Computational Tools for Annotating Antibiotic Resistance in Metagenomic Data

Gustavo Alonso Arango Argoty

Dissertation submitted to the faculty
of the Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy in

Computer Science and Applications

Liqing Zhang, Chair,
Amy Pruden
Lenwood S. Heath
Na Meng
Weidong Xiao

March 27, 2019
Blacksburg, Virginia

Keywords: Metagenomics, Bioinformatics, Deep learning, Antibiotic resistance genes, Web services

Computational Tools for Annotating Antibiotic Resistance in Metagenomic Data

Gustavo Alonso Arango Argoty

ABSTRACT

Metagenomics has become a reliable tool for the analysis of the microbial diversity and the molecular mechanisms carried out by microbial communities. By the use of next generation sequencing, metagenomic studies can generate millions of short sequencing reads that are processed by computational tools. However, with the rapid adoption of metagenomics a large amount of data has been generated. This situation requires the development of computational tools and pipelines to manage the data scalability, accessibility, and performance. In this thesis, several strategies varying from command line, web-based platforms to machine learning have been developed to address these computational challenges.

Interpretation of specific information from metagenomic data is especially a challenge for environmental samples as current annotation systems only offer broad classification of microbial diversity and function. Therefore, I developed MetaStorm, a public web-service that facilitates customization of computational analysis for metagenomic data. The identification of antibiotic resistance genes (ARGs) from metagenomic data is carried out by searches against curated databases producing a high rate of false negatives. Thus, I developed DeepARG, a deep learning approach that uses the distribution of sequence alignments to predict over 30 antibiotic resistance categories with a high accuracy.

Curation of ARGs is a labor intensive process where errors can be easily propagated. Thus, I developed ARGminer, a web platform dedicated to the annotation and inspection of ARGs by using crowdsourcing.

Effective environmental monitoring tools should ideally capture not only ARGs, but also mobile genetic elements and indicators of co-selective forces, such as metal resistance genes. Here, I introduce NanoARG, an online computational resource that takes advantage of the long reads produced by nanopore sequencing technology to provide insights into mobility, co-selection, and pathogenicity.

Sequence alignment has been one of the preferred methods for analyzing metagenomic data. However, it is slow and requires high computing resources. Therefore, I developed MetaMLP, a machine learning approach that uses a novel representation of protein sequences to perform classifications over protein functions. The method is accurate, is able to identify a larger number of hits compared to sequence alignments, and is >50 times faster than sequence alignment techniques.

Computational Tools for Annotating Antibiotic Resistance in Metagenomic Data

Gustavo Alonso Arango Argoty

GENERAL AUDIENCE ABSTRACT

Antimicrobial resistance (AMR) is one of the biggest threats to human public health. It has been estimated that the number of deaths caused by AMR will surpass the ones caused by cancer on 2050. The seriousness of these projections requires urgent actions to understand and control the spread of AMR. In the last few years, metagenomics has stand out as a reliable tool for the analysis of the microbial diversity and the AMR. By the use of next generation sequencing, metagenomic studies can generate millions of short sequencing reads that are processed by computational tools. However, with the rapid adoption of metagenomics, a large amount of data has been generated. This situation requires the development of computational tools and pipelines to manage the data scalability, accessibility, and performance. In this thesis, several strategies varying from command line, web-based platforms to machine learning have been developed to address these computational challenges. In particular, by the development of computational pipelines to process metagenomics data in the cloud and distributed systems, the development of machine learning and deep learning tools to ease the computational cost of detecting antibiotic resistance genes in metagenomic data, and the integration of crowdsourcing as a way to curate and validate antibiotic resistance genes.

ACKNOWLEDGMENTS

I would like to thank my advisor Dr. Liqing Zhang for helping me through the learning process in my Ph.D. degree. Her knowledge, patience, guidance, and feedback were the core principles to identify and formulate novel solutions to different scientific challenges. I am really grateful for her support as she motivates not only the intellectual but also the personal success. I would like to thank Dr. Amy Pruden and Dr. Peter Vikesland for their valuable support, patience, and guidance towards an interdisciplinary research, Dr. Heath, for his helpful computational advice on the different projects I carried out, and Dr. Na Meng and Dr. Weidong Xiao for their support and their valuable suggestions for completing my degree.

I am also grateful to Mohammad, Tihi, Min, Zhahoa, Hong and Dhoha, my lab mates, for their productive feedback during our meeting sessions as well as their experiences and wisdom at Virginia Tech that made the doctorate journey much easier.

I would like to thank the support given by my siblings Manuel, Miguel, Martha and friends for their support through this long journey, because, without their support it wouldn't be possible.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 INTRODUCTION

The major biological diversity of the planet is composed of microorganisms that continuously evolve (estimated in 3.8 billion years of evolution) [1]. However, most of those microorganisms cannot be cultured in the laboratory by standard procedures. Therefore, the development of culture-independent techniques was needed. Metagenomics has been recognized as a powerful technique for the discovery of the microbial diversity through different environments [2-6]. This technique consists of extracting and sequencing DNA material directly from the environment as a set of short sequence reads with high quality e.g., Illumina technology [7] or long low quality sequence reads e.g., MinION nanopore sequencing technology [8]. Metagenomics has been applied to many different environments, such as soil [9], water [10], wastewater [11-13], human gut [14-16], and extreme environments [17, 18]. However, the analysis of the metagenomics data has several challenges. First, when short sequences are directly compared to reference databases, it is normally processed by using a high identity cutoff [19]. This cutoff has the potential to filter out reads that are indeed real signals [20, 21]. Second, the quality of the reference database plays an essential role in the data analysis [20, 22-26]. Errors in the database can lead to misleading conclusions. Third, metagenomic data is often large. The growing data generation (MGRast: 53,439 samples, EBI-metagenomics: 87,972 samples) requires the development of accurate and fast computational analysis tools. Finally, third generation sequencing techniques, particularly MinION nanopore sequencing, offers the opportunity to explore the microbial mechanisms and diversity to the species resolution. However, its low coverage and high sequencing error rates limit its wide application.

Antimicrobial resistance currently causes hundreds of thousands of deaths each year globally and it has been recognized as a serious threat to public health [27, 28]. Antibiotic resistance arises when bacteria are able to survive antibiotic exposure. The spread of antibiotic resistance genes (ARGs) occurs by direct contact among different environmental microbiota [29] with horizontal gene transfer (HGT) as one of the major mechanisms of dissemination [30]. Characterizing the antibiotic resistance genes (resistome) and their dissemination profile across environments could lead to the detection of antibiotic resistance genes that potentially affect human health [29].

1.1 Problems

Antimicrobial resistance profiling using metagenomic data is still a growing area with many research opportunities. In this aspect, we identified five different research challenges that are explained in detail in this thesis.

- **Customizable metagenomics annotation (Chapter 2,** *published***)**: With the growing efforts in the understanding of antimicrobial diversity, the functional role of microbes and

their contribution in different environments, MetaStorm (**Chapter 2**) a public web service pipeline for the analysis of metagenomic data was developed. The novelty of this system is its ability to perform customized analysis specific to any particular research question. For instance, researchers interested in antimicrobial resistance would find it useful to analyze the metagenomics samples using databases specific to antibiotic resistance (e.g., CARD, and ARDB). Other systems such as MGRast and EBI-metagenomics do not allow user customization. This capability proves to be extremely useful as many metagenomic applications tend to focus on a specific set of genes in the database. In addition, MetaStorm provides enhanced visualization that allows users to view their analysis results and download publishable figures and tables.

- **Improving antibiotic resistance annotation (Chapter 3,** *published***)**: Identifying ARGs is usually done through the best hit homology search approach. This method conducts a sequence homology search and filters out all the sequences that are below certain thresholds. For instance, MetaStorm and ARGs-OAP [19, 20] use a 80% identity, 1e-5 e-value, and 25aa minimum alignment size, while Li, et al., [31] use a 90% identity, and Kleinheinz et al., [32] uses 50% identity to identify antibiotic resistance genes in phage genomes. All these combinations have advantages and disadvantages [21]. For instance, a high identity cutoff would identify only those sequences that are highly similar to the ARGs in the reference database. Because of the high microbial diversity of environmental samples, it is possible to filter out sequences that are not highly similar to the reference ARG sequences but are indeed ARGs. On the other hand, when using low cutoffs (50% identity), it is possible to classify a sequence as an ARG when it is not an ARG (e.g., efflux pump multidrug genes). Therefore, using a simple identity cutoff is not an effective way to identify ARGs in metagenomic samples. To address this issue, a machine learning method named deepARG is proposed. The method takes the full distribution of the sequence alignments over the database to make a decision using deep learning. DeepARG proved to be effective to filter out both false positives and false negatives through different scenarios [see **Chapter 3**].
- **Curation of antibiotic resistance genes (Chapter 4)**: One of the most critical aspect of the antimicrobial resistance analysis using metagenomic data is the quality of the reference database. Curation of ARGs is a labor-intense task that requires expert knowledge and is generally performed by individual labs. Therefore, it is not surprising that ARG databases contain multiple inconsistences. The most up-to-date database is the Comprehensive Antibiotic Resistance Database (CARD) [22]. However, there is not a clear protocol on how ARGs should be inspected and annotated. To overcome these limitations, a crowdsourcing system for the inspection and annotation of ARGs is proposed. The goal of this project is to build a web-based service where curators (experts and nonexperts) can perform simple tasks and contribute to the annotation of ARGs.
- **Annotation of long nanopore reads (Chapter 5)**: The mechanisms of antibiotic resistance propagation and dissemination cannot be fully understood by looking at ARGs

alone [31, 33-36]. Indeed, environmental contamination can trigger mechanisms such as co-resistance (linkage between ARGs), cross-resistance (single gene conferring resistance to antibiotics and metals), co-regulation (shared regulatory system conferring resistance to antibiotics and metals) and co-transfer (dissemination of resistance by mobile genetic elements). An alternative to discover the mechanisms consists of assembling short sequence reads into longer contigs. Thus, ARG arrangement and co-occurrence with other elements can be characterized. However, the high diversity in metagenomic data and the similarities between related species can introduce various assembly errors (e.g., chimeric contigs) that adversely affect the inference of co-occurrence of ARGs and other elements based on the assembled contigs. Fortunately, with the development of the nanopore technique [37], a third-generation sequencing technology that generates longer reads, it is possible to inspect the context of the antimicrobial resistance in a much higher resolution (e.g., species profiling). On the other hand, nanopore sequencing is far from being perfect. It has a high error rate up to 35% [38]. To overcome these limitations and to provide a pipeline for a comprehensive analysis of the antimicrobial resistance in metagenomic data, a pipeline and web service is proposed. The system, named NanoARG, is designed to analyze nanopore reads using a novel antibiotic resistance annotation method (**Chapter 3**), the first release of our antimicrobial database (**Chapter 4**), and the common practices for metagenomic annotations (**Chapter 2**).

- **Speeding up the metagenomics annotation (Chapter 6)**: The accelerated production of sequencing data requires the development of fast and sensitive systems [39]. To date, most metagenomics analysis is performed by using sequence alignments [40], which, although effective, are computationally expensive [41]. Here, a computational method to perform functional annotation of metagenomic samples using any reference database (e.g., biological process, antibiotic resistance, metal resistance, or mobile genetic elements) is proposed. Our approach named MetaMLP (Metagenomics Machine Learning Profile) uses a simple but effective text classification technique based on text embeddings [42, 43] particularly popular in Natural Language Processing (NLP). Results show that MetaMLP is as accurate as DIAMOND, a widely used sequence alignment tool, and it is >50x times faster. In addition, the comparison of relative abundances computed from DIAMOND and MetaMLP show a high correlation (>0.9), indicating that MetaMLP produces very similar results to sequence alignment but much faster.

1.2 Overall aim

To develop computational tools (command line and web services) that facilitate the data processing, sharing, and interpretability of microbial metagenomes, promoting the understanding of the roles of antimicrobial resistance in different contexts.

1.3 Specific aims

1. To build an user friendly system dedicated to the analysis of metagenomics samples that incorporates current practices and allows user customization (**Chapter 1**).
2. To build a machine learning system capable of improving the annotation of antibiotic resistance genes without being sensitive to the percentage of identity (**Chapter 2**).
3. To build a comprehensive antibiotic resistance database that incorporates crowdsourcing as a validation strategy (**Chapter 3**).
4. To quantify the diversity of antibiotic-, metal-resistance, and mobile genetic elements, as well as their interactions from long nanopore read sequences. In addition, to develop an online platform to perform and retrieve several metagenomics analysis including: co-occurrence, taxonomy annotation and detection of critically important pathogens (**Chapter 4**).
5. To develop and implement a computational system for functional annotation of metagenomes, capable of processing millions of reads in a short period of time (**Chapter 5**).

CHAPTER 2 CUSTOMIZABLE METAGENOMICS ANNOTATION

Metagenomics is a trending research area, calling for the need to analyze large quantities of data generated from next generation DNA sequencing technologies. The need to store, retrieve, analyze, share, and visualize such data challenges current online computational systems. Interpretation and annotation of specific information is especially a challenge for metagenomic data sets derived from environmental samples, because current annotation systems only offer broad classification of microbial diversity and function. Moreover, existing resources are not configured to readily address common questions relevant to environmental systems. Here, we developed a new online user-friendly metagenomic analysis server called MetaStorm (http://bench.cs.vt.edu/MetaStorm/), which facilitates customization of computational analysis for metagenomic data sets. Users can upload their own reference databases to tailor the metagenomics annotation to focus on various taxonomic and functional gene markers of interest. MetaStorm offers two major analysis pipelines: an assembly-based annotation pipeline and the standard read annotation pipeline used by existing web servers. These pipelines can be selected individually or together. Overall, MetaStorm provides enhanced interactive visualization to allow researchers to explore and manipulate taxonomy and functional annotation at various levels of resolution.

2.1 Introduction

The field of metagenomics has arisen following the advent of next-generation DNA sequencing. Through new technologies, such as Illumina and pyrosequencing, it is now possible to directly shot-gun sequence DNA extracted from various environmental samples, without the need for cloning. Metagenomics is particularly promising for advancing the understanding of the microbial communities molecular mechanisms residing in natural, human, and engineered environments. To date, metagenomic data sets have been obtained from different regions of the human body [44-46], seas and oceans [47-49], lakes and rivers [50-52], wastewater and drinking water treatment systems [12, 13, 53, 54], soil [3, 55], and air [56, 57]. Unlike single organismal genomic characterization, metagenomic data sets contain DNA sequences derived from hundreds or even thousands of microbial species [58, 59]. Thus, a major computational undertaking is to annotate metagenomic samples in terms of the kinds of microbes (taxonomy) and genes (functional annotation), particularly those that are present in complex environmental samples.

Various computational resources have been developed for taxonomic and functional annotation of metagenomics data sets. These resources can be classified into two main categories: 1) Web services organized as a collection of different computational resources that facilitate the storage, analysis, and retrieval of metagenomic data (e.g., MG-RAST [60] and EBI-Metagenomics [5]); and 2) stand-alone programs for various aspects of metagenomic data annotation (e.g., MEGAN [61], MOCAT [62], QIIME [63], MetaPhlAn [64], MetaHIT [65], and MyTaxa [66]), which have been commonly incorporated into Web services. Generally, current services (MG-RAST and EBI-

Metagenomics) annotate metagenomic samples by matching raw sequences against a fixed set of large reference sequence databases (e.g., UniProtKB [67], and Clusters of Orthologous Groups of proteins (COG) [68]. This practice has two major limitations. First, there is a lack of user customization, particularly the inability to select specific sets of genes. Thus, all annotations are made with respect to the same reference databases, which may not be the most suitable depending on the hypotheses driving the research. The ability to select and focus on desired sets or subsets of reference sequences enables testing of domain-specific hypotheses. For instance, conclusions of studies of antibiotic resistance gene occurrence in the environment (e.g., [69]) can vary depending on the database selected, i.e., CARD [22], a specialized antibiotic resistance gene database, versus the full GenBank database. Second, due to short sequence length, the ability to assemble reads can be critical to identifying genes of interest and avoiding loss of information. The assembly of raw reads into longer contigs or scaffolds has proved to be more effective for annotating sequence features such as operons, transcription binding sites, chromosome organization, and taxonomy [70].

Here, we introduce a new online metagenomic analysis server, MetaStorm, which improves available web resources, particularly for environmental samples, while maintaining a user-friendly interface. MetaStorm offers both read matching and assembly-based annotation pipelines, while also enabling customization of reference databases. This allows users to upload databases containing curated genes of interest to facilitate functional and taxonomic annotation. MetaStorm also provides enhanced visualization of annotation results, allowing the user to explore and manipulate taxonomic and functional annotations at various levels of resolution and to compare annotation for similarities and differences across multiple data samples using various graphs.

2.2 Materials and Methods

Raw data is submitted to the MetaStorm server via a user-friendly web interface. Submitted data can remain private or be made public depending on user preference. Users are required to create an account and a profile. This profile allows them to retrieve, submit, analyze, and compare not only their own samples but also other public projects. MetaStorm stores the metagenomics samples and results into user projects which describe the features of the metagenomic experiments. If a project is made public, the raw and any associated results are free for download.

