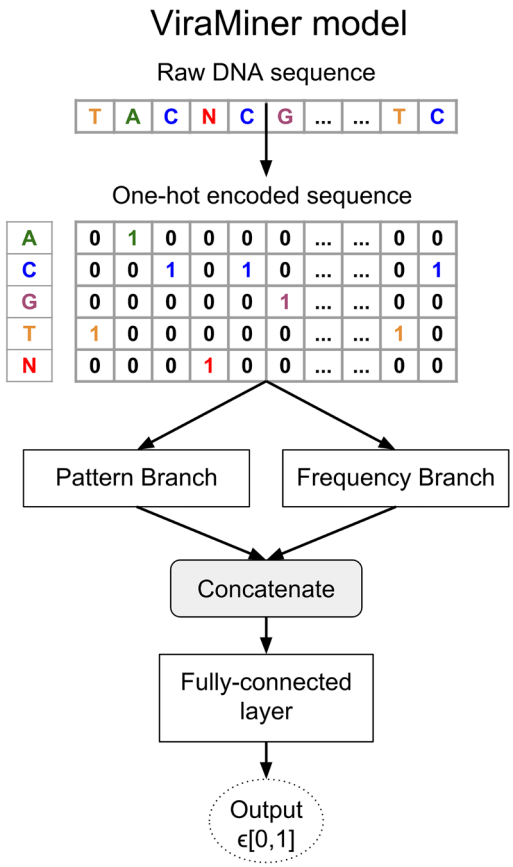

Figure 4. The architecture of the ViraMiner model. *The model has two convolutional branches that receive raw sequences with 300 bp length in a one-hot encoded form. Both of these branches output 1D vector which are then concatenated and all-to-one connected to the output node.*

The pattern branch is followed by a global max pooling that yields max values. The latter was designed based on DeepVirFinder architecture [78]. The model then concatenates outputs of these branches (1D vectors) and connects all-to-one to the output node which gives final value in the range of [0,1] through the sigmoid activation function. The pre-trained model is available here: https://github.com/NIASC/ViraMiner

To train the model, dataset from the metagenomic sequencing projects were shuffled and split into training, validation and testing sets. The hyperparameter scan was performed for the following parameters:

- Filter size
- Learning rate
- Layer size (implies number of filters in convolutional layer as well as number of nodes in the fully connected layer)
- Dropout probability

This hyperparameter search was performed for the Pattern and Frequency branch separately. The performance of these models was then evaluated with area under ROC curve.

For comparison, *k*-mer values were also extracted from the raw metagenomic sequences to train Random Forest. As *k* increases, however, counting *k*-mers becomes more and more computationally expensive. To overcome this obstacle, we designed an algorithm based on Spark, which could conduct counting much faster and much more flexible way.

3.2.4 Study IV

To speed up the analysis of NGS data, we developed a scalable parallel pipeline, ViraPipe, implemented on Hadoop and Apache Spark.

However, implementing a parallel workflow for viral metagenomics implies several significant challenges. Firstly, many existing bioinformatics tools and algorithms such as BWA, BLAST, HMMER3, and de novo assemblies, do not support parallel computations. Secondly, the existing genomic file formats are not compatible with distributed file systems, especially the binary formats such as BAM, which are not distributable without using external tools. In addition, when distributing executions over the entire cluster; the pipeline needs to have repeated access to reference databases, which are essential for alignment-based algorithms.

To conduct experiments and run algorithms we used a Hadoop cluster administered by the department of Laboratory Medicine at Karolinska Institutet. The infrastructure included 24 computing machines with 256 GB of RAM and 56 cores in each. In total, it provides approximately 6 TB of RAM and 1288 CPU cores. One machine was dedicated to the cluster management while 23 machines were deployed as Spark workers.

As metagenomic sequencing datasets can be processed in partitions (short read partitions), many current bioinformatics tools could be directly integrated into the Spark framework without re-implementing Spark versions of them. Data locality over the nodes enabled us to run these algorithms in each node at the same time. Reference databases, however, needed to be replicated in every node in the computing cluster. As for the genomic file formats, the Hadoop-BAM library[62] provided the functionality to distribute BAM files in HDFS and process them in-memory with Spark.

Currently, the ViraPipe workflow includes several essential bioinformatics tools that were implemented or integrated into the Spark framework and can be deployed in the cluster environment. It starts with decompressing input files and distributing interleaved FASTq files into HDFS to make them ready for the BWA alignment. BWA-MEM is implemented with jbwa library (https://github.com/lindenb/jbwa) which executes the library through Java Native Interface (JNI). Read normalization is performed in-memory with Spark, according to digital normalization [47] which filters out highly redundant reads based on *k*-mer abundance. Normalized reads are then assembled with Megahit, which is run in parallel Spark tasks. Reads are assembled per sample and assembled contigs are saved into HDFS. For the BLAST searches, the reference database is replicated in every node across the cluster and contigs, produced by Megahit are repartitioned with Spark RDD. Similarly, to execute HMMER3 [98] on Spark, ViraPipe uses the same data-parallel computation strategy as for the BLAST searches. The pipeline also provides SparkSQL based interface to query generated files such as BAM and Apache parquet formats in parallel. The entire workflow of ViraPipe is shown in Figure 5. The code of the pipeline is available here: https://github.com/NIASC/ViraPipeV2

3.2.5 Study V

In this study, we used the ViraPipe algorithms to analyze 3.7 terabytes of sequencing dataset originating from head and neck squamous cell carcinoma project in TCGA portal, which included 500 patient samples. The main purpose of the study was to investigate what fraction of these tumors can be preventable by HPV vaccination. Hence, we calculated what percentage of these samples contained HPV oncogenic E6/E7 transcripts. According to the analysis, a patient was classified as HPV positive if the corresponding sample contained at least two reads from both E6 and E7 transcription.

The sequencing dataset was initially processed by TCGA portal where the reads were aligned against human and 10 types of human viral genomes including HPV. After downloading and storing the BAM files into HDFS, we processed the files with the Hadoop-BAM library for the parallel read filtering. More specifically, the BAM files were converted into SparkSQL tables from which human reads were filtered out using simple SQL querying. The result FASTQ files were stored back to HDFS.

The code for quality checking of the reads was adopted from the source code of Trimmomatic [51] and integrated into ViraPipe to filter out low quality reads from partitioned FASTQ files. Afterwards, the reads were subject to data normalization and the normalized reads were then aligned against the most recent HPV database.

As the tumor samples in TCGA originate from USA, we compared our findings to the incidence rate of HNSCCs in the USA according to Cancer incidence in Five Continents (CI5). CI5 databases provide detailed statistics about the incidence of cancer around the world recorded by cancer registries.