2.2.1 Required data types

MetaStorm requires the user to upload raw sequences in the widely-used FASTQ format [71]. Any high-throughput DNA sequencing technology (e.g., amplicon or shotgun sequencing) is accepted. Provision of detailed metadata associated with the samples from which the DNA sequences were derived is mandatory during the submission process. Provision of metadata is critical to help users identify similar studies that are already in the MetaStorm repository for additional sample

comparisons. Data is organized in a manner that facilitates retrieval. A project may contain several samples and each sample may be nested with several associated studies within it (e.g., taxonomy annotation, antibiotic resistance, or any functional annotation using both assembly and read matching pipelines). All user, sample, and project information is stored in a relational database.

2.2.2 Reference database

Apart from a set of standard databases (e.g., CARD [22], UniProtKB [72], and GREENGENES [73]) (Table 2.1), MetaStorm also allows users to upload and use their own customized databases as reference databases. The customizability of reference databases is especially useful when researchers seek to test a hypothesis by comparison against a very specific set of sequences. Neither MG-RAST nor the EBI-metagenomics Web service allows for customized reference databases. In this way, MetaStorm enhances user control by allowing them to select reference sequences.

Database	Source	Type	#IDs	annotation
UniProtKB	http://www.uniprot.org/help/uniprotkb	protein	551,705	function
CARD	http://arpcard.mcmaster.ca/	protein	4,120	function
ACLAME	http://aclame.ulb.ac.be/	protein	122,154	function
BACMET	http://bacmet.biomedicine.gu.se/	protein	444	function
CAZy	http://www.cazy.org/	protein	281,237	function
SILVA	http://www.arb-silva.de/	nucleotide	1,756,783	taxonomy
COG	http://www.ncbi.nlm.nih.gov/COG/	protein	346,378	function
GREENGENES	http://greengenes.lbl.gov/cgi-bin/nph-index.cgi	nucleotide	1,262,986	taxonomy

Table 2.1: Default reference databases provided by the MetaStorm Web service

2.2.3 Web-based submission

Submission of metagenomic data is made by an interactive web interface (**Figure 2.1**). Users are first required to log into the MetaStorm web site, select (or create) the project they wish to analyze, and select the desired method (Assembly/Read matching). Once in the project profile page, users need to insert sample information (number of samples, name of the samples, conditions, environment, and library preparation), select reference databases, upload raw FASTQ files, and finally run the annotation pipeline. To simplify the process of data submission, MetaStorm does not require external files such as Excel spreadsheets for sample description and provision of metadata (although this functionality can be easily added in a future update if necessary). This interactive tool also allows users to remove samples and projects or re-run the samples with different pipelines, visualizing the results as needed.

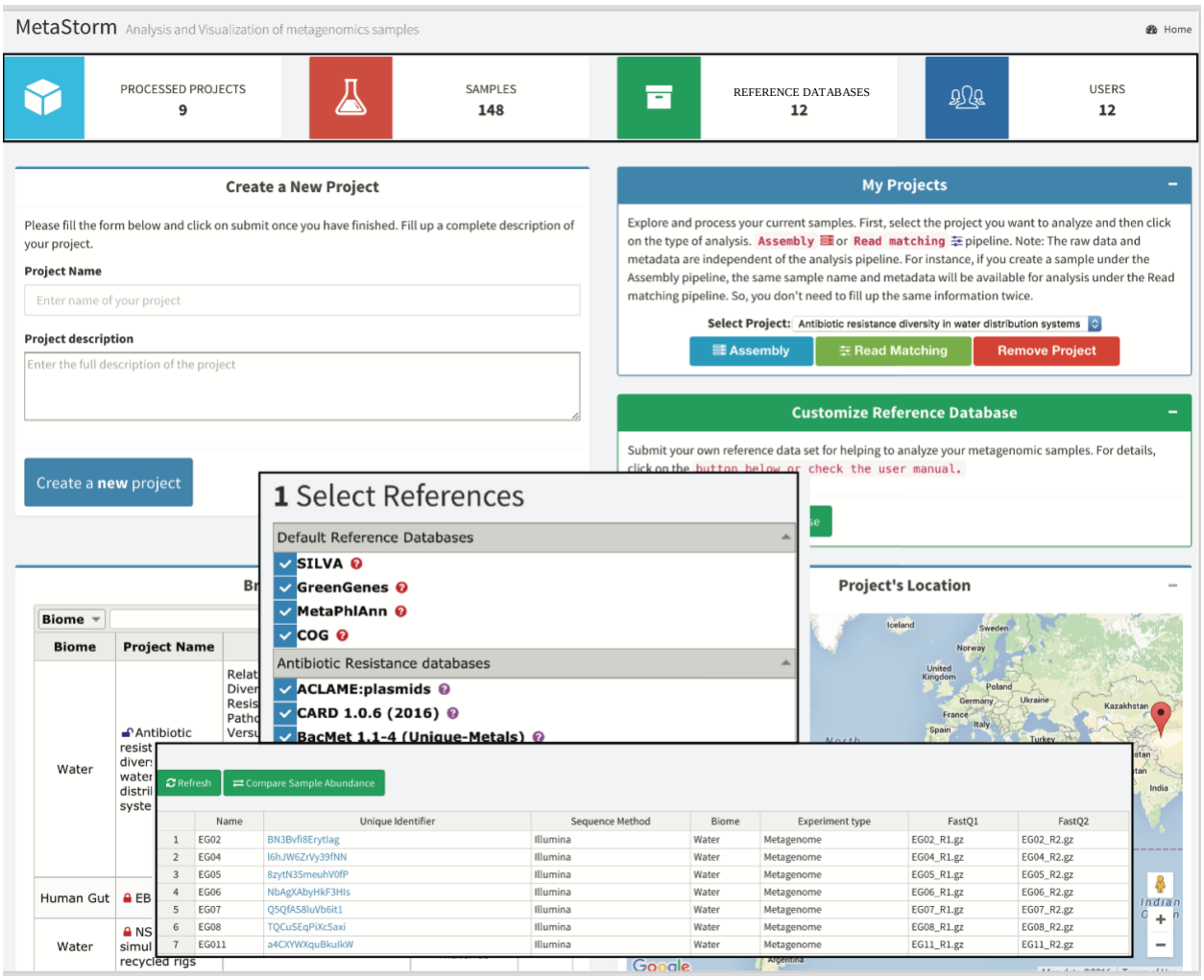

Figure 2.1: Main user interface of MetaStorm. Create a new project allows to submit a project under the user profile. My Projects grant access to the data management interface that includes: Upload raw files, add samples, remove samples, visualize individual samples and compare samples. Customize Reference Database gives access to the form for uploading a customized reference database. Browse projects allows to find samples by biome and/or location. Comparison tool allows users to compare samples from different projects. Profile allows users to modify their personal information and password.

2.2.4 Analysis pipeline

Once stored in the MetaStorm server, raw reads are queued for taxonomic and functional annotations. MetaStorm incorporates two pipelines, the assembly-based pipeline and the read-matching pipeline (**Figure 2.2**). Selecting the appropriate pipeline depends of several parameters including: the design of the experiment, the previous knowledge about the experiment, the research hypothesis and goals. For instance, if the objective is to characterize the most abundant taxonomy in the community, the assembly pipeline may suffice [58].

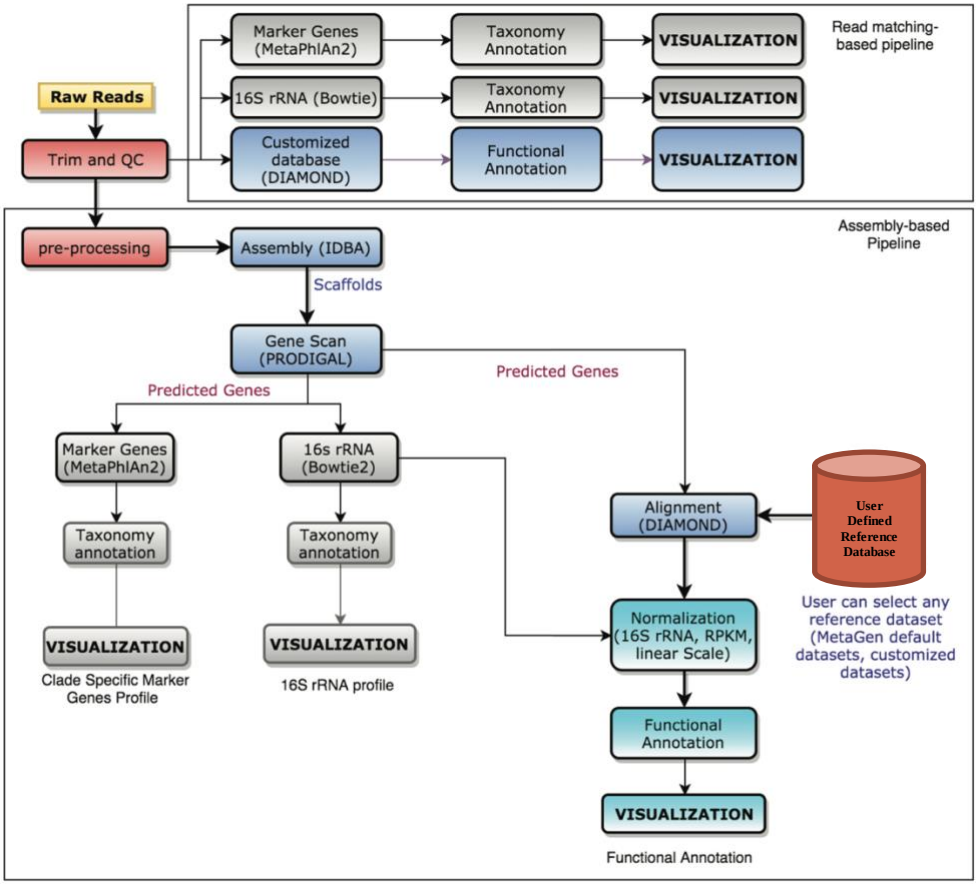

Figure 2.2 Pipelines. Overview of the computational pipelines implemented in the MetaStorm service for taxonomic and functional annotation.

2.2.4.1 Assembly pipeline

Through the assembly process, metagenomics reads are merged into large contiguous sequences varying in length from several hundred bases to nearly complete genomes providing much richer information relative to the raw reads [58, 59]. MetaStorm provides a fully automated assembly pipeline that allows the user to visualize, compare, and analyze the taxonomy and functional content of a sample or set of samples by matching and computing the abundance. The pipeline for assembly and gene finding is similar to the methods reported from the MetaHIT consortium [65] (mainly the metagenome assembly and gene prediction through scaffolds). This pipeline consists of the following major procedures:

1. Quality control (QC): reads are trimmed and filtered out by TRIMMOMATIC [74] to remove low quality sequences from the data set.

2. Assembly: IDBA-UD [75] is a widely used metagenome assembler that has demonstrated consistent production of high quality scaffolds [76-78]. IDBA-UD is used to assemble the QC filtered reads. MetaStorm uses the default parameters.
3. Gene prediction: Once a set of scaffolds are assembled, PRODIGAL [79] (metagenomics version), a microbial gene finding program, is deployed to predict genes within each scaffold.
4. Taxonomy annotation: Predicted genes are matched to a reference database using two alignment tools (BLAST [80] and DIAMOND [40]). Currently included are the following databases:
    a. Two 16S rRNA databases (SILVA [81] and GREENGENES [73]). The 16S rRNA gene abundance is computed by first selecting the best hit (same definition as in MG-RAST representative hit [82]) to the scaffold-genes from the reference database using BLASTN [80] and then computing the number of genes that each taxa contains (E-Value<1e-10, identity >90%). Note that the taxonomy profile is computed based on the abundance of predicted genes, not the number of reads.
    b. A set of marker genes processed by the MetaPhlAn2 [83] pipeline. This technique is included because whole genome sequencing samples typically contain very low 16S rRNA sequence content [65, 66, 83].

5. Functional annotation: Predicted genes (translated proteins from PRODIGAL) are matched to the user selected reference databases using the DIAMOND BLASTP aligner [40] [40]. We use the representative hit strategy with an E-value<1e-10, identity>60% over the entire length [84], and minimum length of 25aa. The reference sequence databases for functional annotation depend on user criteria. For instance, a user interested in antibiotic resistance genes may prefer to run the analysis over the CARD database [22], whereas a project related to the degradation process may use the CAZy database [85].

2.2.4.2 Read matching pipeline

The read matching pipeline conducts taxonomic and functional annotation of metagenomic data comparing the raw sequence reads to a reference database. This approach is also called *marker gene analysis* [58]. For taxonomy annotation, MetaStorm uses a matching scheme similar to MG-RAST and EBI-metagenomic where reads are first trimmed out and quality filtered using TRIMMOMATIC [74] and then mapped to a 16S rRNA sequence database (SILVA or GREENGENES). To speed up the read matching process, we use Bowtie2 [86], a fast and sensitive read matching tool specialized for mapping short reads to reference genomes (--local-sensitive, identity>90%, best-hit-alignment). It has proven to be particularly efficient for matching marker gene databases; MetaPhlAn2 [44] using Bowtie2 for read matching produced more accurate results than its earlier version MetaPhlAn1 [64] that uses BLAST. MetaPhlAn2 [83] which uses a set of clade specific genes, is also offered by MetaStorm to estimate the taxonomic abundance.

Functional annotation is made comparing the high quality reads to the reference database using the DIAMOND BLASTX [40] aligner with the representative hit approach [82] (E-value<1e-10, identity>90%, and minimum length of 25aa).

2.2.5 Sample normalization and comparison

Sample comparison consists of the analysis of relative abundance through a set of samples, allowing the user to visualize similarities and differences among samples. One of the critical aspects of sample comparison is data normalization. MetaStorm implement three different normalization techniques as follows:

1. Scaling: Normalize the number of matches obtained per sample, with relative abundance between 0 and 100.
2. RPKM: Normalize the number of matches using the Reads per Kilobase per Million Mapped Reads of each gene.
3. Relative to 16S rRNAs: We use the normalization concept described in [69], which defines the relative abundance as the copy of a functional gene per copy of 16S rRNA genes.

Normalizations are calculated differently for both pipelines. For the assembly-based pipeline, all the computations are made in terms of number of *matched genes* whereas the read-matching pipeline normalizes the samples using the number of *matched reads*.

2.2.6 Visualization of taxonomic abundance

MetaStorm offers interactive visualization, allowing users to see in detail the main features of the sequence context of each sample. A taxonomic tree encodes relative abundance information of different lineages in the sample. For example, in **Figure 2.3**, a user interested in the relative abundance of various kinds of *Proteobacteria* will find that the genus *Achromobacter* is the most abundant. Unlike other metagenomic tools, such as MG-RAST and EBI-metagenomics, we allow interactive visualization to improve the user experience. In particular, the tree allows users to keep track of of various levels of the phylogenetic hierarchy. Also, when the user clicks on any specific node (taxa), all descendants from that node will be displayed as a pie chart. The overall abundance of a taxonomy level can also be displayed as a pie chart. Node colors represent relative abundance. All visualization formats are available for the taxonomic annotation methods.

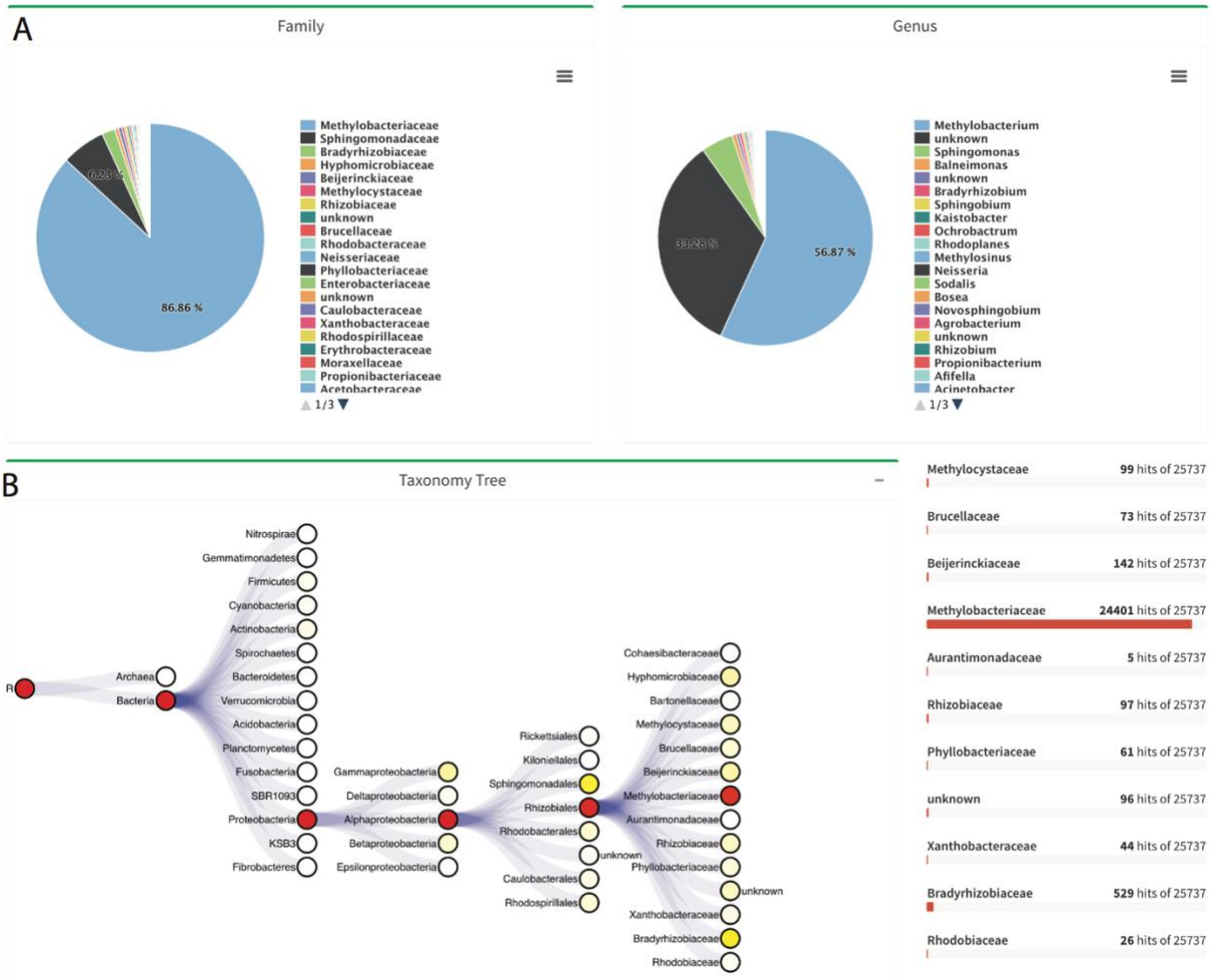

Figure 2.3. **Taxonomy visualization.** Taxonomy levels are shown as pie charts (only Family and Genus are shown for illustration). The interactive tree allows users to follow the path of the abundant taxas and the chart displays the selected taxonomy level. The right panel shows the hits distribution to the open node in the taxonomy tree. In this example, the families under the order *Rhizobiales* are shown in the left panel.

2.2.7 Visualization of functional abundance

Functional relative abundance is described by a set of interactive pie charts and bar plots (**Fig 4 A**) that relate functional categories with the genes involved in each category. Users can select the reference database to analyze and all the tables in text format can be downloaded. When analyzing individual samples, read/gene counts are normalized using a linear scale between 0 to 100.

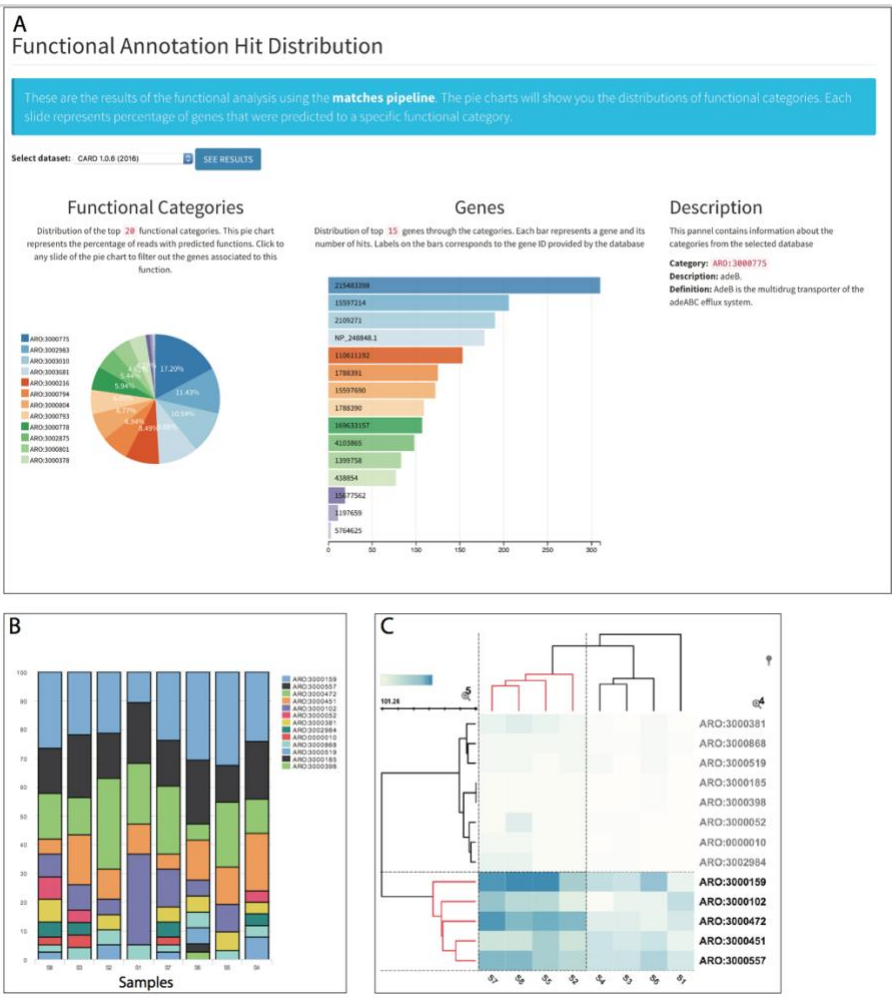

Figure 2.4. Functional and sample comparison visualization. (A) Functional annotation is depicted by a pie chart, where the user can select the database to visualize. (B) Sample comparison visualization using stacked bars for both taxonomy and function. (C) interactive heat map visualization where users can click on the branches to zoom over the related functions or taxas.

2.2.8 Visualization of sample comparison

Visualization techniques employed by MetaStorm include: heat maps, stacked bars, and interactive trees (taxonomy annotation). As for single sample visualization, the response tree shows relative abundance for each node (taxa) and also for each taxonomic hierarchical level, allowing a high level of specificity. This type of interactive visualization features (**Fig 4 B and 4 C**) is not available in other visualization tools, such as MG-RAST or EBI-Metagenomics.

2.3 Data access

Similar to MG-RAST and EBI-Metagenomics, all the information on a project tagged public, such as raw read files, processed files, description files, and visualization tables, are freely available through MetaStorm. From the home page, the user can access descriptions of all the recently listed (public) projects and the reference databases that other users submitted. A search tool is available for users to identify potential sets of reference sequences that can match their analysis. MetaStorm's reference sharing capability aims to support **1**) the focus of knowledge based on user runs and **2**) the projected run time for reporting MetaStorm results. Expectedly, small customized databases will report results faster than full reference databases. A novice user can use this database for analysis and jump to the specific biological problem, thus saving the computing time. Moreover, the search tool enables users to find similar existing metagenome samples in MetaStorm (public ones) and include them for more comprehensive comparison studies. Comparison across different samples is made feasible by the normalization criteria implemented in MetaStorm. Finally, all the raw and generated files for the metagenomic analysis can be downloaded in a variety of formats by clicking on the download button of each section in the visualization page.

2.4 Results and Discussion

Compared to other metagenomic resources, such as MG-RAST and EBI-metagenomics, MetaStorm extends the analysis and visualization of metagenomic samples by: 1) adding a fully developed assembly-based annotation pipeline, in addition to the read matching pipeline deployed by these Web servers; 2) offering a customized analysis where the user can select and upload reference databases, which enables focus on specific genes of interest as well as inter-project comparison; and 3) interactive visualization capabilities, including an interactive taxonomic tree, which permit users to interrogate and compare specific aspects of the sequence data. MetaStorm includes a wide variety of databases used for metagenomics analysis (section customizable reference database). Those databases have been used as default by several current metagenomics resources. While the assembly pipeline implemented by MetaStorm is similar to that of the MetaHIT pipeline [65], it incorporates a more meaningful relative abundance determination in which copies are normalized to 16S rRNA gene copies [69] . Normalization enables comparison across multiple metagenomics data sets, including those generated by external labs, empowering researchers to address broad. This last feature is particularly promising for the future applicability of the MetaStorm server.

2.5 Conclusion

MetaStorm is a free and public metagenomics resource that enables a more specific user customization through various improvements of visualization, data management, and user interactivity. MetaStorm offers two main metagenomic analysis pipelines: the read matching pipeline (similar to the current web resources) and the assembly pipeline. MetaStorm, unlike any

other web resources, incorporates user reference customization, which will help to streamline the annotation process when a research hypothesis requires specific and customized databases.

CHAPTER 3 IMPROVING ANTIBIOTIC RESISTANCE ANNOTATION

Growing concerns about increasing rates of antibiotic resistance call for expanded and comprehensive global monitoring. Advancing methods for monitoring of environmental media (e.g., wastewater, agricultural waste, food, and water) is especially needed for identifying potential resources of novel antibiotic resistance genes (ARGs), hot spots for gene exchange, and as pathways for the spread of ARGs and human exposure. Next-generation sequence now enables direct access and profiling of the total metagenomic DNA pool, where ARGs are typically identified or predicted based on the "best hits" of sequence searches against existing databases. Unfortunately, this approach produces a high rate of false negatives. To address such limitations, we propose here a deep learning approach, taking into account a dissimilarity matrix created using all known categories of ARGs. Two deep learning models, DeepARG-SS and DeepARG-LS, were constructed for short read sequences and full gene length sequences, respectively. Evaluation of the deep learning models over 30 antibiotic resistance categories demonstrates that the DeepARG models can predict ARGs with both high precision (>0.97) and recall (>0.90). The models displayed an advantage over the typical best hit approach, yielding consistently lower false negative rates and thus higher overall recall (>0.9). As more data become available for under-represented ARG categories, the DeepARG models' performance can be expected to be further enhanced due to the nature of the underlying neural networks. Our newly developed ARG database, DeepARG-DB, encompasses ARGs predicted with a high degree of confidence and extensive manual inspection, greatly expanding current ARG repositories. The deep learning models developed here offer more accurate antimicrobial resistance annotation relative to current bioinformatics practice. DeepARG does not require strict cutoffs which enables identification of a much broader diversity of ARGs. The DeepARG models and database are available as a command line version and as a Web service at http://bench.cs.vt.edu/deeparg.

3.1 Introduction

Antibiotic resistance is an urgent and growing global public health threat. It is estimated that the number of deaths due to antibiotic resistance will exceed ten million annually by 2050 [28]. Antibiotic resistance arises when bacteria are able to survive an exposure to antibiotics that would normally kill them or stop their growth. This process allows for the emergence of 'superbugs' that are extremely difficult to treat. A few examples include methicillin-resistant Staphylococcus aureus (MRSA), a drug-resistant bacteria associated with several infections [87], multidrug-resistant (MDR) Mycobacterium tuberculosis, which is resistant to rifampicin, fluoroquinolone, and isoniazid [88], and colistin-carbapenem-resistant Escherichia coli, which has gained resistance to last-resort drugs through the acquisition of the *mcr-1* and bla_{NDM-1} antibiotic resistance genes (ARGs) [89, 90].

The advent of high throughput DNA sequencing technology now provides a powerful tool to profile the full complement of DNA, including ARGs, derived from DNA extracts obtained from a wide range of environmental compartments. For example, ARGs have now been profiled using this kind of metagenomic approach in livestock manure, compost, wastewater treatment plants, soil, water, and other affected environments [91-96], as well as within the human microbiome [29, 97]. Identification of ARGs from such samples is presently based on the computational principle of comparison of the metagenomic DNA sequences against available online databases. Such comparison is performed by aligning raw reads or predicted open reading frames (full gene length sequences) from assembled contigs to the database of choice, using programs such as BLAST [98], Bowtie [86], or DIAMOND [40], and then predicting or assigning the categories of ARGs present using a sequence similarity cutoff and sometimes an alignment length requirement [20, 99, 100].

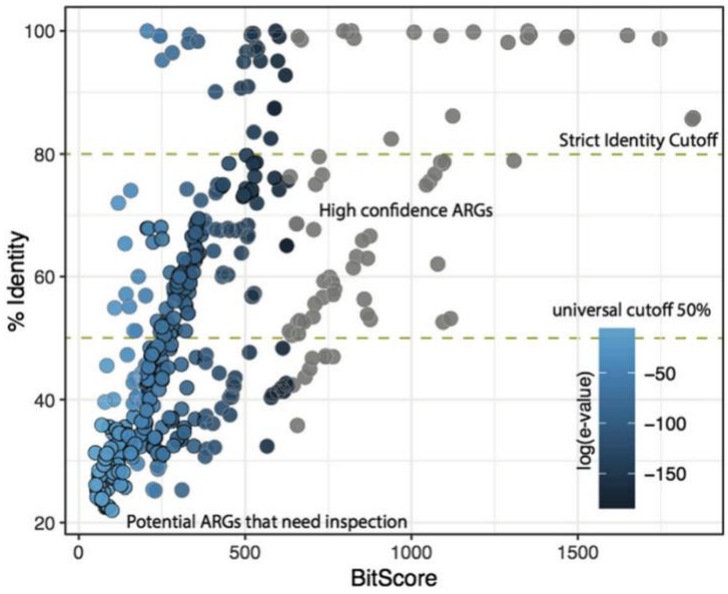

Figure 3.1. Bit score vs. identity distribution, illustrating the relationship between the UNIPROT genes against the CARD and ARDB genes in terms of the percentage identity, bit score, and e-value. Colors depict the exponent of the e-value.

Existing bioinformatics tools focus on detecting known ARG sequences from within genomic or metagenomic sequence libraries and thus are biased towards specific ARGs [101]. For instance, ResFinder [99] and SEAR [102] predict specifically plasmid-borne ARGs, and Mykrobe predictor [103] is dedicated to 12 types of antimicrobials, while PATRIC [100] is limited to identifying ARGs encoding resistance to carbapenem, methicillin, and beta lactam antibiotics. Most of these tools use existing microbial resistance databases along with a "best hit" approach to predict whether a sequence is truly an ARG. Generally, predictions are restricted to high identity cutoffs, requiring a best hit with an identity greater than 80% by many programs such as ResFinder [99] and ARGs-OAP [91, 99, 104]. In some studies the identity cutoff is even higher, as high as 90%

for determining structure and diversity of ARGs through several resistomes [91] or analyzing the co-occurrence of environmental ARGs [69].

Although the best hit approach has a low false positive rate, that is, few non-ARGs are predicted as ARGs [92], the false negative rate can be very high and a large number of actual ARGs are predicted as non-ARGs [101, 104]. Figure 3.1 shows the distribution of manually-curated potential ARGs from the Universal Protein Resource (UNIPROT) database [105] against the Comprehensive Antibiotic Resistance Database (CARD) [106] and the Antibiotic Resistance Genes Database (ARDB) [107]. All of the gene comparisons indicate significant e-values < 1e-20 with the sequence identity ranging from 20% to 60% and bit scores > 50, which is considered statistically significant [84]. Thus, high identity cutoffs clearly will remove a considerable number of genes that in reality are ARGs. For example, the entry O07550 (Yhel), a multidrug ARG conferring resistance to doxorubicin and mitoxantrone, has an identity of 32.47% with a significant e-value of 6e-77 to the best hit from the CARD database; the gene POCOZ1 (VraR), conferring resistance to vancomycin, has an identity of only 23.93% and an e-value 9e-13 to the best hit from the CARD database. Therefore, more moderate constraints on sequence similarity should be considered to avoid an unacceptable rate of false negatives. On the other hand, for short metagenomic sequences (e.g., ~ 25aa or 100bp), a stricter identity constraint of ~80% is recommended [84, 99] to avoid a high false positive rate. In principle, the best hit approach works well for detecting known and highly conserved categories of ARGs but may fail to detect novel ARGs or those with low sequence identity to known ARGs [20, 108].

To address the limitation of current best hit methodologies, a deep learning approach was used to predict ARGs, taking into account the similarity distribution of sequences in the ARG database, instead of only the best hit. Deep learning has proven to be the most powerful machine learning approach to date for many applications, including image processing [109], biomedical signaling [110], speech recognition [111], and genomic-related problems, such as the identification of transcription factor binding sites in humans [112, 113]. Particularly in the case of predicting DNA sequence affinities, the deep learning model surpasses all known binding site prediction approaches [112]. Here, we develop, train, and evaluate two deep learning models, DeepARG-SS and DeepARG-LS, to predict ARGs from short reads and full gene length sequences, respectively. The resulting database, DeepARG-DB, is manually curated and is populated with ARGs predicted with a high degree of confidence, greatly expanding the repertoire of ARGs currently accessible for metagenomic analysis of environmental data. DeepARG-DB can be queried either online or downloaded freely to benefit a wide community of users and to support future development of antibiotic resistance-related resources.

3.2 Materials and Methods

3.2.1 Database Merging

The initial collection of ARGs was obtained from three major databases: CARD [106], ARDB [107], and UNIPROT [105]. For UNIPROT, all genes that contained the Antibiotic Resistance keyword (KW-0046) were retrieved, together with their metadata descriptions when available. All identical or duplicate sequences were removed by clustering all the sequences (ARDB + CARD + UNIPROT) with CD-HIT [114], discarding all except one that had 100% identity and the same length. The remaining set of sequences comprised a total of 2,290 genes from ARDB (50% of the original ARDB genes), 2,161 from CARD (49% of the original CARD genes), and 28,108 from UNIPROT (70% of the original UNIPROT genes). This indicates a high redundancy of sequences within and among the ARG databases.

3.2.2 ARG Annotation of CARD and ARDB

The ARDB and CARD databases both contain information to aid in the classification of ARGs, including the antibiotic category to which a gene confers resistance (e.g., macrolides, beta lactamases, or aminoglycosides) and the antibiotic group to which the gene belongs (e.g., *tetA*, *sul1*, *macB*, *oxa*, *mir*, or *dha*). Manual inspection revealed that some genes have been assigned to specific sets of antibiotics instead of antibiotic resistance categories. For instance, carbapenem, carbenicillin, cefoxitin, ceftazidime, ceftriaxone, and cephalosporin are actually a subset of the beta lactamases category. Thus, a total of 102 antibiotics that were found in the ARDB and CARD databases were further consolidated into 30 antibiotic categories.