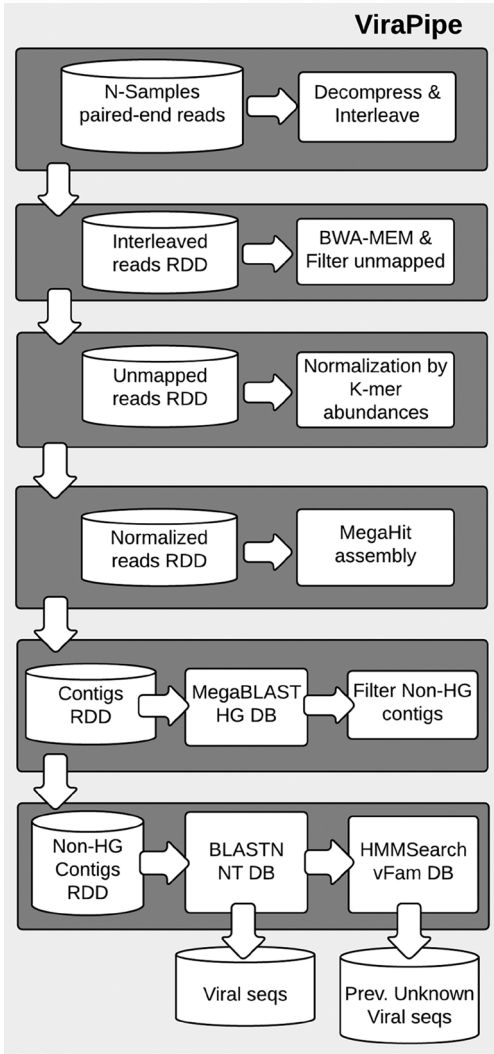

Figure 5. Workflow of ViraPipe.

4 RESULTS AND DISCUSSIONS

4.1 Putative novel viruses

In Paper I, by using HMMER3 algorithm with the profile *vFams* database, we identified 510 potential viral sequences that were missed by BLAST. Among these contigs, some were related to small single-stranded DNA viruses such as Anelloviridae and Circoviridae whereas many of them were identified to have similarities with larger double-stranded DNA viral families such as Mimiviridae, Phycodnaviridae, and Herpesviridae.

These contigs were then compared to the *Pfam* database to detect conserved proteins. The contigs that were identified as distant relatives of small circular viral families contained sequences that are similar to genes that encode the viral hallmark genes such as Rolling-circle replication initiation endonuclease, helicases and SpoIIIE/FtsK motifs. The contigs that were related to large viral families included genes that are present in both viruses and eukaryotes.

The length of these viral-related contigs varied from 500 to 100 000 bp (mean = 3362). For comparison, sequences that remained "unknown" after applying both BLAST and HMM-based pipeline had a much lower average length (mean = 365). This suggested that viral the HMM pipeline is particularly effective in classifying contigs whose length is relatively long.

The evaluation of the HMM pipeline with the simulation tools of NGS data showed that the algorithm had particularly high accuracy (~99%) when identifying viral genomes belonging to ssDNA viral families such as Anelloviridae, Circoviridae and Parvoviridae as well as sequences from several dsDNA viral families including Papillomaviridae and Polyomaviridae. However, the pipeline had very poor performance when classifying Mimiviridae (~3%) in which case this viral family was confused mostly with plant genomes. This suggested that before applying HMM-based pipeline, it is essential to discard all cellular genomes in the initial step of the analysis.

In general, the analysis showed that BLAST should not be replaced by HMM-based pipeline but rather be used as the second stage algorithm after BLAST. Considering that the *vFams* database is constructed based on all viral proteins from GenBank, the methodology is likely to become even more effective as the database grows with novel viruses.

4.2 Machine learning models

4.2.1 Random Forest

In paper II, we designed two models. The first model was trained based on sequences originating from GenBank whereas the second model was trained on sequences coming from metagenomic experiments. In both cases, we extracted relative synonymous codon usage (RSCU) values to design training datasets.

While the GenBank model achieved a superb performance when testing it on sequences coming from the same source (area under ROC = 0.99), it failed to generalize on assembled contigs originating from metagenomic sequences and performed very close to a random classifier (area under ROC = 0.51). As the main purpose of the study was to create a model that could classify metagenomic sequences, we trained the second model based on 19 metagenomic projects.

Combining the results from the model performances on different testing sets (*leave-one-experiment-out cross validation*) showed area under the micro- and micro-averaged ROC curve 0.789 and 0.785 respectively (Figure 6). This demonstrated that the model performed far better than the GenBank model or a random classifier. The model could also obtain 75% *precision* at 8% *recall* or 95% *precision* with 3.7% *recall*.

To confirm the results or possibly improve the accuracy, we also trained a Feed-Forward Neural Networks on the same RSCU dataset. FFNN model gave almost the same performance with area under the micro-averaged ROC curve 0.790.

To extract codons that contributed most to the RF classification model, we applied the RF feature important analysis. RSCU values for six codons (TCG, CGC, CGA, GCG, GTA, and CCG) appeared to have the most decisive roles in the RF model, out of which TCG and CGC were the top two (Figure 7). In the training dataset, these six codons had very low RSCU values among non-viral contigs whereas they had higher values in viral contigs. By comparison, these codons are not commonly found in the human genome either[103]. This suggested that the algorithm chose codons for its classification model that are not frequently found in non-viral contigs.

Overall, we found that designing machine learning algorithms based on RSCU values can deliver important information in addition to alignment-based tools for taxonomy classification of metagenomic sequencing data. In fact, the proposed machine learning methods can be used to further search and sort unknown sequences where potential viral genomes might be hidden.

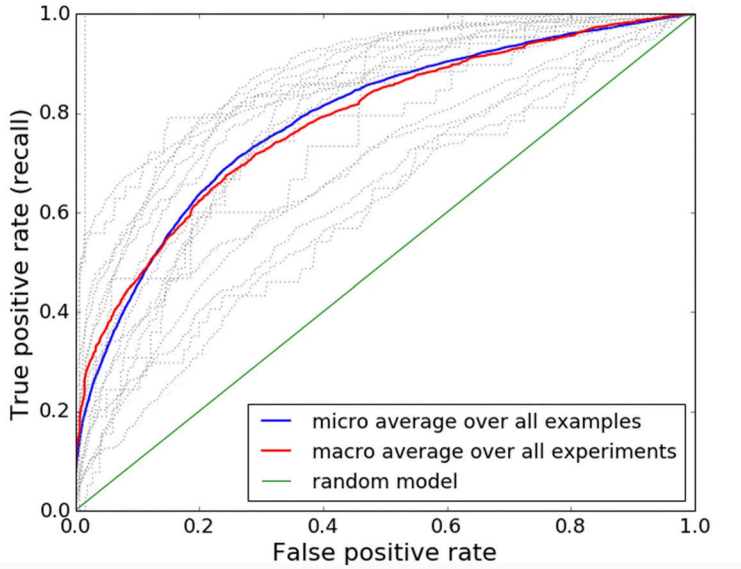

Figure 6. Performance of the metagenomic model for each LOEO fold. *The red line represents macro-averaged ROC curve and the blue line shows the micro-averaged ROC curve across the experiments. The grey lines depict the model's performance for each leave-out experiment.*