3.2.3 UNIPROT Gene Annotation

Compared to the ARGs in CARD and ARDB, the UNIPROT genes with antibiotic resistance keywords are less well curated. Therefore, additional procedures were applied to further annotate the UNIPROT genes. Specifically, based on the CD-hit [114] clustering results, clusters that contained only UNIPROT genes were classified into two categories: 1) those without any annotation were tagged as “unknown” and 2) those with descriptions were text mined to identify possible association with antibiotic resistance.

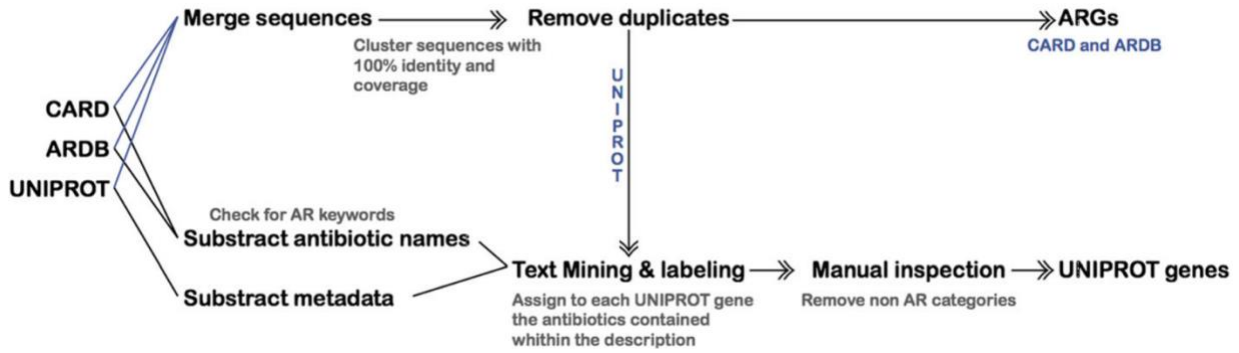

Figure 3.2: Preprocessing and UNIPROT ARGs annotation. Antibiotic resistance genes from CARD, ARDB, and UNIPROT were merged and clustered to remove duplicates. Then sequences from UNIPROT are annotated using the matches between the metadata and the names of antibiotic categories from ARDB and CARD.

UNIPROT's sequence description contains a variety of features including a description of possible functions of the protein, the gene name based on HUGO nomenclature [115] for each sequence, and the evidence indicating whether a sequence has been manually inspected or not. A text mining approach was used to mine the genes' descriptive features to identify their antibiotic resistance associations with the 30 antibiotic categories. The Levenshtein distance [116] was used to measure the similarities between gene description and antibiotic categories. This text mining approach was used because the names of the antibiotic resistance categories are not standardized among the databases and flexibility is needed to identify as many antibiotic associations as possible. For instance, genes linked to beta lactamases were sometimes tagged as beta-lactam, beta-lactamases, or beta-lactamase. Thus, text mining using all the alternative words allows comprehensive identification of antibiotic associations for each gene. Using this strategy, genes from UNIPROT were tagged either to their antibiotic resistance associations based on their description, or to "unknown" if no link to any antibiotic was found. Then manual inspection was performed to remove misleading associations that passed the similarity criteria. The final set of genes and their tagged antibiotic resistance categories are shown in Figure 3.2. Altogether, 16,360 UNIPROT genes remained after this refinement procedure.

The text mining procedure enabled the UNIPROT genes to become linked to one or more categories of antibiotics. However, the text mining procedure is purely based on gene metadata. Therefore, there was no evidence at the sequence level that the UNIPROT genes were truly associated with antibiotic resistance. For that reason, the UNIPROT gene's annotation was further validated by their sequence identity to the CARD and ARDB databases. DIAMOND, a program that has similar performance to BLAST [117], but is much faster [40], was used for this purpose. For simplicity, UNI-gene is used here to denote a UNIPROT-derived gene, and CARD/ARDB-ARG is used to denote a gene derived from either CARD or ARDB (Figure 3.3). According to the sequence identity, each UNI-gene was classified into the following categories based on their potential to confer antibiotic resistance defined as an annotation factor:

1. High quality ARGs (High): A UNI-gene is tagged with a "High" annotation factor if it has ≥ 90% identity to a CARD/ARDB-ARG over its entire length. This similarity cutoff has been used in other studies to identify relevant ARGs [97, 118] and is stricter than that used in the construction of the ARDB database [107].
2. Homologous ARGs (Mid): A UNI-gene is tagged with a "Mid" annotation factor if it has ≥ 50% and ≤ 90% identity and an e-value lower than 1e-10 to a CARD/ARDB-ARG and also consistent annotation to the CARD/ARDB-ARG.
3. Potential ARGs (Manual Inspection): A UNI-gene is tagged with "Manual inspection" if it has < 50% identity and an e-value lower than 1e-10 to CARD/ARDB-ARGs and also consistent annotation to CARD/ARDB-ARGs. This gene is considered a potential ARG but with insufficient evidence and therefore warrants further analysis for the veracity of its antibiotic resistance.
4. Discarded ARGs (Low): A UNI-gene is discarded if its annotation differs from the best hit CARD/ARDB-ARG and the e-value is greater than 1e-10. Note the gene can potentially still be an ARG, but due to a lack of sufficient evidence, it is removed from current consideration to ensure ARG annotation quality.

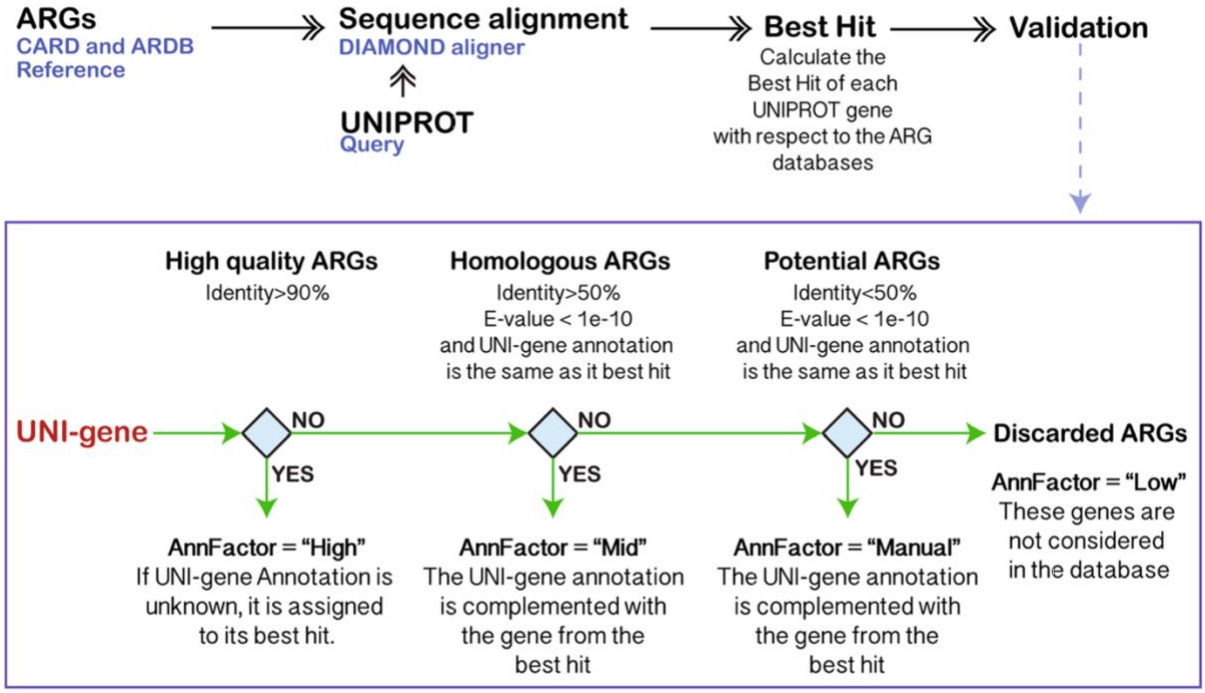

Figure 3.3: Validation of UNIPROT annotations. UNIPROT genes were aligned against the CARD and ARDB databases. The alignment with the highest bit score was selected for each UNI-gene (best hit) and a set of filters were applied to determine the UNI-gene annotation factor (AnnFactor).

Altogether 16,222 genes were tagged in the categories of "High" and "Mid" annotation factors. After removing sequences annotated as conferring resistance by single nucleotide polymorphisms (SNPs), a total of 10,602 UNIPROT, 2,203 CARD and 2,128 ARDB genes remained for downstream analysis. In total the DeepARG-DB comprises 14,933 genes including the three databases (CARD, ARDB, and UNIPROT). This database was used for the construction of the deep learning models.

3.2.4 Deep Learning

Supervised machine learning models are usually divided into characterization, training, and prediction units. The characterization unit is responsible for the representation of DNA sequences as numerical values called features. It requires a set of DNA descriptors that are based on global or local sequence properties. Here the concept of dissimilarity based classification [119] was used, where sequences were represented and featured by their identity distances to known ARGs. The CARD and ARDB genes were selected to represent known ARGs, whereas the UNIPROT (High+Mid) genes were used for training and validation of the models. DeepARG consists of two models: deepARG-LS, which was developed to classify ARGs based on full gene-length sequences, and DeepARG-SS, which was developed to identify and classify ARGs from short sequence reads (see Figure 3.4). The bit score was used as the similarity indicator, because it takes into account the extent of identity between sequences and, unlike the e-value, is independent of the database size [84]. The process for computing the dissimilarity representation was carried out as follows. The UNIPROT genes were aligned to the CARD and ARDB databases [106, 107] using DIAMOND [40] with very permissive constraints: 10,000 maximum number of hits representing the total number of reported hits to which a UNIPROT gene is aligned, a 20% minimum identity (--id 20), and an e-value smaller than 1e-10. The bit score was then normalized to the interval [0, 1] to represent the sequence similarity as a distance. Hence, scores close to 0 represent small distance or high similarity, and scores around 1 represent distant alignments. Thus, a feature matrix was built where the rows correspond to the sequence similarity of the UNIPROT genes to the features (ARDB/CARD genes).

A deep learning model, DeepARG, was subsequently created to annotate metagenomic sequences to antibiotic resistance categories. One of the main advantages of deep learning over other machine learning techniques is its ability to discriminate relevant features without the need for human intervention [120-122]. It has been highlighted for its ability to resolve multiclass classification problems [112, 123-126]. Here, a deep learning multiclass model was trained by taking into account the identity distance distribution of a sequence to all known ARGs. This distribution represents a high level of sequence abstraction propagated through a fully connected network. The DeepARG model consists of four dense hidden layers of 2000, 1000, 500, and 100 units that propagate the bit score distribution to dense and abstract features. The input layer consists of 4,333 units that correspond to the ARGs from ARDB and CARD. These features are used during training

and evaluation. To avoid overfitting, random hidden units were removed from the model at different rates using the dropout technique [127]. Lastly, the output layer of the deep neural network consists of 30 units that correspond to the antibiotic resistance categories. The output layer uses a softMax [128, 129] activation function that computes the probability of the input sequence against each ARG category. The probability is used to define the ARG category to which the input sequence belongs. The DeepARG architecture is implemented using the Python Lasagne [130] module, a high-level wrapper for the widely used Theano [131] deep learning library. Because deep learning demands intensive computational resources, the training was carried out using the GPU routines from Theano. However, heavy computation was required only once to train the deep learning model, and the prediction routines do not require such computational resources.

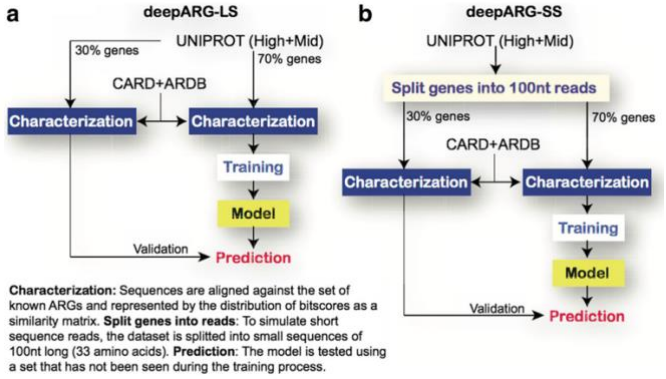

Figure 3.4: Classification framework. UNIPROT genes were used for validation and training whereas the CARD and ARDB databases were used as features. The distance between genes from UNIPROT to ARGs databases is computed using the sequence alignment bit score. Alignments are done using DIAMOND with permissive cutoffs allowing a high number of hits for each UNIPROT gene. This distribution is used to train and validate the deep learning models.

Two strategies have generally been used to identify ARGs from metagenomic data; one predicts ARGs directly using short reads, while the other uses predicted open reading frames (i.e., full gene-length sequences) from assembled contigs to predict ARGs. To allow for both annotation strategies, two deep learning models, DeepARG-SS and DeepARG-LS, were developed to process short reads and full gene length sequences, respectively. The DeepARG-SS model was designed specially to classify short reads generated by NGS technologies such as Illumina. Therefore, ARGs are split into small sequences to simulate short sequence reads (see Figure 3.4B). DeepARG-LS was trained using complete ARG sequences and can be used to annotate novel ARG genes (see Figure 3.4A), for instance, in open reading frames detected in assembled contigs from the MetaHit consortium [132]. Note that each model was trained and validated separately to ensure high performance.

3.3 Results and Discussion

To evaluate the performance of the DeepARG models (DeepARG-SS and DeepARG-LS), five different experiments were conducted and compared to the best hit approach. The prediction quality was evaluated by precision, recall, and F1-score metrics defined as,

$$Precision = \frac{TP}{TP+FP},$$

$$Recall = \frac{TP}{TP+FN},$$

$$F1\ score = 2 * \frac{precision*recall}{precision+recall},$$

where TP represents true positives (i.e., an ARG from the class of interest is predicted correctly as that ARG class), FP false positives (an ARG from a different class is predicted as from the class of interest), and FN false negatives (an ARG from the class of interest is predicted as a different ARG category).

Note because the first step of the DeepARG pipeline consists of sequence alignment using DIAMOND, nonARGs (short reads or full length genes) are filtered out and not considered for further prediction. Therefore, the alignment stage only passes ARG-like sequences that have e-value<1e-10 and identity>20% to DeepARG for prediction. Thus, the performance reflects the capability of the DeepARG models in differentiating the 30 antibiotic resistance categories.

3.3.1 Antibiotic Resistance Database

After the databases were merged and duplicates were removed, a total of 2,161, 2,290, and 28,108 genes were collected from the ARDB (50% of full ARDB), CARD (49% of all CARD genes), and UNIPROT (70% of total ARG-like sequences from UNIPROT) databases, respectively. For UNIPROT genes, a total of 16,222 genes were annotated using the available gene description. Following validation through sequence similarity and removing genes conferring resistance due to SNPs, 10,602 UNIPROT, 2,203 CARD, and 2,128 ARDB ARG-like sequences remained. The resulting database, DeepARG-DB, comprises 30 antibiotic categories, 2,149 groups, and 14,933 reference sequences (CARD+ARDB+UNIPROT). Over 34% of the genes belong to the beta lactamase category (5136), followed by 28% to the bacitracin category (4205), 7.4% to the macrolide-lincosamide-streptogramin (MLS) (1,109), 6.1% to the aminoglycoside (915), 5.8% to

the polymixin (879) and 5.8% to the multidrug (877) classes (see Figure 3.5A). The categories where the UNIPROT database made the greatest contribution correspond to beta-lactam, bacitracin, MLS, and polymyxin. However, not all ARG categories were found in the UNIPROT database, such as elfamycin, fusidic acid, and puromycin, among others (see Figure 3.5B for details). One of the limitations of DeepARG-DB is its dependency on the quality of the CARD and ARDB databases. Thus, to avoid the propagation of errors from the CARD and ARDB, gene categories and groups were manually inspected and corrected. In particular, the ARGs with conflicting annotations from ARDB and CARD databases. Because UNIPROT and CARD are continuously updated, the DeepARG-DB will likewise be updated and versioned accordingly as the trained deep learning models.