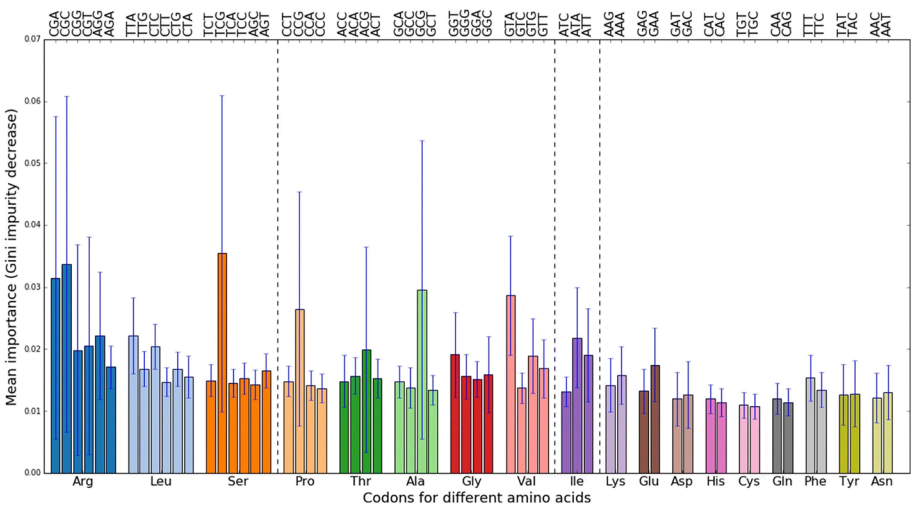

Figure 7. The feature importance in the metagenomic model. *six codons TCG, CGC, CGA, GCG, GTA, and CCG had more decisive roles in the RF classification model compared to other codons.*

4.2.2 ViraMiner

In Paper III, we applied Convolutional Neural Networks on raw metagenomic datasets from human samples to improve the accuracy shown in Paper II. The training dataset included 300 bp length contigs and was divided into training, validation and testing sets. The proposed architecture with two convolutional branches called ViraMiner achieved the area under the ROC curve of 0.923 (Figure 8). With ViraMiner, we could obtain 90% *precision* with 32% *recall* or 95% *precision* with 24% *recall*.

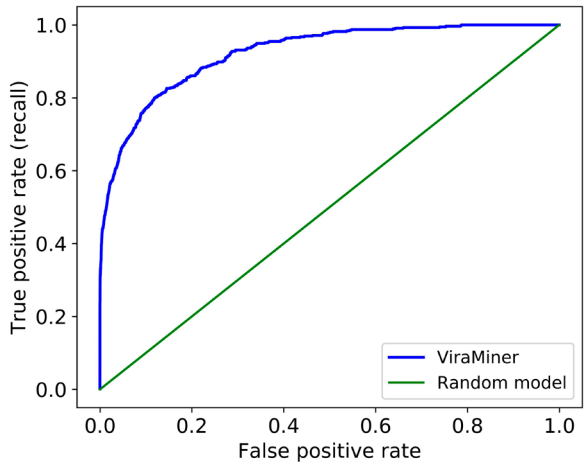

Figure 8. Performance of ViraMiner model on the test set. *The blue line represents the ROC curve of ViraMiner (0.923) whereas the green line shows how a random classifier would perform on the same dataset.*

For comparison, we also extracted *k*-mers (3- 4- 5- 6- and 7-mers) from the same partitioned dataset as above and trained Random Forest. While the best RF performance was reached on 6-mers with AUROC value 0.875, RF produced lower but very similar accuracy on 3- 4- 5- and 7-mers with AUROC scores as follows respectively: 0.867, 0.872, 0.873, and 0.869

Considering that ViraMiner includes two convolutional branches in its architecture and each branch was trained separately for the study, we also tested how they performed independently on the same dataset. The results showed that the Pattern branch with the max pooling operator yielded a test AUROC of 0.905 whereas the Frequency branch with the average pooling operator achieved AUROC of 0.917. Even though each branch produced very high accuracies on the test set, ViraMiner achieved even higher AUROC value (0.923) with the combination of the two (Figure 9).

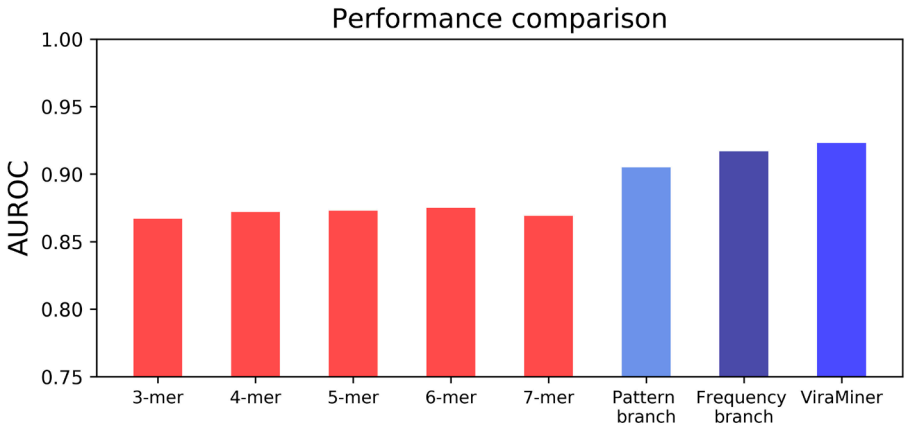

Figure 9. Comparison of different models trained on human metagenomic sequencing data. *Red bars represent performances of RF on extracted k-mer values whereas blue bars show CNN models trained on raw sequences. Frequency and Pattern branches produced AUROC values above 0.9 but the combination of the two achieves the best performance (AUROC of 0.923) out of all models.*

Although ViraMiner achieved impressive classification performance on the test set, the main goal of the study was to design a deep learning model that could generalize its classification capabilities on totally new and unseen metagenomic projects from which no data point was included in the training set. For that purpose, we tested the proposed architecture on specific metagenomic projects originating from one specific sample type. In the dataset, we had five, the largest number of metagenomic projects sequenced from serum samples and we investigated how ViraMiner would perform on thesse five metagenomic datasets if they were not included in the training set. The architecture was retrained five times but each time, sequences from each dataset were left out to use them as the test set. ViraMiner performed somewhat differently for each dataset but combining the results with the micro-average measure showed AUROC of 0.94 (Table 3).

Table 3. ViraMiner accuracy on unseen metagenomic experiments originating from serum samples.

Left-out experiment	2011_G5	2014_G1	2014_G5	2016_G6	2014_G7	Micro-average
Test AUROC	0.95	0.89	0.96	0.92	0.86	0.94

Considering the results described above, using ViraMiner as a recommendation system can be useful to further investigate unknown sequences. Even though it is obvious that ViraMiner cannot replace the alignment-based methods at this moment because of the limited training data, it can provide important prediction

capabilities. Unlike BLAST and HMMER3, ViraMiner does not use a reference database for sequence classification which implies that the architecture extracts different kinds of features of genome composition compared to the conventional methods. We thus recommend the model for exploring and inspecting unknown sequences for highly-divergent viruses which in turn may help us understand more about the biodiversity of viral species in human samples.