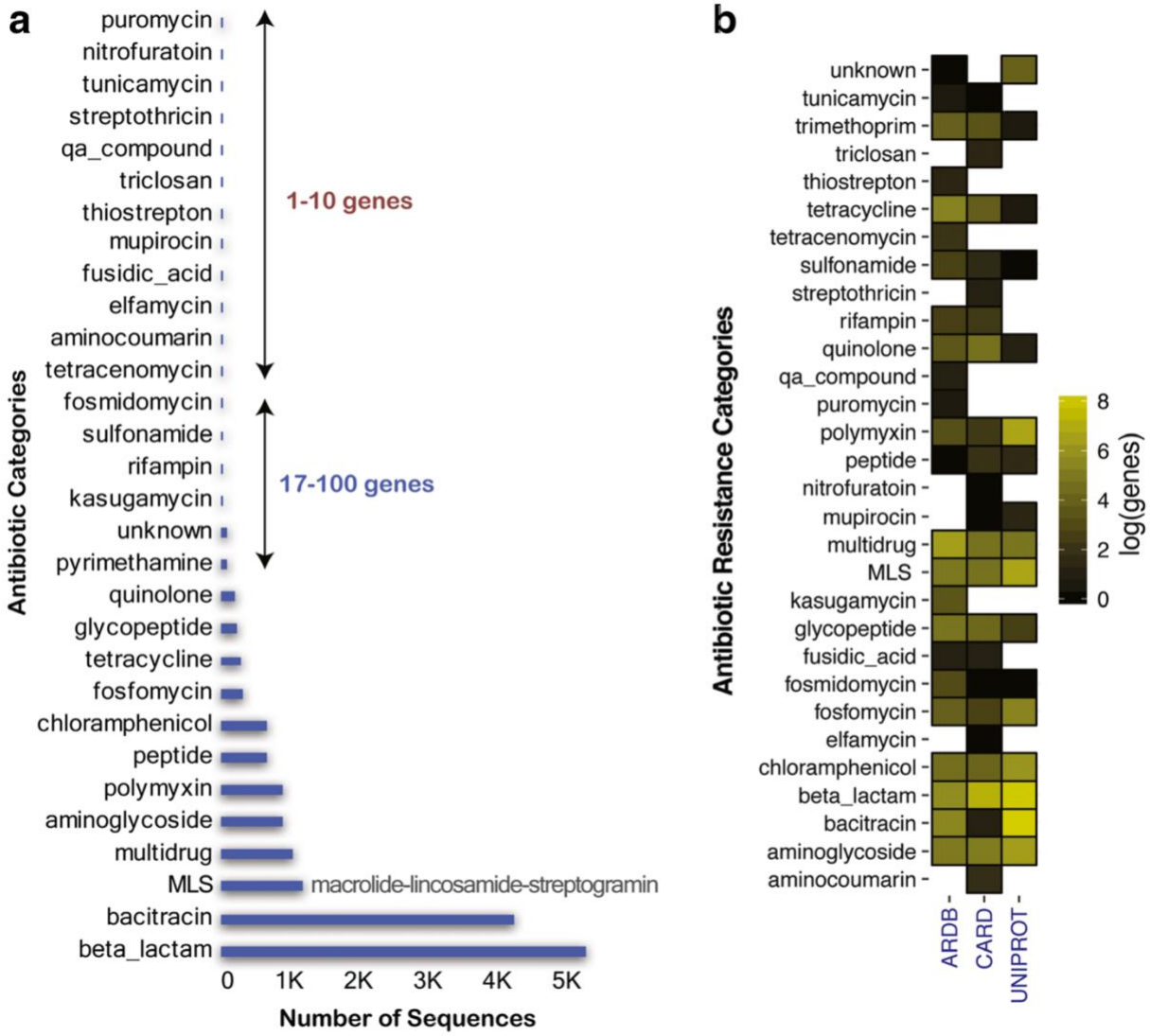

Figure 3.5: A) Distribution of the number of sequences in the 30 antibiotic categories in DeepARG-DB. B) The relative contribution of ARG categories in the ARDB, CARD, and UNIPROT databases.

3.3.2 Prediction of Short Sequence Reads

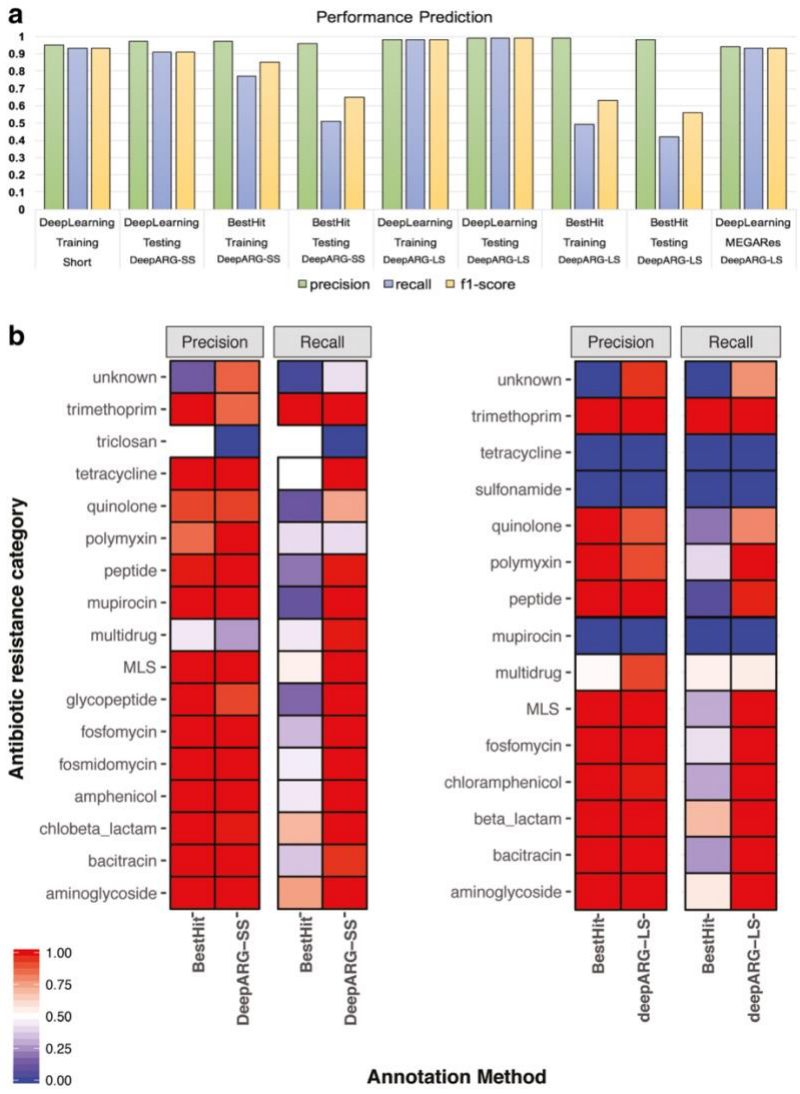

Figure 3.6: A) Performance comparison of the DeepARG models with the best hit approach using precision, recall, and F1-score as metrics for the training and testing datasets. The MEGARes bars corresponds to the performance of DeepARG-LS using the genes from the MEGARes database. B) Precision and recall of DeepARG models against the best hit approach for each individual category in the testing dataset. *UNIPROT genes are used for testing and not all the ARG categories have genes from the UNIPROT database.

To simulate a typical metagenomic library, UNIPROT genes were split into 100 nucleotide long sequences, with a total of 321,008 reads generated. The DeepARG-SS model was subsequently trained and tested in a manner in which 70% of the reads were randomly selected for training,

while the remaining 30% were reserved for validation. An overall precision of 0.97 and a recall of 0.91 were achieved among the 30 antibiotic categories tested (see Figure 3.6A). In comparison, the best hit approach achieved an overall 0.96 precision and 0.51 recall. Achieving high precision for the best hit approach is not surprising, as the method relies on high identity constraints and has been reported to predict a low number of false positives, but a high number of false negatives [104]. We observed that both methods yielded high precision for most of the categories (see Figure 3.6B). However, both methods performed poorly for the triclosan category, likely because the category was only represented by four genes in the database.

The DeepARG-SS model performed particularly well for antibiotic resistance categories that were well-populated, such as beta lactamases, bacitracin, and MLS, but not as well for categories represented by a small number of ARGs, such as triclosan and aminocoumarin. This result is expected due to the nature of neural network models. As more data becomes available to train the models, the better their ultimate performance. In contrast, the best hit approach yielded perfect prediction for some ARG categories containing a limited number of ARGs, but not for categories with a large number of ARGs (see Figure 3.6B).

For the multidrug antibiotic resistance category, the DeepARG-SS model had an almost perfect recall (0.99), implying that only a small number of multidrug reads were classified to other categories. However, the DeepARG-SS model also had the highest false positive rate compared to other categories (precision 0.27), implying that many non-multidrug reads were annotated as multidrug sequences. On the other hand, the best hit approach showed a higher precision (0.44), but a much lower recall (0.44). The multidrug category contains genes that confer resistance to multiple antibiotic categories such as macrolides, beta-lactamases, glycopeptides, quinolones, as well as other antimicrobials such as metals [133, 134]. These genes often share similar sequences, which makes it challenging for computational methods to determine the true identity of a short read. Therefore, when reads yield a best prediction probability less than 0.9, DeepARG reports the top two ARG categories for manual inspection. The low precision seen in both methods suggests that other non-multidrug categories may contain genes that have high sequence similarity to the multidrug category. This illustrates that there is still much room for improvement in existing databases.

Contrary to the multidrug category, the “unknown” antibiotic resistance category has a high precision of 0.87, but a low recall of 0.42, indicating a high false negative rate. Thus, reads from the unknown antibiotic resistance category can be mistakenly predicted as another antibiotic resistance category. This highlights the need to check whether the unknown category actually contains genes from other ARG categories such as beta-lactam, macrolides, or triclosan, among others. Comparatively, the best hit approach has the worst performance for the “unknown” antibiotic category (see Figure 3.6B). In general, the DeepARG-SS model demonstrated

significant improvement in the false negative rate compared to the best hit approach for nearly all ARG categories.

3.3.3 Prediction of Long ARG-like Sequences

The DeepARG-LS model was trained and tested using full gene-length sequences. The UNIPROT validated genes were split into a training set (70% of the data) and a validation set (30% of the data) with the CARD and ARDB databases were used as features. The DeepARG-LS model shows similar results, with an overall precision of 0.99 and recall of 0.99 for predicting different categories of ARGs. Better performance in DeepARG-LS than DeepARG-SS is expected, because longer sequences contain more information than short reads (Figure 3.6). Particularly DeepARG-LS achieved a high precision ($\mathbf{0.97 \pm 0.03}$) and an almost perfect recall ($\mathbf{0.99 \pm 0.01}$) for the antibiotic categories that were highly represented in the database, such as bacitracin, beta lactamase, chloramphenicol, and aminoglycoside (See Figure 3.6B). Comparatively, the best hit approach achieved a perfect precision ($\mathbf{1.00 \pm 0.00}$) but a much lower recall ($\mathbf{0.48 \pm 0.2}$) for these categories. Similar to DeepARG-SS, DeepARG-LS did not perform well for categories with few genes, such as sulfonamide and mupirocin.

3.3.4 Performance Prediction of Known and Validated ARGs

To further evaluate and validate performance, the DeepARG-LS model was applied to all of the ARG sequences in the MEGARes database [23]. This database contains manually curated ARGs from CARD [106], ARG-ANNOT [26], and RESFINDER [99]. ARGs conferring resistance by mechanisms that result from SNPs are removed in this test. Comparison of the DeepARG-LS prediction with the database annotation yielded an overall precision and recall of 0.94 and 0.93, respectively (Figure 3.6A). The DeepARG-LS model achieved an almost perfect precision of 0.99 ± 0.05 and recall of 0.96 ± 0.03 for categories with a large number of genes, such as beta lactamases, elfamycin, fosfomycin, glycopeptides, MLS, and sulfonamide. However, the model performed poorly for categories that had a small number of genes. For instance, MEGARes has a Tunicamycin gene that was assigned by the DeepARG-LS model as quinolone with a probability of 0.6. Such a low probability 0.6 suggests that the gene has more than one annotation. When the complete annotation for this gene was manually inspected, it was found that the DeepARG-LS model predicted the correct label (Tunicamycin) with a 0.3 probability, indicating that for this particular category more gene sequences are required to train the model. The DeepARG-DB database has only three Tunicamycin genes, which may explain why this gene was not properly classified. However, it is worth noting that the thiostrepton category was predicted correctly despite its lower number of training genes. The multidrug category is one of the most difficult categories to predict, containing about 200 genes. For the multidrug category, the DeepARG-LS model yielded a 0.7 precision with a 0.6 recall. This result suggests the need to manually inspect the genes tagged as multidrug as well as the genes from other categories that were assigned to the multidrug category. Challenges annotating genes belonging to the multidrug category further

highlights the broader need to review, compare, and seek consensus among different antibiotic resistance databases.

3.3.5 Validation through *Novel* ARGs

To test the ability of the DeepARG-LS model to predict novel ARGs, a set of 76 metallo beta lactamase genes were obtained from an independent study by Berglund et al. [135]. These novel genes have been experimentally-validated via a functional metagenomics approach to confer resistance to carbapenem in E. *coli*. In the study, a large scale analysis was carried out by screening thousands of metagenomes and bacterial genomes to a curated set of beta lactamases. Using a hidden Markov model trained and optimized over a set of beta lactamases, 76 beta lactamase candidate novel genes were collected. Experimental validation was performed and 18 out of the 21 tested genes were able to hydrolase imipenem. Therefore, these 76 beta lactamase genes are expected to be mostly true ARGs and provide a unique opportunity to further test and validate the DeepARG-LS model. Interestingly, out of the 76 novel ARGs, the DeepARG-LS model was able to predict 65 (85% accuracy assuming all 76 are real ARGs) as the correct antibiotic category of beta lactamase with a probability greater than 0.99. The remaining nine genes were also predicted correctly by the DeepARG-LS model but were filtered out because of their low alignment coverage (i.e., <50%; alignment-length/ARG-length). Important to note is that the DeepARG-LS model was trained across 30 antibiotic categories and was not optimized to detect any one particular antibiotic category. Therefore, this result strongly demonstrates the capability of the DeepARG-LS model to detect novel ARGs. Of course, one possibility for the high accuracy of the DeepARG prediction is that these 76 genes or their closely related genes were included in training the DeepARG-LS model. To examine this possibility, the 76 beta lactamase genes were compared against all the sequences in DeepARG-DB using DIAMOND [40] and the best hit for each gene was extracted. Figure 3.7A shows that, surprisingly, all of the best hits identified in DeepARG-DB had less than 40% sequence similarity to the 76 beta lactamases, indicating that the high accuracy of the DeepARG prediction is not due to inclusion of these genes and/or their close related genes in training the DeepARG-LS model. In fact, Figure 3.7B shows the pairwise identity distribution of the beta lactamase genes used in training. Most of the beta lactamase genes are very similar to each other with pairwise identities greater than 90%, and only a small number of them having low pairwise identity values. Taken together, these analyses show that using a diverse set of beta lactamase genes for training, the DeepARG-LS model was able to learn the specificities of distantly related genes and consequently detect them. Thus, the DeepARG-LS model shows promise for the identification of novel ARGs. In contrast, the common practice of using the best hit approach with a universal 50% (or higher) identity cutoff [136] will fail to detect all these novel ARGs. Note that the length requirement imposed by DeepARG can be relaxed and adjusted depending on the specific research question. For example, if identifying as many potential novel ARGs as possible is the main focus, one can use a more relaxed length constraint than DeepARG's default.

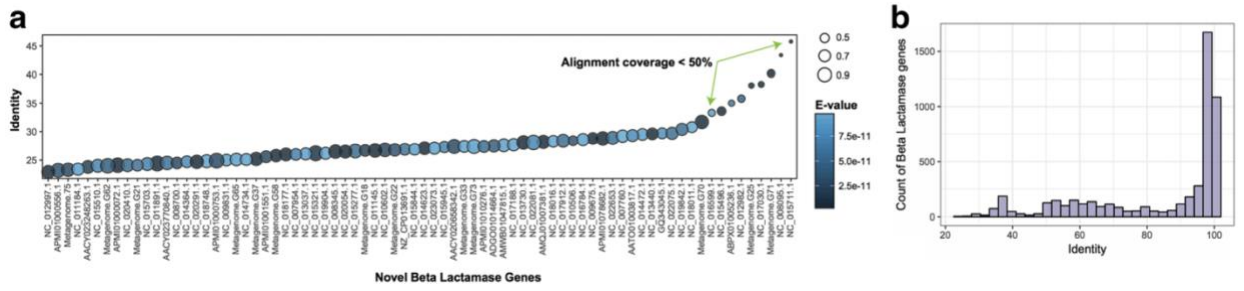

Figure 3.7: A) Identity distribution of 76 novel beta lactamase genes against the DeepARG database (DeepARG-DB). Each dot corresponds to the best hit of each novel gene where color indicates the E-value (<1e-10) and size depicts the alignment coverage (>40%). B) Pairwise identity distribution of the beta lactamase genes in the DeepARG database.

3.3.6 Validation through an in *Silico* Spike-in Experiment

For metagenomic data sets derived from real-world samples, ARG reads may account for only a small fraction of the total reads. Thus, it is important to examine how the DeepARG-SS model performs in situations where non-target genes are dominant. In order to measure the ability of the DeepARG-SS model to discriminate and identify a small number of ARG reads among a large majority of nonARG reads, a negative metagenomic data set was constructed that mimics a spike-in metagenomic experiment. First, a set of 6,485,966 reads of 100 bps were extracted from several eukaryote genomes (*Homo sapiens, Mus musculus, and Acanthisitta cholirs*) to generate the majority of nonARG reads (since eukaryote genomes are expected to have few ARG-like sequences). Second, a positive set of ARG reads was built by screening known ARGs against the bacterial genomes from the PATRIC database [137]. Only regions with an identity between 70% to 90% over the entire gene with an e-value below 1e-10 were used, and 10,000 short reads of 100 bps were extracted randomly from these regions to form the small set of ARG reads.