4.3 Data-parallel processing of NGS data

4.3.1 ViraPipe

In Paper IV, we developed a data-parallel computing workflow called ViraPipe to speed up the analysis of metagenomic sequencing data. The workflow is implemented on top of Hadoop and Spark and includes several standard bioinformatics algorithms for processing raw NGS reads. As the main goal of the study was to create a scalable pipeline, we compared the performance of ViraPipe with the performance of the existing sequential pipeline and we also tested how ViraPipe performed with different sizes of input datasets.

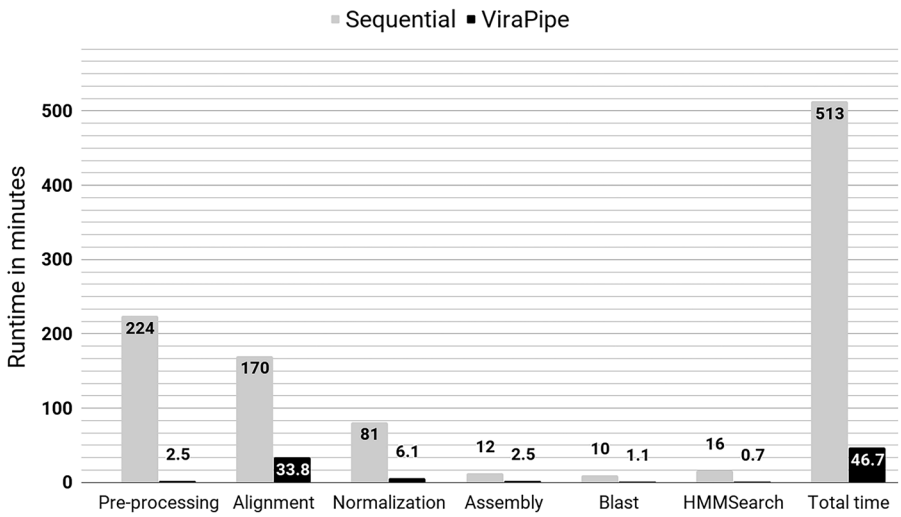

Figure 10. Performance comparison between ViraPipe and sequential pipeline in terms of processing time. *The input dataset contained 105.5 GB sequencing data originating from 13 human samples***.**

Figure 10 shows that ViraPipe managed to process the input dataset (105.5 GB) approximately 11 times faster than the sequential pipeline. The analysis of scalability regarding larger input datasets revealed that the overall runtime of ViraPipe increased linearly in proportion to dataset size and thus, showed good scalable performance (Figure 11).

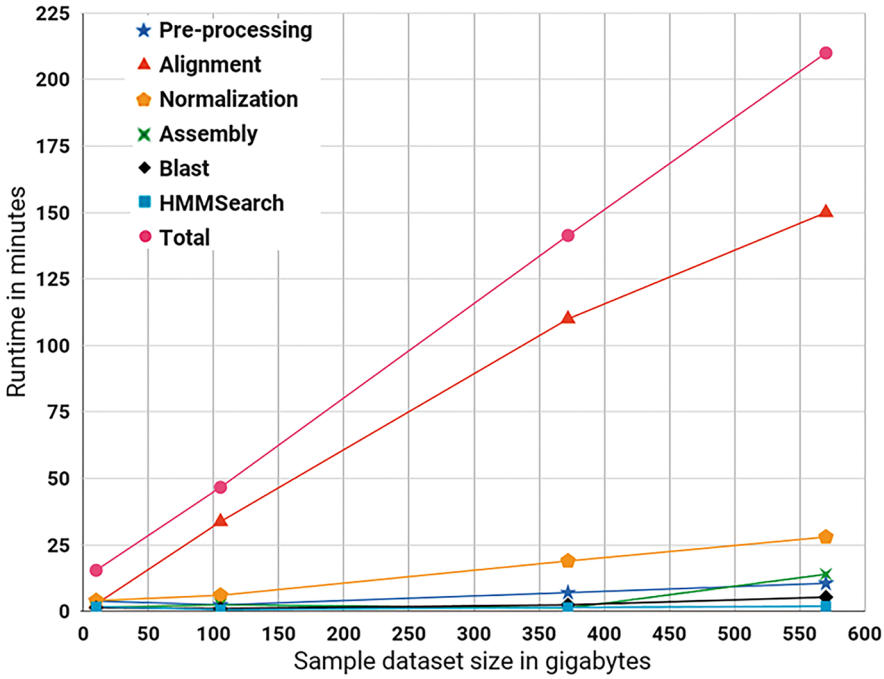

Figure 11. *Scalability of ViraPipe.*

Overall, the experiments showed that ViraPipe was able to process hundreds of gigabytes of sequencing data in a reasonable time. Dividing data into smaller partitions enabled algorithms to harness all necessary computer power from the cluster. In general, this study demonstrated that Apache Hadoop and Spark brings great improvements and flexibilities to bioinformatic pipelines.

4.3.2 HPV oncogenic transcriptions in head and neck squamous cell carcinoma patients

In Paper V, we analyzed 500 patient samples belonging to Head and Neck Squamous Cell Carcinoma project in the TCGA portal. The primary aim of the study was to estimate what proportion of the cancers are caused by HPV infection and can be preventable by vaccination.

Therefore, we investigated what percentage of samples included HPV E6 and E7 oncogenic transcripts.

As the dataset contained 3.7 TB sequencing reads, for the analysis we used ViraPipe which was developed for data-parallel processing of massive NGS data.

According to the results, HPV reads were identified in 114/500 individuals. Only 92 had transcriptions of E6 and E7 regions. Separating the numbers according to gender showed that the HPV transcriptions were found in 11/133 females and 81/367 males. HPV types (16/18/33) that are targeted by vaccination were detected in 87/500 patients. Transcripts of other HPV types such as 31/45/52/58 that are also targeted by vaccination were not found in these specimens.

These numbers were then compared to the incidence rate of HNSCCs in the USA as reported in Cancer Incidence in five Continents (CI5). Given the fact that the number of tumor samples that were sequenced in TCGA portal did not precisely reflect how common these cancers are in the USA, we estimated the proportion of tumors containing transcription of HPV viral oncogenes for each tumor form separately (Table 4).

The comparison indicated that vaccine-preventable HPV viral oncogene transcriptional activity might be found in about 2.4% female and 21.1% male HNSCC patients in the United States. This suggested that if these cancers are truly caused by HPV, vaccination might prevent 36 000 head and neck cancers in the United States.

Table 4. Transcription of HPV viral oncogenes in the head and neck cancers of the TCGA.

ICD-10-CM	Cancer site	HPV 16	HPV 18	HPV 33	HPV 35	HPV Neg	HPV Pos	Total	HPV 16/18/33		HPV 16/18/33 (%)		CI5		Preventable no. of cases	
									Female	Male	Female	Male	Female	Male	Female	Male
C00	Lip	0	0	0	0	3	0	3	0	0	0%	0%	2 368	6 836	0	0
C01	Base of tongue	8*	1*	2*	1	10	10	20	0	9	0,7%	11%	16 283	40 060	110.77	4 632.79
C02	Other unspecified parts of tongue	6	0	3	1	117	10	127	1	8						
C03	Gum	4	0	0	0	7	4	11	0	4	3,7%	9%	14 108	20 455	517.72	1 876.61
C04	Floor of mouth	3	0	1	1	48	5	53	0	4						
C05	Palate	3	1	0	0	1	4	5	3	1						
C06	Other unspecified parts of mouth	1	1	0	0	38	2	40	1	1						
C09	Tonsil	29	1	3	1	6	34	40	4	29	10%	72,5%	6273	28 000	627.3	20 300
C10	Oropharynx	4	0	0	0	6	4	10	0	4	0%	40%	2000	6454	0	2581.6
C13	Hypopharynx	2	0	0	1	6	3	9	0	2	0%	22,2%	2185	8965	0	1992.22
C14	Other ill-defined sites in lip,oral cavity, pharynx	5	1	3	0	61	9	70	1	8	1,4%	11,4%	1142	3502	16.31	400.23
C32	Larynx	4	1	2	0	104	7	111	1	6	0,9%	5,4%	12 866	49 048	115.91	2651.24
C41	Bones, joints and articular cartilage of other and unspecified sites	0	0	0	0	1	0	1	0	0	0%	0%	NA[1]	NA[1]	0	0
	Total (Female/Male)	69 (8/61)	6 (2/4)	14 (1/13)	5 (0/5)	408 (122/286)	92 (11/81)	500 (133/367)	11	76			57 225	163 320	1388.01	34 434 .69

The first column displays the codes from the International classification of diseases (ICD, 10th revision, clinical modification) whereas the second to last column represents the yearly numbers of female and male cases for each cancer form in the USA according to CI5 statistics. The last column multiplies the annual number of cases in the USA by the fraction of tumors that had HPV viral oncogene transcription of vaccine-targeted HPV types detected in this study. * indicates samples with an HPV infection with more than one HPV type.
1) NA= Not available, CI5 does not provide the separate codes for different bone cancers and the majority of bone cancers are not considered as head and neck cancers.

5 CONCLUDING REMARKS

I. Viruses in human samples seem to be more diverse compared to what is suggested by using conventional alignment-based methods for taxonomy classification. By using an algorithm based on HMM we identified several hundreds of potential viral sequences that were missed by BLAST. As the reference database is constructed by multiple sequence alignment from all viral proteins from Genbank, it is highly likely that this method will become even more effective when the database contains more novel viruses.

II. Machine learning methods based on codon usage bias can provide supplementary information for the identification of highly-divergent viruses. This method can sort and prioritize unknown sequences for further examination.

III. Convolutional Neural Networks based on raw metagenomic sequences showed considerably improved accuracy for the detection of viral sequences compared to other machine learning models. ViraMiner is the first deep learning tool which can detect the presence of viral genomes among raw assembled contigs originating from various human biospecimens.

IV. The popular distributed computing frameworks Apache Hadoop and Apache Spark were used to create a scalable parallel bioinformatics pipeline for large-scale analyses of metagenomic studies. The results showed that ViraPipe, the new distributed workflow is able to process sequencing data from thousands of human samples in a reasonable time.

V. By using ViraPipe, we analyzed metagenomic sequencing data originating from hundreds of tumor samples of head and neck squamous cell carcinoma. Our findings revealed that vaccine-preventable HPV16/18/33 oncogenic transcripts were present in 17% of such patients. Comparing the results to the cancer incidence rate in the United States implied that roughly 36 000 annual head and neck cancer might be preventable in the USA.

6 FUTURE PERSPECTIVES

Identification of highly-divergent viruses from human samples remains a major challenge in the field. Studies included in this thesis provided further evidence that machine learning is a very powerful tool to uncover complex features of genome compositions that can be helpful for detecting viral sequences. We have successfully applied several machine learning concepts on the human metagenomic sequencing datasets, but it should be noted that the supervised learning algorithms which we employed to design these models usually work best when the number of data points in each class is approximately equal. Because of the fact that in viral metagenomics, viral data points typically include less than 1% of the training data, it is a significant challenge to train a model to separate the minority class with high accuracy. Resampling techniques such as oversample minority class or undersample majority class are not very effective either because of the massive scale of data imbalance.

To further improve the achieved accuracy of the models, a semi-supervised approach based on Generative Adversarial Networks (GANs) can be useful [104]. GANs have two neural networks: *generator* and *discriminator* which are trained simultaneously. *Generator* generates synthetic new data points from random noise that look real whereas *discriminator* estimates them for authenticity whether they came from the actual training dataset or not. Both *generator* and *discriminator* improve their ability through the adversarial training framework. This algorithmic architecture shows very promising results for anomaly detection tasks where the primary goal is to identify very rare and unusual objects compared to what is considered to be normal [105, 106]. Given the fact that viruses contain a very small portion of all genomes in human samples, viral sequences can also be considered as anomalies. During the training phase, *discriminator* can learn the distribution of anomaly-free data containing non-viral genomes including human, bacteria and plant genomes whereas *generator* can learn how to mimic the available dataset. If the training of these two adversarial networks is successful, *discriminator* will be able to detect a sequence that does not belong to the anomaly-free dataset by using an anomaly score. Consequently, we could save this model and apply it to future NGS projects to identify and distinguish viral genomes from the other organisms.

Another way to increase the accuracy of machine learning algorithms for viral metagenomics is to improve de-novo assemblers to produce better quality contigs. Nowadays, as these algorithms are in their infancy, they are susceptible to several errors such as substitutions, insertions or deletions while reconstructing full genomes. These errors are likely to have a big impact on the quality of contigs and consequently, their taxonomy assignment. Moreover, these algorithms are required to assemble millions of short reads into longer contigs, which is one of the most computationally expensive steps in the sequence analysis workflow. To design scalable,

efficient and more accurate assemblers Spark framework and Graphx API can be used. The Graphx library offers parallel construction and computing graphs that can be employed to parallelize de-Bruijn graph based de novo assemblers and improve their abilities to generate better quality contigs. However, there are several significant challenges that need to be tackled while constructing the distributed de-Bruijn graphs including erroneous reads, chimerical connections between the nodes as well as redundancy in graphs. Addressing these challenges and further improvements of de-novo assemblers will most probably result in a better understanding of the human virome and its impact on human health.

7 ACKNOWLEDGEMENTS

I would like to express my gratitude to everyone who supported me as a PhD student and inspired me to accomplish all the work presented in this thesis. I am particularly grateful to:

My main supervisor **Joakim Dillner**, for letting me join your group and teaching me how to think more critically and work more effectively. I learned so many things from you which helped me to develop as a person and as a scientist. Thank you for being an excellent supervisor.

My co-supervisors: **Karin Sundström** for your guidance and support whenever I needed most. During this time, there have been several critical moments when your advice and support were crucial for me to choose the right path. **Piotr Bała** for introducing me to high performance computing clusters and supercomputers. With your help, I could analyze terabytes of data in the HPC cluster which later resulted in a couple of published papers.