Figure 3.8 shows the prediction result of DeepARG-SS for the 10,000 non-dominant ARG reads. Only one nonARG read was predicted to be a ARG read with an identity of 78%, while the remaining nonARG reads were discarded during the sequence alignment step due to failure to meet the requirement for a minimum of 20% sequence identity to at least one of the 4,333 feature ARGs imposed by deepARG. Thus, even though the data set contains largely nonARG reads, the DeepARG-SS model was able to identify and predict the small number of ARG reads with high sensitivity. For example, using the default prediction probability cutoff of 0.8, the number of true positives (the ARG reads that were predicted to the correct antibiotic categories) is 9976, while the number of false negatives (the ARG reads that were predicted to the wrong antibiotic categories) was 24, yielding a 0.99 (9976/10000) sensitivity. These results show that, first, the alignment step in DeepARG acts as a filter that can effectively remove nonARG sequences, and second, despite the weak signal, DeepARG-SS predicts ARG reads correctly and with high sensitivity. Note that, despite the ARG-like regions having 70% to 90% sequence identities to the known ARGs, the

extracted reads have a much wider range of sequence identity of 50% to 100% to the ARGs due to different degrees of sequence conservation and diversity along the entire sequences of the ARGs (Figure 3.8).

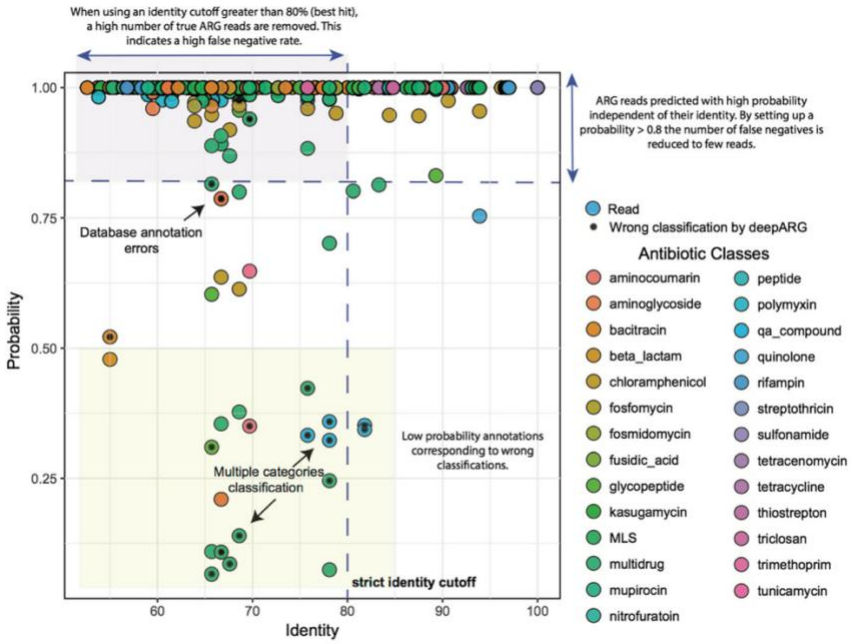

Figure 3.8: Prediction result using the DeepARG-SS model to classify ARGs for the spike-in data set. Results for nonARG reads (eukaryotic reads) are not shown because DeepARG-SS was able to remove them during the alignment step using DIAMOND.

In practice, the annotation of short reads is often performed with the best hit approach. For this strategy, an identity cutoff between 80 and 90% to known ARGs is widely accepted as it has a low false positive rate [136]. When using the 80% cutoff, the best hit method yielded 4486 false negatives and 5514 true positives, thus a much lower sensitivity (0.55) than DeepARG. As expected, the best hit approach with these cutoffs can lead to underestimation or even erroneous inference of ARG contents in metagenomic data sets. Comparatively, the DeepARG-SS model aims to identify as many true positives as possible and, at the same time, to minimize the number of false negatives. To achieve this, the DeepARG-SS model examines the distribution of all the hits instead of relying on the best hit solely. As a result, the DeepARG-SS model was able to identify the correct antibiotic category and more importantly, to minimize the misclassification errors by providing a classification probability for each prediction. Our empirical analysis showed that this likelihood is an important metric to consider when one uses DeepARG for prediction. For instance, most of the classifications that have low prediction probabilities (<0.5) are wrong and correspond to reads commonly found in different ARG categories, whereas only two erroneous predictions were observed for classification with high probabilities (>0.8). Therefore, a probability cutoff of 0.8 is recommended when performing the classification. In addition, the DeepARG

probability is independent of the sequence identity, which means that even with low sequence identities, the likelihood of obtaining the correct classification can still be high.

Still, it is important to clarify that, despite the low false negative and false positive rate of this evaluation, the performance of the DeepARG models is dependent on the quality of the training database. As illustrated in Figure 3.8, there are four incorrect classifications that have > 0.75 probability. These errors are likely generated by erroneous labels in the database. Hence, continued curation and/or validation of ARGs is crucial for improving the accuracy of ARGs predictions.

Also observed were several incorrect classifications with prediction probability < 0.5. The low probability for these reads suggests that they are predicted to multiple antibiotic categories. As a result, the probability is shared among different antibiotic categories. To avoid such errors, DeepARG uses a 0.8 minimum probability cutoff (as default) that can be modified by users. DeepARG also enables the adjustment of the identity cutoff used during the alignment stage. These parameters allow users to produce more or less stringent classification according to their needs.

3.3.7 Validation through PseudoARGs

To further examine the ability of DeepARG to discriminate genes that may contain segments of ARGs but are not true ARGs (i.e., pseudoARGs), a set of pseudoARGs were created. These genes were constructed by randomly picking k-mers from different ARG categories as follows: To build one gene, five k-mers of 50 amino acids long were randomly selected from one specific ARG category. Then, two 50-mers were randomly selected from ten more ARG categories. Finally, this process was repeated to build 300 genes with partial ARG content. This false positive data set mimics the cases where genes from different categories share similarities within their sequences, e.g., the same domains or motifs. The pseudoARG data set was then classified using the DeepARG-LS model and the best hit approach. As expected, the best hit approach was not able to filter out the false positive ARGs and produced a high false positive rate of 57% with the identity cutoff of 50% (Figure 3.9), while using lower cutoffs would increase the number of false positives even more. In contrast, using the default classification probability cutoff of 0.8, the DeepARG-LS model was able to filter out 285 of the 300 pseudoARGs (5% false positive rate). This shows the superiority of the DeepARG-LS model in distinguishing pseudoARGs over the best hit approach, further supporting that the DeepARG model learns the uniqueness of the ARG categories through taking into account the similarities of the target sequence to all the ARG categories.

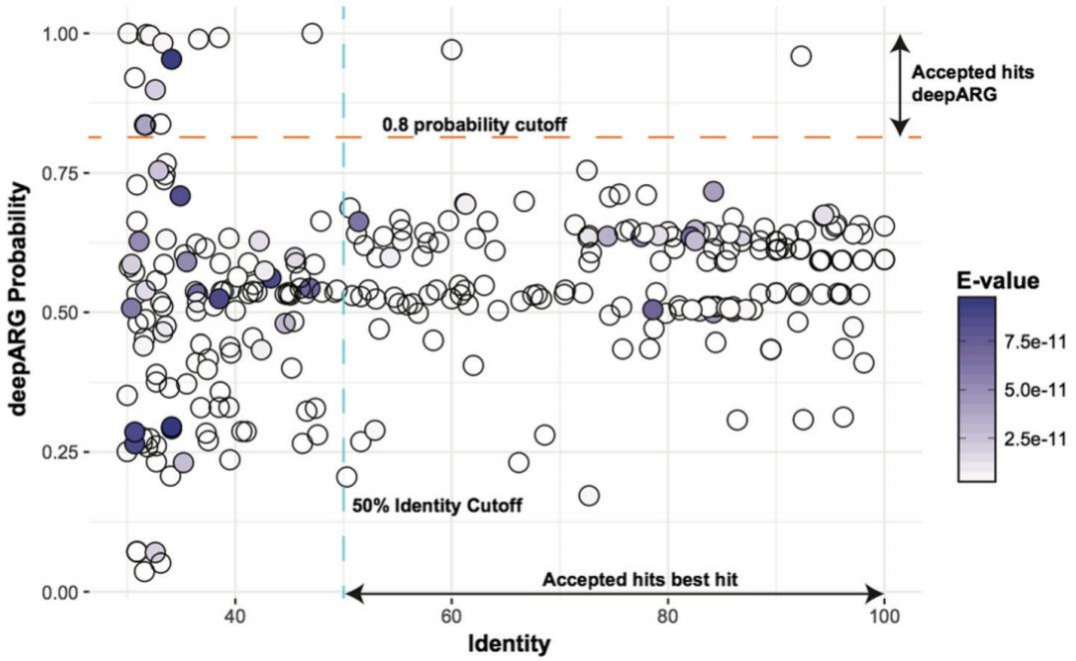

Figure 3.9: Distribution of DeepARG classification probability and the best hit identity. Each point indicates the alignment of each "partial" negative ARG against the DeepARG database. The horizontal line indicates the default setting for DeepARG predictions, i.e., the predictions with a probability higher than 0.8 are considered by DeepARG as high-quality classifications.

3.3.8 Limitations of DeepARG and Usage Recommendation

The two DeepARG models, DeepARG-LS and DeepARG-SS, are tailored to different ARG prediction strategies. For example, it is now a common practice for researchers to collect different environmental samples, sequence the DNA to obtain metagenomic data, and use the data to address the question "what kinds of ARGs are present in the samples?". In this case, with the metagenomic data, one can simply predict the ARG categories where the reads belong to by using the DeepARG-SS model directly to the reads, similar to what was done for the *in silico* spike-in metagenomic experiment. This task can be done rapidly as experiments demonstrated that predicting 100 million short reads required only 50 minutes on a personal MacBook pro with i7 processor and 16Gb of RAM. As pointed out previously, training the DeepARG model is very time consuming, but is only done once. Alternatively, one can first assemble the short reads into contigs, obtain open reading frames (ORFs) using an ORF identification/prediction program for the contigs, and then run the DeepARG-LS model on the ORFs to predict ARG categories. Comparatively, the latter strategy can be much slower as it involves sequence assembly, but the prediction might be more accurate than direct prediction on reads. This is expected as the longer the sequences are, the more information contained, and therefore the more confidence one has in the ARG prediction. This is also clear from the results where the DeepARG-LS model performed better than the DeepARG-SS model (Figure 3.6). In cases where full gene length sequences are readily obtainable such as

the 76 novel beta lactamase genes, DeepARG-LS can be deployed to predict the corresponding ARG categories.

Several points are worthy of discussion. First, the DeepARG models were trained across 30 ARG categories and are intended to predict which of these categories a gene or short read belongs to. It is not intended and cannot be used to predict antibiotic resistance that arises from SNPs. Second, the DeepARG models can only predict whether a gene or read belongs to one of the 30 categories that are considered by the model. If the gene or read belongs to an entirely new ARG category, DeepARG will not be able to predict it. In such a case, it is worth noting that prediction probabilities for the 30 categories are expectedly low and one should treat the predictions with caution and may discard the prediction if a high quality set of ARG predictions is desired. Third, the performance of the DeepARG models hinges on the quality of the training database, i.e., the higher quality the training data, the higher prediction accuracy the model. Detailed analyses of the prediction results suggest that some of the ARG categories may have annotation errors, especially the multidrug and "unknown" categories, which in turn adversely affects the prediction of the models. This highlights the importance of continued and synergistic effort from the research community in curating and improving ARG nomenclature and annotation databases. Fourth, as with *all in silico* prediction, the DeepARG models can be used to get an overview or inference of the kinds of antibiotic resistance in a collection of sequences; strictly speaking, downstream experimental validation is required to confirm whether the sequences truly confer resistance.

3.4 Conclusions

Here, a new computational resource for the identification and annotation of ARGs derived from metagenomic data is developed, trained, and evaluated. The deep learning approach proved to be more accurate than the widely used best hit approach and is not restricted to strict cutoffs, thus greatly reducing false negatives and offering a powerful approach for metagenomic profiling of ARGs in environmental compartments. Further, the DeepARG database developed here greatly expands the available ARGs individually available in the currently most widely used CARD, ARDB, and UNIPROT databases, including their existing sequence content and extensive metadata. DeepARG provides a publicly-available database structured into a simple category and group hierarchy for each ARG. While DeepARG is not intended to replace CARD or ARDB, in conjunction with deep learning, it aims to improve ARG annotation by drastically reducing the false negative rate, while maintaining a similarly high true positive rate associated with the traditional best hit approach. The performance of DeepARG highly depends on the quality of the training database. Therefore, the inclusion of new entries based on alignment similarity could integrate genes that have not been validated to produce antibiotic resistance *in vivo*. However, this

in silico gene integration is useful to expand the diversity of ARGs, as it is shown by the analysis of novel ARGs where distant genes have been predicted to the correct antibiotic resistance category.

The source code for the DeepARG models can be downloaded from a git repository (https://bitbucket.org/gusphdproj/deeparg-ss). It consists of a command line program where the input can be either a FASTA file or a BLAST tabular file. If the input is a FASTA sequence file, DeepARG will perform the sequence search first and then annotate ARGs. If the input is already a BLAST tabular file, DeepARG will annotate ARGs directly. An online version of deepARG is also available where a user can upload a metagenomics raw sequence files (FASTQ format) for ARG annotation (http://bench.cs.vt.edu/deeparg). Once the data is processed, the user receives an email with results of annotated ARGs with the absolute abundance of the ARGs and the relative abundance of ARGs normalized to the 16S rRNA content in the sample as used in [19, 20]. This normalization is useful to compare the ARG content from different samples. The web service also allows users to modify the parameters (identity, probability, coverage and E-value) of the DeepARG analysis. With the command line version, the user also has access to more elaborate results such as the probabilities of each read or gene belonging to the specific antibiotic resistance categories. In addition to prediction of antibiotic categories and the associated probabilities, the DeepARG model reports the entries with multiple classifications. In detail, if a read or complete gene sequence is classified to an antibiotic category with a probability below 0.9, the top two classifications will be provided. This would help researchers identify reads or sequences with less confident predictions, and it is recommended that the detailed output be examined together with domain knowledge to determine the more likely ARG category. The DeepARG-DB is freely available under the DeepARG Web site (http://bench.cs.vt.edu/deeparg) as a protein FASTA file, and it is included in the git repository. Each entry in the database has a complete description that includes the gene identifier, the database where the gene is coming from, the antibiotic category, and the antibiotic group. For users interested on a particular set of genes, DeepARG also provides the steps to create a new deep learning model using the architecture of DeepARG. This architecture is not restricted to ARGs and can be used to train any set of genes.

CHAPTER 4 CURATION OF ANTIBIOTIC RESISTANCE GENES

Curation of antibiotic resistance gene (ARG) databases is labor intensive and requires expert knowledge to manually collect, correct, and annotate individual genes. Consequently, most existing ARG databases contain only a small number of ARGs (~5k genes) and updates to these databases tend to be infrequent, commonly requiring years for completion and often containing inconsistencies. Thus a new approach is needed to achieve a truly comprehensive ARG database while also maintaining a high level of accuracy. Here we propose a new web-based curation system, ARGminer, that supports the annotation and inspection of several key attributes of potential ARGs, including gene name, antibiotic category, resistance mechanism, evidence for mobility, and occurrence in clinically-important bacterial strains. Here, we employ crowdsourcing as a novel strategy to overcome limitations of manual curation and expand curation capacity towards achieving a truly comprehensive and perpetually up-to-date database. Further, machine learning is employed as a powerful means to validate database curation, drawing from natural language processing to infer correct and consistent nomenclature for each potential ARG. We develop and validate the crowdsourcing approach by comparing performances of multiple cohorts of curators with varying levels of expertise, demonstrating that ARGminer is a time and cost efficient means of achieving accurate ARG curation. We further demonstrate the reliability of a trust validation filter for rejecting input generated by spammers. Crowdsourcing was found to be as accurate as expert annotation, with an accuracy >90% for the annotation of a diverse test set of ARGs. The ARGminer public search platform and database is available at http://bench.cs.vt.edu/argminer.

4.1 Introduction

Antimicrobial resistance (AMR) has been identified by the World Health Organization (WHO) as a major global health threat. It is projected that AMR will increase exponentially by 2050, leading to substantial human morbidity and mortality [138, 139]. Therefore, swift action is required to enable enhanced monitoring and help tackle the spread of AMR, including: understanding the mechanisms controlling dissemination of antibiotic resistance genes (ARGs) via environmental sources and pathways [94, 136, 140], discovering novel ARGs before they are found to be problematic in the clinic [135], developing new computational strategies for ARG annotation [20, 22, 23, 25, 141], and expansion of current ARG repositories [23, 25].

Metagenomic sequencing has provided a powerful means for accessing the diverse array of ARGs, or "resistomes," [142] characteristic of various environments [91, 143-145] and has supported the discovery of novel ARGs and their interactions [29, 92]. Existing metagenomic approaches are largely dependent upon predicting antibiotic resistance attributes through sequence similarity computation, which is subject to major limitations. First, such similarity computations require a high quality and up-to-date ARG reference and annotation database to enable consistent and

accurate ARG identification. Second, the scope of such analyses is limited to previously characterized ARGs, either due to the parameter cutoff stringency employed in the sequence alignment or to lack of a comprehensive target gene for alignment [20].