My bioinformatics team: **Roxana Merino Martinez** and **Suyesh Amatya** for being always very positive and helpful especially when we had very challenging projects. During these several years, I learned many things from you and it was a great experience to be a member of the team.

Sara Arroyo Muhr, **Mehran Ghaderi and Emilie Hultin** to have the patience to answer all my questions about viruses, metagenomic samples and the lab work. Thank you for giving me the necessary knowledge to move forward. Without your help, it would have been impossible to understand anything about the field.

My fellow PhD students in the group: **Maria Hortlund** and **Hanna Kann** for sharing your experiences and always being there for me to have long necessary conversations.

Helena Andersson and **Mia Bjerke** for always being kind and helpful in the administrative issues.

Sara Nordqvist Kleppe, Miriam Elfström, **Camilla Lagheden**, **Carina Eklund**, **Sadaf Sakina Hassan**, **Ulla Rudsander** and all my colleagues from the Dillner group for creating a very friendly atmosphere in the group.

My machine learning team from University of Tartu: **Raul Vicente** and **Ardi Tampuu** for hosting me several times in Tartu to work together on very interesting projects. It was a great experience to work with you and considering the results, it is obvious that we make a dream team.

My amazing friends whom I met during my doctoral studies: **Joman Javadi**, **Dhanu Gupta**, **Magali Merrien**, **Antje Zickler**, **André Görgens**, **Rim Javad**, **Oscar Wiklander**, **Anja Reithmeier**, **Maria Hortlund**, **Laia Mira Pascual**, **Suchita Desai**, **Christina Patlaka**, **Ashish Kumar Singh**, **Agata Wasik**, **Anh Tran**, **Sergo Smedli**, **Mikheil kapanadze**, **Max Jaehnke**, **Martina Sara Ros**, **Fredrik Brusdal**. There is no way I could accomplish all this work without the joy and fun I had while hanging out with you.

Rezi Bzhalava, **Shalva Lochoshvili**, **Ani Tsverava**, **Vano Tsverava**, **Nini Khazalia**, **Gio Gurashvili**, **Emre Tetik**, **Lauro Reino**. I can write another PhD thesis to properly explain why I am mentioning each one of you here but without any further explanations: you know what I mean.

I would like to particularly thank my parents: **Marina** and **Revaz**, and my brothers: **Davit** and **Levan** for inspiring me in life and teaching me how to push harder than possible. As the youngest member of the family, I have been able to observe how you have kept moving forward no matter what, which made me realize that one could achieve anything if there is a will to work hard to succeed. Thank you for your endless support in achieving my goals. I also want to thank **Davit** for introducing me to big data technologies which became the foundation of this PhD.

განსაკუთრებული მადლობა მინდა გადავუხადო ჩემს მშობლებსა და ოჯახის წევრებს ყოველთვის გვერდში დგომისა და მხარდაჭერისთვის. ასევე, იმ სასარგებლო რჩევებისა და ინსპირაციისთვის რომელიც საბოლოოდ ჯამში ამ დოქტორანტურის საფუძველი გახდა.

8 REFERENCES

1. Organization, W.H., *IARC monographs on the evaluation of carcinogenic risks to humans: volume 100B-Biological Agents.* A review of human carcinogens, 2012.

2. Bouvard, V., et al., *A review of human carcinogens--Part B: biological agents.* The Lancet. Oncology, 2009. **10**(4): p. 321.

3. Chen, C.-J., et al., *Epidemiology of virus infection and human cancer*, in *Viruses and Human Cancer*. 2014, Springer. p. 11-32.

4. McLaughlin-Drubin, M.E. and K. Munger, *Viruses associated with human cancer.* Biochimica et Biophysica Acta (BBA)-Molecular Basis of Disease, 2008. **1782**(3): p. 127-150.

5. Plummer, M., et al., *Global burden of cancers attributable to infections in 2012: a synthetic analysis.* The Lancet Global Health, 2016. **4**(9): p. e609-e616.

6. Lindelöf, B., et al., *Incidence of skin cancer in 5356 patients following organ transplantation.* British Journal of Dermatology, 2000. **143**(3): p. 513-519.

7. Schulz, T.F., *Cancer and viral infections in immunocompromised individuals.* International Journal of Cancer, 2009. **125**(8): p. 1755-1763.

8. Arroyo Mühr, L.S., et al., *Viruses in cancers among the immunosuppressed.* International Journal of Cancer, 2017. **141**(12): p. 2498-2504.

9. Kinlen, L.J., *Infection and childhood leukemia.* Cancer Causes & Control, 1998: p. 237-239.

10. Mercalli, A., et al., *No evidence of enteroviruses in the intestine of patients with type 1 diabetes.* Diabetologia, 2012. **55**(9): p. 2479-2488.

11. Sundström, P., et al., *An altered immune response to Epstein-Barr virus in multiple sclerosis: a prospective study.* Neurology, 2004. **62**(12): p. 2277-2282.

12. Moore, P.S. and Y. Chang, *Why do viruses cause cancer? Highlights of the first century of human tumour virology.* Nature reviews cancer, 2010. **10**(12): p. 878.

13. Hortlund, M., et al., *Cancer risks after solid organ transplantation and after long-term dialysis.* International journal of cancer, 2017. **140**(5): p. 1091-1101.

14. Vajdic, C.M., et al., *Cutaneous melanoma is related to immune suppression in kidney transplant recipients.* Cancer Epidemiology and Prevention Biomarkers, 2009. **18**(8): p. 2297-2303.

15. Lindelöf, B., et al., *Incidence of skin cancer in 5356 patients following organ transplantation.* British Journal of Dermatology, 2000. **143**(3): p. 513-519.

16. Bashline, B., *Skin Cancer: Squamous and Basal Cell Carcinomas.* FP essentials, 2019. **481**: p. 17-22.

17. Kinlen, L., et al., *Collaborative United Kingdom-Australasian study of cancer in patients treated with immunosuppressive drugs.* Br Med J, 1979. **2**(6203): p. 1461-1466.

18. Gupta, A.K., C.J. Cardella, and H.F. Haberman, *Cutaneous Malignant Neoplasms in Patients With Renal Transplants.* Archives of Dermatology, 1986. **122**(11): p. 1288-1293.

19. Foulongne, V., et al., *Human skin microbiota: high diversity of DNA viruses identified on the human skin by high throughput sequencing.* PloS one, 2012. **7**(6).

20. Contraception, H. and H. Therapy, *IARC monographs on the evaluation of carcinogenic risks to humans.* Lyon: International Agency for Research on Cancer, 1999.

21. Arroyo Mühr, L.S., et al., *Human papillomavirus type 197 is commonly present in skin tumors.* International journal of cancer, 2015. **136**(11): p. 2546-2555.

22. Van Leeuwen, M.T., et al., *Immunosuppression and other risk factors for lip cancer after kidney transplantation.* Cancer Epidemiology and Prevention Biomarkers, 2009. **18**(2): p. 561-569.

23. Lucas, R., et al., *Environmental burden of disease series, No. 13. Solar Ultraviolet Radiation. Global burden of disease from solar ultraviolet radiation.* Public Health and the Environment, 2006.