To improve the capacity of metagenomic-based approaches to broadly and accurately detect the full range of ARGs present in a given sample, it is necessary to continuously expand and improve curation of corresponding databases [25]. However, risk of incorporation of false positives, i.e., "ARG-like" genes that do not necessarily induce an AMR phenotype, stands as a major impediment to expanded curation efforts. Therefore, manual inspection and validation of potential ARG entries is a critical aspect of ensuring the validity of AMR databases and their application.

Manual curation of ARGs is typically carried out by a few experts associated with research groups committed to maintaining public databases. This process is complex, tedious, and time-consuming. For instance, the last update of the Antibiotic Resistance Database (ARDB) was in 2009 [24], and, therefore, it does not contain newly discovered ARGs, such as *bla*$_{\text{NDM-1}}$ or *mcr*-1. The MEGARes database [23], which was designed to simplify the organization of ARG annotation, has not been updated since December, 2016. The resqu database, which contains genes for which there is evidence of having been transferred via Mobile Genetic elements (MGEs), has not been updated since 2013 [21]. The Comprehensive Antibiotic Resistance Database (CARD) [22] is widely considered to be the most up-to-date ARG resource. First introduced in 2016, CARD has been updated more than 21 times, with corresponding changes to the ARG sequences and metadata (e.g., antibiotic class, gene name, and mechanism). This acutely illustrates how complex and time-consuming ARG database curation is, even for domain experts.
Attempts have been made to address limitations of currently available databases, including introduction of new databases, such as the structured antibiotic resistance database (SARG), which employed intense manual curation to address issues such as inconsistencies in nomenclature and elimination of single-nucleotide polymorphisms and housekeeping genes [20]. In our own research group, we previously introduced DeepARG, a computational approach for predicting of ARGs using deep learning [25]. Along with the machine learning models, we also released a curated database named DeepARG-DB. This database employs manual curation, literature review of ARGs, and annotation of ARGs using sequence alignments. DeepARG-DB was first released in July, 2017, and most recently updated August, 2018. However, the DeepARG database depends on annotations from multiple resources, making it sensitive to the propagation of errors from other databases. This highlights the need for a specialized tool that brings all the ARG information from different resources to easily integrate new ARGs or to validate the annotations of current ARGs.

To overcome the difficulties in curation and manual validation of an extensive number of ARGs, a novel approach that breaks down this complex task into simpler and smaller microtasks is proposed. The core of this approach consists of aggregating a compendium of AMR resources and deploying a crowdsourcing strategy, which simplifies the ARG information to allow nonexperts,

i.e., the general public, and domain experts collectively to curate the ARG database. Application of crowdsourcing in biology, particularly for data curation, is not new and comprises a variety of areas including: name entity recognition (NER) for drug and diseases [146-148], identification of medically-relevant terms from patient online posts [149], annotation of diseases described in PubMed [150], and systematic examination of databases and other resources for drug indications, biomedical ontologies, and gene-disease interactions [147, 151-153]. Interestingly, in most of the studies, crowdsourcing has proven to be as effective as expert curation [147, 154].

A major problem that encompasses all ARG resources is the lack of a standardized gene nomenclature. In particular, the naming of ARGs does not follow the general nomenclature for naming bacterial genes [155]. For instance, macrolide resistance genes are structured so that the class is indicated in parenthesis (e.g., *ole*(B), *srm*(B), *vga*(B) or *ere*(B)) [156]. When compared to tetracycline genes, this gene nomenclature differs radically, because, in tetracycline genes, the determinant is placed as a capital letter after the gene name (e.g., tetA, tetB, tetC) [156]. At the same time, those nomenclatures differ from the gene convention proposed to annotate beta lactamases genes [157]. Other examples to highlight these differences include the aminoglycoside gene [158] *aad*A1 found under different names across the available ARG databases (ANT(3'')-I, aadA1-pm, ANT3-DPRIME, and ant3ia). Therefore, the diversity and variation in the ARG nomenclature and naming conventions complicate and greatly hinder consistent ARG curation.

Here, we introduce ARGminer, an online platform to enhance manual curation of ARGs. ARGminer enables users to curate and retrieve all the information available from several ARG resources, including CARD [22], DeepARG-DB [25], ARDB [24], MEGARes [23], UniProt [159], the National Database of Antibiotic Resistant Organisms (NDARO) (https://www.ncbi.nlm.nih.gov/pathogens/antimicrobial-resistance/), the structured antibiotic resistance (SARG) database [20], ResFinder [99], and the ARGANNOT [26] database. Manual crowd-source-based curation is enhanced by a machine learning model based on word embeddings [160, 161], a technique widely used in natural language processing (NLP), to aid in validation and achieve consistency in ARG nomenclature. On the other hand, mobile genetic elements (MGEs), such as plasmids, phages, and viruses, play an important role in the dissemination of ARGs [136, 159, 162]. Therefore, ARGminer also interfaces with the PATRIC [137] and Classification of Mobile Genetic Elements (ACLAME) [159] databases, which provide information on potential carriage of ARGs by pathogens or MGEs, respectively. The ARGminer platform is designed, built, and implemented as an open-source project facilitating a collaborative and integrative approach for the standardization of ARG annotation by the broad community of scientists and citizens motivated by a common desire to contribute towards combating the spread of AMR. ARGminer also includes a community blog for users to post questions, share solutions and participate in discussion regarding antimicrobial resistance with the objective to keep the scientific community actively engaged in the latest updates and development of ARG databases (see **Figure 4.1**). All

data associated with ARGminer, as well as the source code, is freely available under a public repository at http://bench.cs.vt.edu/argminer.

4.2 Materials and Methods

4.2.1 ARG Database

ARGs were downloaded from the following resources: CARD [22], which contains ARG information; the ARDB [24] database, which comprises a vast number of homology-predicted ARGs; DeepARG-DB [25], which integrates ARGs from UniProt [72], CARD, and ARDB; MEGARes [23] database, which incorporates genes from the ARG-ANNOT [26], ResFinder [99], the Lahey Clinic beta-lactamase archive [163] available from the National Center for Biotechnology Information (NCBI), the SARG database [20], and the NDARO database version 2 (https://www.ncbi.nlm.nih.gov/pathogens/antimicrobial-resistance/).

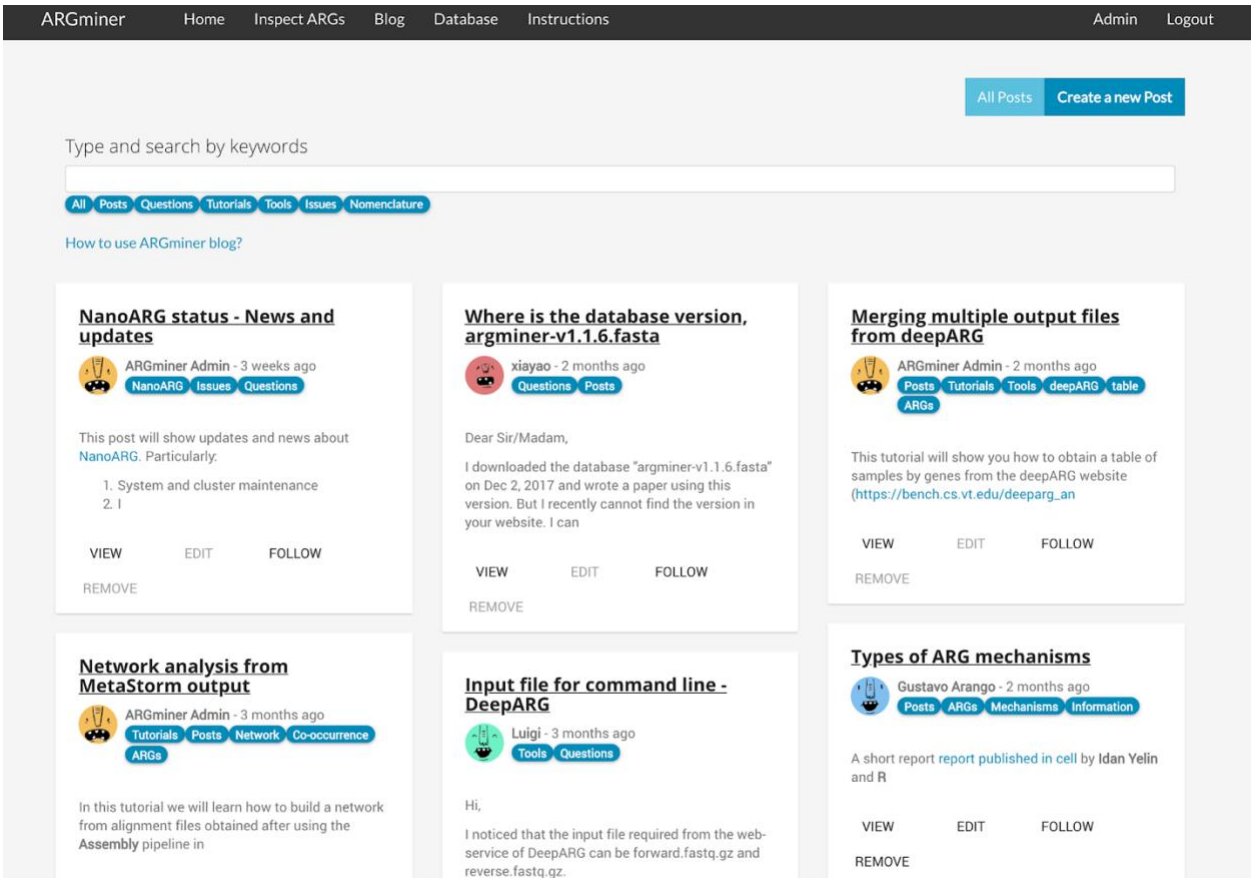

Figure 4.1: ARGminer blog available for users to upload questions, posts, tutorials about analysis of ARGs or general nomenclature questions.

To obtain a clean collection of ARGs, the DeepARG-DB database was updated with a more recent version of the CARD (v 2.0.4) and UniProt databases using their corresponding sequence identifiers. Discontinued UniProt sequences were removed from DeepARG-DB, whereas the newly-added ARGs from CARD were incorporated. Also, genes from CARD known to confer resistance due to single point mutations were removed. All sequences from all databases were clustered to remove duplicates by using cd-hit and identity cutoff of 100%. The resulting collection of ARGs was then aligned to all databases using DIAMOND [40] and TBLASTN [164] to extract the best hit of each ARG along with its corresponding metadata. In this manner, each ARG is represented by its best hit to each database, upholding consistency in annotation among the ARG resources. The metadata from the UniProt database is accessed via the UniProt API (Application Programming Interface), which allows retrieval of up-to-date information for each gene. Therefore, each ARG is displayed in the user interface as a set of sections containing an ARG's best hits, its metadata, and the alignment quality. Scores are shown as bars to enhance readability and curator interpretation (see **Figure 4.2A)**.

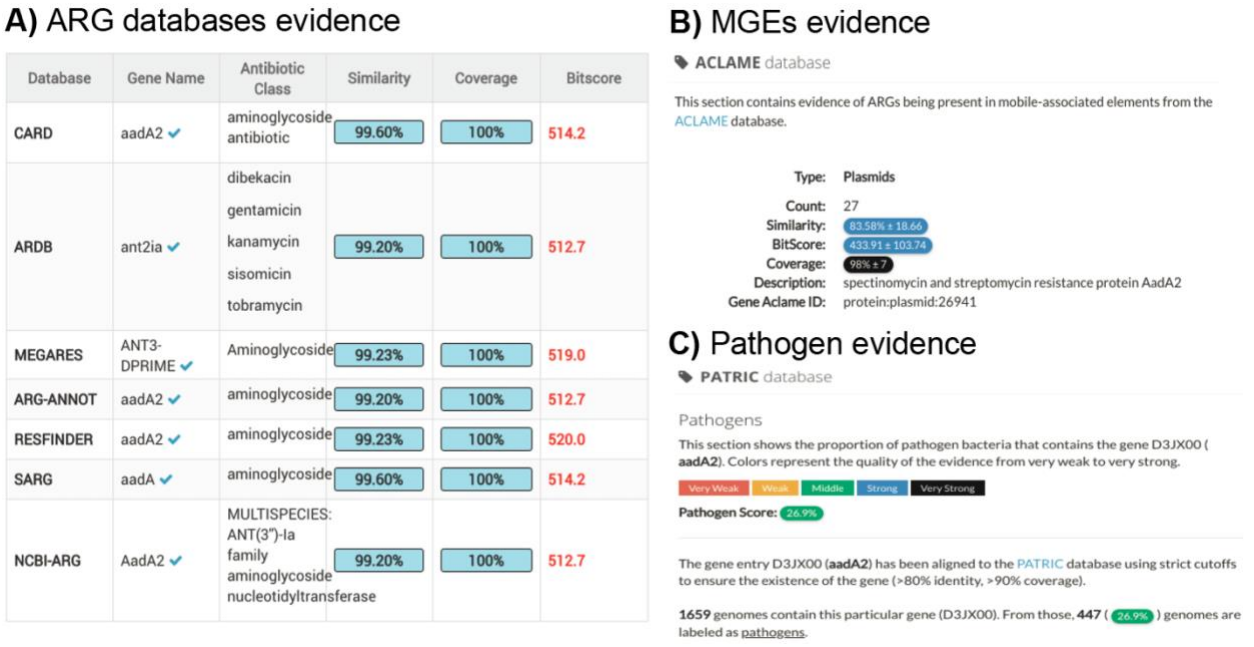

Figure 4.2: Evidence of ARGs in ARGminer platform. **A)** Antibiotic resistance database hits. **B)** Mobile Genetic Elements. **C)** Evidence the ARG is being carried by a pathogen.

4.2.2 Mobile Genetic Element Curation

The ACLAME database [159] primarily houses genes associated with plasmids, viruses, and phages and was used to identify ARGs that have potential of being mobilized by MGEs. DIAMOND [40] was used to compare ARGs to MGEs via sequence alignment (parameters e-

value < 1e-10). Alignment information, along with MGE metadata, is presented in the interface for users to make a decision on whether an ARG has enough evidence of being carried by an MGE or not. This evidence is scored from 0 to 5. A colorimetric ranking scale depicts the degree of confidence for the information presented in the MGE panel (see **Figure 4.2B**).

4.2.3 Pathogen Sequence Curation

A total of 98,758 bacterial genomes were downloaded from the PATRIC [137] database. This database contains information about bacterial pathogenicity, antimicrobial resistance phenotype, corresponding diseases, and host organisms, information that is particularly valuable to clinicians seeking to identify pathogens and potential antibiotic resistance traits. For instance, the UniProt gene entry BAE06009.1 was present in 2,037 bacterial genomes, of which, 1,004 belong to pathogenic bacteria, 40 are involved in cystic fibrosis disease in humans, and 706 exhibit intermediate and resistant phenotypes (see **Figure 4.2C**). The collection of ARGs were then screened against the genome sequences from PATRIC using DIAMOND [40]. To ensure the quality of the assignments, all genes with an identity below 90% and an alignment coverage below 90% were discarded. Users are asked to rate the potential pathogenicity of known bacterial hosts of ARGs based on the evidence provided by PATRIC (frequency of pathogenic genomes, diseases, antimicrobial phenotype, and hosts).

4.2.4 Annotation microtasks

An annotation task consists of labeling ARGs based on the evidence provided on the web site. Users are requested to classify an ARG in terms of gene name, antibiotic class, and antibiotic mechanism. In addition, users are asked to rank the evidence that this ARG sometimes occurs on MGEs or pathogens. A user-friendly web interface makes it easy to follow the annotation process (**Figure 4.3**). Through crowdsourcing and converting the complex annotation tasks into achievable microtasks, ARGminer advocates mass collaboration from an open community that includes both experts and the general public to tackle the difficult task of ARG annotation.

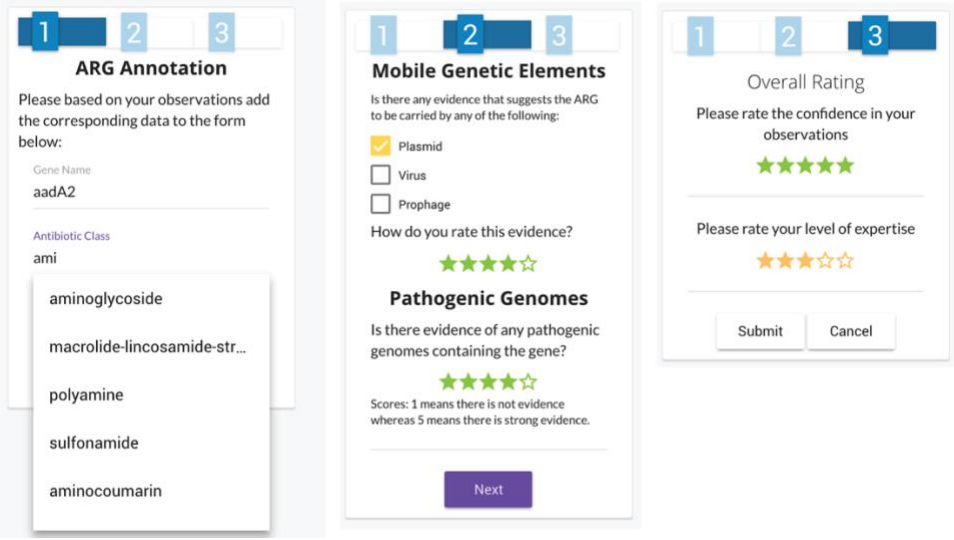

Figure 4.3: Annotation process: First, ARG name, antibiotic class and ARG mechanism are requested to be filled by curator. Then, workers are requested to check the evidence about MGEs and pathogens and score their observations. Finally, workers are requested to rate their confidence and expertise in a scale from 1 to 5.