24. Humans, I.W.G.o.t.E.o.C.R.t., et al., *Human papillomaviruses*. Vol. 90. 2007: World Health Organization.

25. De Villiers, E.-M., et al., *Classification of papillomaviruses.* Virology, 2004. **324**(1): p. 17-27.

26. Werness, B.A., A.J. Levine, and P.M. Howley, *Association of human papillomavirus types 16 and 18 E6 proteins with p53.* Science, 1990. **248**(4951): p. 76-79.

27. Mork, J., et al., *Human papillomavirus infection as a risk factor for squamous-cell carcinoma of the head and neck.* New England Journal of Medicine, 2001. **344**(15): p. 1125-1131.

28. Institute, N.C., *HPV and Cancer.* 2020, https://www.cancer.gov/about-cancer/causes-prevention/risk/infectious-agents/hpv-and-cancer.

29. Ndiaye, C., et al., *HPV DNA, E6/E7 mRNA, and p16INK4a detection in head and neck cancers: a systematic review and meta-analysis.* The Lancet Oncology, 2014. **15**(12): p. 1319-1331.

30. Finn, O.J., *The dawn of vaccines for cancer prevention.* Nature Reviews Immunology, 2018. **18**(3): p. 183.

31. Institute, N.C., *Human Papillomavirus (HPV) Vaccines.* https://www.cancer.gov/about-cancer/causes-prevention/risk/infectious-agents/hpv-vaccine-fact-sheet.

32. Whitford, K., et al., *Long-term impact of infant immunization on hepatitis B prevalence: a systematic review and meta-analysis.* Bulletin of the World Health Organization, 2018. **96**(7): p. 484.

33. Drolet, M., et al., *Population-level impact and herd effects following the introduction of human papillomavirus vaccination programmes: updated systematic review and meta-analysis.* The Lancet, 2019. **394**(10197): p. 497-509.

34. Schiller, J.T. and D.R. Lowy, *Vaccines to prevent infections by oncoviruses.* Annual review of microbiology, 2010. **64**: p. 23-41.

35. Thomas, T., J. Gilbert, and F. Meyer, *Metagenomics-a guide from sampling to data analysis.* Microbial informatics and experimentation, 2012. **2**(1): p. 3.

36. Wylie, K.M., G.M. Weinstock, and G.A. Storch, *Emerging view of the human virome.* Translational Research, 2012. **160**(4): p. 283-290.

37. Bzhalava, D., et al., *Unbiased approach for virus detection in skin lesions.* PloS one, 2013. **8**(6): p. e65953.

38. Mardis, E.R., *A decade's perspective on DNA sequencing technology.* Nature, 2011. **470**(7333): p. 198.

39. Sanger, F., S. Nicklen, and A.R. Coulson, *DNA sequencing with chain-terminating inhibitors.* Proceedings of the national academy of sciences, 1977. **74**(12): p. 5463-5467.

40. Zhang, J., et al., *The impact of next-generation sequencing on genomics.* Journal of genetics and genomics, 2011. **38**(3): p. 95-109.

41. Head, S.R., et al., *Library construction for next-generation sequencing: overviews and challenges.* Biotechniques, 2014. **56**(2): p. 61-77.

42. Ekström, J., et al., *High throughput sequencing reveals diversity of Human Papillomaviruses in cutaneous lesions.* International journal of cancer, 2011. **129**(11): p. 2643-2650.

43. Johansson, H., et al., *Metagenomic sequencing of "HPV-negative" condylomas detects novel putative HPV types.* Virology, 2013. **440**(1): p. 1-7.

44. Ewing, B., et al., *Base-calling of automated sequencer traces usingPhred. I. Accuracy assessment.* Genome research, 1998. **8**(3): p. 175-185.

45. Bzhalava, D., et al., *Phylogenetically diverse TT virus viremia among pregnant women.* Virology, 2012. **432**(2): p. 427-434.

46. Bokulich, N.A., et al., *Quality-filtering vastly improves diversity estimates from Illumina amplicon sequencing.* Nature methods, 2013. **10**(1): p. 57.

47. Brown, C.T., et al., *A reference-free algorithm for computational normalization of shotgun sequencing data.* arXiv preprint arXiv:1203.4802, 2012.

48. Khan, A.R., et al., *A comprehensive study of de novo genome assemblers: current challenges and future prospective.* Evolutionary Bioinformatics, 2018. **14**: p. 1176934318758650.

49. Vollmers, J., S. Wiegand, and A.-K. Kaster, *Comparing and evaluating metagenome assembly tools from a microbiologist's perspective-not only size matters!* PloS one, 2017. **12**(1): p. e0169662.

50. Li, Z., et al., *Comparison of the two major classes of assembly algorithms: overlap–layout–consensus and de-bruijn-graph.* Briefings in functional genomics, 2012. **11**(1): p. 25-37.

51. Bolger, A.M., M. Lohse, and B. Usadel, *Trimmomatic: a flexible trimmer for Illumina sequence data.* Bioinformatics, 2014. **30**(15): p. 2114-2120.

52. Grabherr, M.G., et al., *Trinity: reconstructing a full-length transcriptome without a genome from RNA-Seq data.* Nature biotechnology, 2011. **29**(7): p. 644.

53. Luo, R., et al., *SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler.* Gigascience, 2012. **1**(1): p. 18.

54. Peng, Y., et al., *IDBA-UD: a de novo assembler for single-cell and metagenomic sequencing data with highly uneven depth.* Bioinformatics, 2012. **28**(11): p. 1420-1428.

55. Li, D., et al., *MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph.* Bioinformatics, 2015. **31**(10): p. 1674-1676.

56. Shvachko, K., et al. *The hadoop distributed file system*. in *MSST*. 2010.

57. Dean, J. and S. Ghemawat, *MapReduce: simplified data processing on large clusters.* Communications of the ACM, 2008. **51**(1): p. 107-113.

58. Taylor, R.C. *An overview of the Hadoop/MapReduce/HBase framework and its current applications in bioinformatics*. in *BMC bioinformatics*. 2010. BioMed Central.

59. Chang, Y.-J., et al. *A de novo next generation genomic sequence assembler based on string graph and MapReduce cloud computing framework*. in *BMC genomics*. 2012. BioMed Central.

60. Decap, D., et al., *Halvade: scalable sequence analysis with MapReduce.* Bioinformatics, 2015. **31**(15): p. 2482-2488.

61. McKenna, A., et al., *The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data.* Genome research, 2010. **20**(9): p. 1297-1303.

62. Niemenmaa, M., et al., *Hadoop-BAM: directly manipulating next generation sequencing data in the cloud.* Bioinformatics, 2012. **28**(6): p. 876-877.

63. Zaharia, M., et al., *Spark: Cluster computing with working sets.* HotCloud, 2010. **10**(10-10): p. 95.

64. Guo, R., et al., *Bioinformatics applications on apache spark.* GigaScience, 2018. **7**(8): p. giy098.

65. Singh, D. and C.K. Reddy, *A survey on platforms for big data analytics.* Journal of big data, 2015. **2**(1): p. 8.

66. Massie, M., et al., *Adam: Genomics formats and processing patterns for cloud scale computing.* University of California, Berkeley Technical Report, No. UCB/EECS-2013, 2013. **207**: p. 2013.

67. Huang, L., J. Krüger, and A. Sczyrba, *Analyzing large scale genomic data on the cloud with Sparkhit.* Bioinformatics, 2017. **34**(9): p. 1457-1465.

68. Zhou, W., et al., *Metaspark: a spark-based distributed processing tool to recruit metagenomic reads to reference genomes.* Bioinformatics, 2017. **33**(7): p. 1090-1092.

69. de Castro, M.R., et al., *SparkBLAST: scalable BLAST processing using in-memory operations.* BMC bioinformatics, 2017. **18**(1): p. 318.

70. Abuín, J.M., et al., *SparkBWA: speeding up the alignment of high-throughput DNA sequencing data.* PloS one, 2016. **11**(5): p. e0155461.

71. Abu-Doleh, A. and Ü.V. Çatalyürek. *Spaler: Spark and graphx based de novo genome assembler*. in *2015 IEEE International Conference on Big Data (Big Data)*. 2015. IEEE.

72. Soueidan, H., et al., *Finding and identifying the viral needle in the metagenomic haystack: trends and challenges.* Frontiers in microbiology, 2015. **5**: p. 739.

73. Fancello, L., D. Raoult, and C. Desnues, *Computational tools for viral metagenomics and their application in clinical research.* Virology, 2012. **434**(2): p. 162-174.

74. Nguyen, T.H., et al., *Deep learning for metagenomic data: using 2d embeddings and convolutional neural networks.* arXiv preprint arXiv:1712.00244, 2017.

75. Fiannaca, A., et al., *Deep learning models for bacteria taxonomic classification of metagenomic data.* BMC bioinformatics, 2018. **19**(7): p. 198.

76. Arango-Argoty, G., et al., *DeepARG: a deep learning approach for predicting antibiotic resistance genes from metagenomic data.* Microbiome, 2018. **6**(1): p. 23.

77. Ren, J., et al., *VirFinder: a novel k-mer based tool for identifying viral sequences from assembled metagenomic data.* Microbiome, 2017. **5**(1): p. 69.

78. Ren, J., et al., *Identifying viruses from metagenomic data by deep learning.* arXiv preprint arXiv:1806.07810, 2018.

79. Vervier, K., et al., *Large-scale machine learning for metagenomics sequence classification.* Bioinformatics, 2015. **32**(7): p. 1023-1032.

80. Alpaydin, E., *Introduction to machine learning*. 2014: MIT press.

81. Breiman, L., *Random forests.* Machine learning, 2001. **45**(1): p. 5-32.

82. Touw, W.G., et al., *Data mining in the Life Sciences with Random Forest: a walk in the park or lost in the jungle?* Briefings in bioinformatics, 2012. **14**(3): p. 315-326.

83. Bressler, R., et al., *CloudForest: a scalable and efficient random forest implementation for biological data.* PloS one, 2015. **10**(12): p. e0144820.

84. Fusaro, V.A., et al., *Prediction of high-responding peptides for targeted protein assays by mass spectrometry.* Nature biotechnology, 2009. **27**(2): p. 190.

85. Lunetta, K.L., et al., *Screening large-scale association study data: exploiting interactions using random forests.* BMC genetics, 2004. **5**(1): p. 32.

86. Díaz-Uriarte, R. and S.A. De Andres, *Gene selection and classification of microarray data using random forest.* BMC bioinformatics, 2006. **7**(1): p. 3.

87. Bishop, C.M., *Neural networks for pattern recognition*. 1995: Oxford university press.

88. LeCun, Y., et al., *Backpropagation applied to handwritten zip code recognition.* Neural computation, 1989. **1**(4): p. 541-551.

89. Krizhevsky, A., I. Sutskever, and G.E. Hinton. *Imagenet classification with deep convolutional neural networks*. in *Advances in neural information processing systems*. 2012.

90. LeCun, Y. and Y. Bengio, *Convolutional networks for images, speech, and time series.* The handbook of brain theory and neural networks, 1995. **3361**(10): p. 1995.

91. Hinton, G., et al., *Deep neural networks for acoustic modeling in speech recognition.* IEEE Signal processing magazine, 2012. **29**.

92. Kelley, D.R., J. Snoek, and J.L. Rinn, *Basset: learning the regulatory code of the accessible genome with deep convolutional neural networks.* Genome research, 2016. **26**(7): p. 990-999.

93. Angermueller, C., et al., *Deep learning for computational biology.* Molecular systems biology, 2016. **12**(7).

94. Yoon, B.-J., *Hidden Markov models and their applications in biological sequence analysis.* Current genomics, 2009. **10**(6): p. 402-415.

95. Rabiner, L.R., *A tutorial on hidden Markov models and selected applications in speech recognition.* Proceedings of the IEEE, 1989. **77**(2): p. 257-286.

96. Munch, K. and A. Krogh, *Automatic generation of gene finders for eukaryotic species.* BMC bioinformatics, 2006. **7**(1): p. 263.

97. Liang, K.-c., X. Wang, and D. Anastassiou, *Bayesian basecalling for DNA sequence analysis using hidden Markov models.* IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB), 2007. **4**(3): p. 430-440.

98. Mistry, J., et al., *Challenges in homology search: HMMER3 and convergent evolution of coiled-coil regions.* Nucleic acids research, 2013. **41**(12): p. e121-e121.

99. Skewes-Cox, P., et al., *Profile hidden Markov models for the detection of viruses within metagenomic sequence data.* PloS one, 2014. **9**(8): p. e105067.

100. Nowicki, M., D. Bzhalava, and P. Bała, *Massively parallel implementation of sequence alignment with basic local alignment search tool using parallel computing in java library.* Journal of Computational Biology, 2018. **25**(8): p. 871-881.

101. Sharp, P.M., T.M. Tuohy, and K.R. Mosurski, *Codon usage in yeast: cluster analysis clearly differentiates highly and lowly expressed genes*. Nucleic acids research, 1986. **14**(13): p. 5125-5143.

102. Van Asch, V., *Macro-and micro-averaged evaluation measures.* Tech. Rep., 2013.

103. Castro-Chavez, F., *Most used codons per amino acid and per genome in the code of man compared to other organisms according to the rotating circular genetic code.* NeuroQuantology: an interdisciplinary journal of neuroscience and quantum physics, 2011. **9**(4).

104. Goodfellow, I., et al. *Generative adversarial nets*. in *Advances in neural information processing systems*. 2014.

105. Schlegl, T., et al. *Unsupervised anomaly detection with generative adversarial networks to guide marker discovery*. in *International conference on information processing in medical imaging*. 2017. Springer.

106. Berg, A., J. Ahlberg, and M. Felsberg, *Unsupervised Learning of Anomaly Detection from Contaminated Image Data using Simultaneous Encoder Training.* arXiv preprint arXiv:1905.11034, 2019.