4.2.5 ARG nomenclature prediction

ARGminer includes a machine learning model based on a word embedding representation for the prediction of the gene name nomenclature given the text metadata information available from different databases. To this end, information such as ARG names, antibiotic classes, and other text data was collected from the CARD database [22]. In total 2,355 ARGs' metadata and names were used for training and testing the model. Labels were defined as the shape of the ARG names. For instance, the label for *opm*E is xxxX, the label for the tetracycline gene tet(A) is xxx(X), and the label for the Beta-lactamase gene TEM-21 is XXX-N. X, x and N correspond to a letter (uppercase or lowercase) and a number, respectively. The nomenclature data set for training and testing was built as follows:

1. Obtain ARG sequences, names, and metadata from CARD.
2. Align CARD sequences to other databases (deepARG-DB, ARDB, ResFinder, ARGANNOT, SARG, NDARO) and extract the best hit.
3. Extract corresponding metadata for the best hits.
4. Build the data set.

Once the data set was built, 80% of the entries were randomly selected for training and the remaining for validation. This process was repeated ten times to check consistency and variability of the results. FastText, a library for text classification and representation using word embeddings [42, 43], was used to build the model. Briefly, the model was trained with default parameters along

an embedding space of 100 dimensions during 100 epochs. **Figure 4.4** shows the workflow of the ARG nomenclature prediction framework.

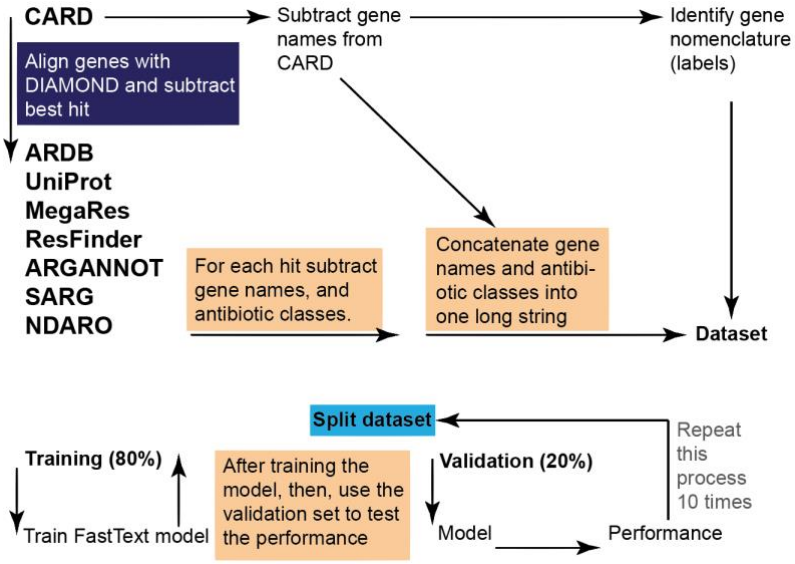

Figure 4.4: General framework used for building the gene nomenclature dataset and to build the machine learning predictor using the natural language processing library FastText.

4.2.6 Expert gold standard data set

To assess the accuracy and quality of classifications generated by crowd-sourcing, three domain experts who are actively engaged in environmental ARG research were asked to annotate a gold standard set of 35 ARGs by name, antibiotic class, and mechanism. In total, 34 out of the 35 ARG annotations were in agreement among at least two of the three experts in terms of antibiotic class and gene name. Note that the gene with UniProt accession number AFU35065.1 was labeled with a different gene name (Isa, Isa-A, Isa-E) by the three experts and therefore removed from the gold standard dataset. These 34 ARGs were further considered in downstream analysis.

4.2.7 Crowdsourcing microtasks

Annotations were performed by two groups of curators. One group was recruited via Amazon Mechanical Turk (MTurk), an online platform that allows access to a broad crowdsourcing audience to perform Human Intelligent Tasks (HITs). When a curator performs an annotation, ARGminer prompts a token that curators need to submit to the MTurk web site for validation to then obtain a monetary reward. Because of the high diversity of MTurk curator backgrounds, the ARGminer HITs were opened to a broad audience, including domain experts and nonexperts, and curators were allowed to perform a limited number of annotations (maximum 20). Another group of curators consisted of students enrolled in a graduate-level microbiology class. Not all of the students possessed deep antimicrobial resistance domain knowledge, but they all had general familiarity with microbiology.

4.2.8 User interface

The ARGminer interface has three main components or sections:

1. **Current Annotation**: Summarizes current available information for a given ARG. It consists of the gene name, antibiotic class, database from which the sequence was extracted, and number of times the gene has been inspected by curators (see **Figure 4.5A**).
2. **Evidence**: Corresponds to the metadata available for the ARG as well as the best hit from the CARD, ARDB, and MEGARes databases. It also provides evidence and information regarding whether the gene is likely carried by an MGE (the ACLAME database) and whether the gene is known to be found in pathogen genomes (the PATRIC database, see **Figure 4.5B**).
3. **Microtask**: Refers to the section where a curator enters his or her annotation. The information in this panel must be consistent with the observations from the evidence. It consists of three simple steps. First, curators must validate the gene name, antibiotic class, and mechanism by looking at the Evidence section. Second, curators must rank the MGE and pathogen evidence. Third, curators must rank their overall annotation by scoring their expertise (how familiar they are with ARGs) and confidence (how strong the evidence is, see **Figure 4.5C**).

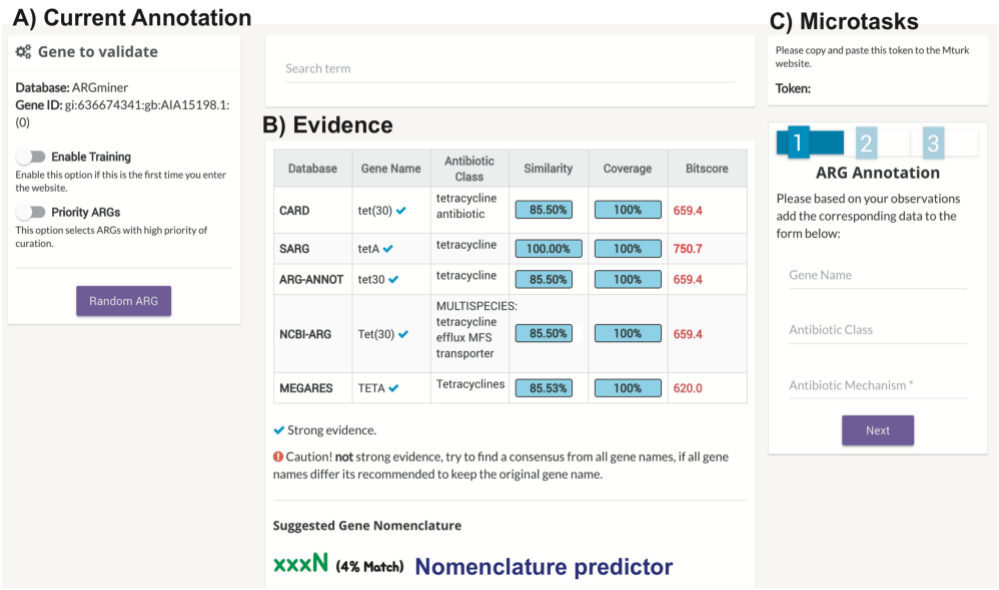

Figure 4.5: General overview of the ARG-miner platform. **A)** Current annotation. This panel contains the current information available for the ARG entry that requires validation. The "priority ARGs" option enables to curate ARGs in the database that have conflicting annotations. **B)** Evidence. This is the main panel and provides all of the metadata and information extracted from the different databases and resources. **C)** Microtasks. This section contains the three microtasks needed for the ARG curation.

The web interface provides a training step for new users that is mandatory for AMT curators (required for getting a monetary reward). The goal of this step is to familiarize the curators with the platform environment by performing two microtasks. ARGminer also provides a list of problematic ARGs that have inconsistent annotations. These problematic ARGs are identified by comparing the annotation of the genes with their best hits from ARDB, CARD, and MEGARes. All tests performed during validation were completed using these problematic ARGs.

ARGminer also provides an administrative interface to update the ARG database. This interface comprises a set of figures that show the distribution of different labels as well as the MGE and pathogenic evidence scores. In this interface, ARGminer administrators are able to accept or reject the annotations made by the crowd and update the ARG database (see **Figure 4.6**).

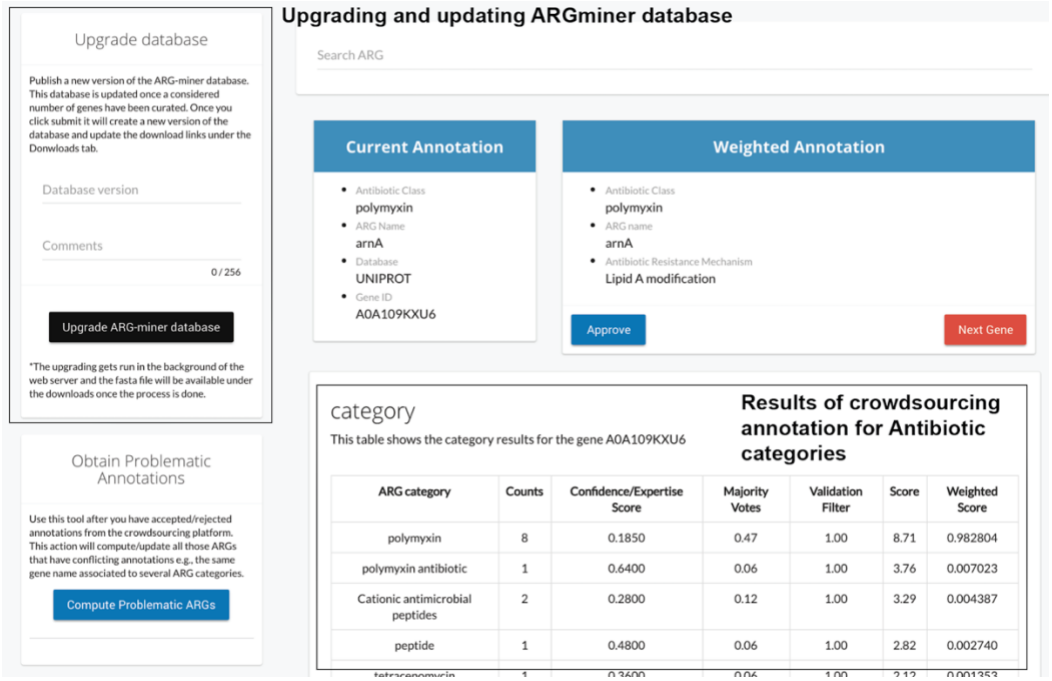

Figure 4.6: Administration page of ARGminer. Once a gene is reviewed by a select number of workers, administrators of ARGminer can evaluate their annotations and approve or disapprove the crowd classification.

4.2.9 Trust validation filter

Because of the unsupervised nature of crowdsourcing, users can provide erroneous feedback or just ignore the evidence and enter random inputs. Under an uncontrolled scenario, spammers can even get a monetary reward. More critically, too much random and/or erroneous feedback can increase the variance in ARG annotations and propagate annotation error. To circumvent this problem, ARGminer implements a trust-validation filter to evaluate whether the input corresponds to actual evidence or not. This score is computed in real time, and unless the user provides valid information, the system will not proceed to the next stage. **Figure 4.7** shows an example of a user providing erroneous input for the antibiotic class field.

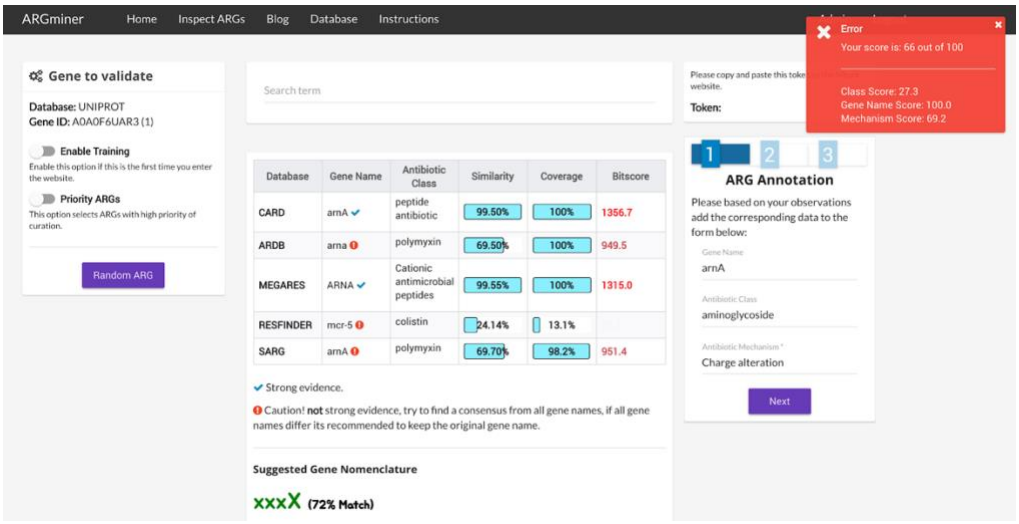

Figure 4.7: Trust validation blocks entries with values that are not in the evidence section.

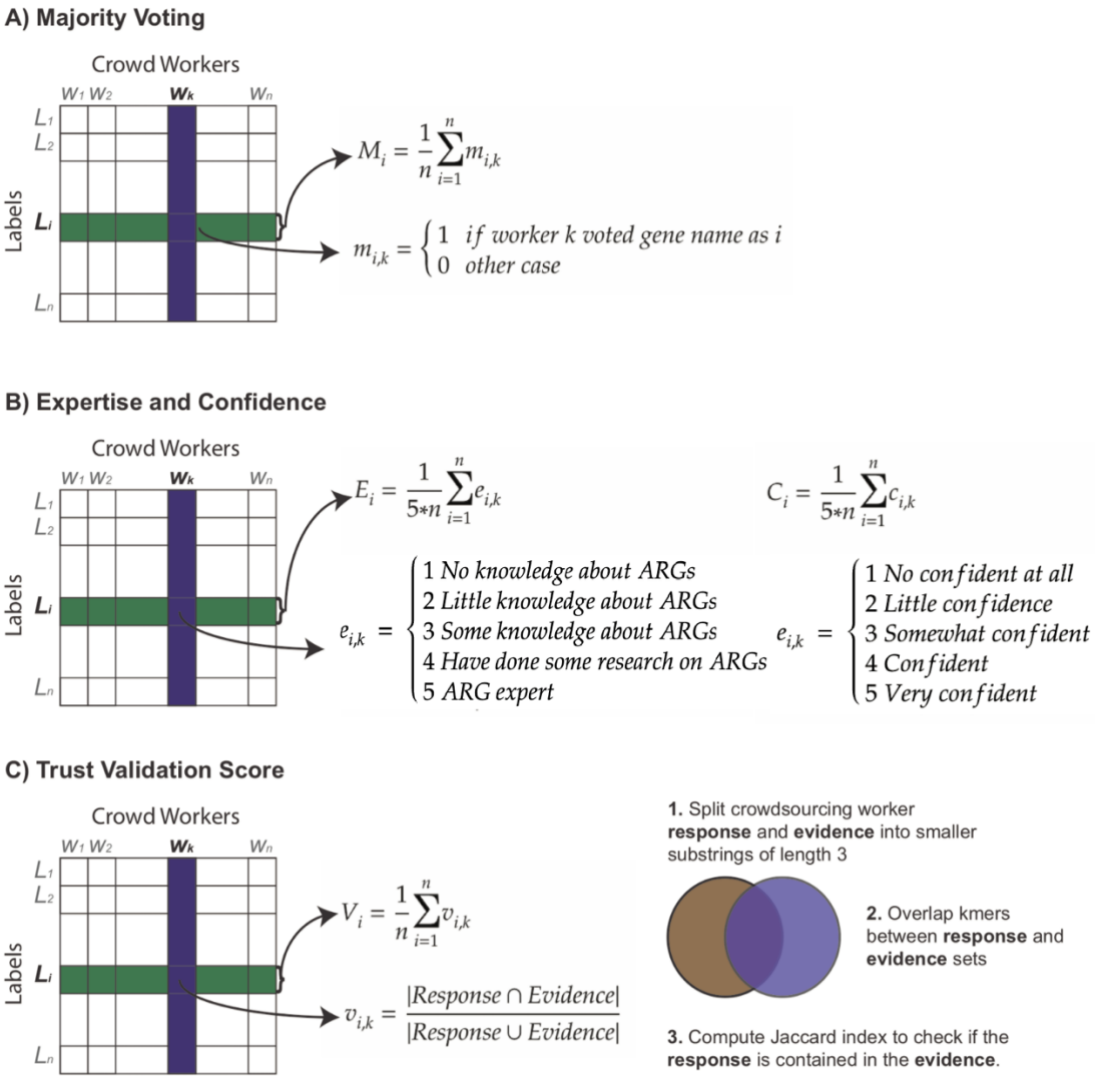

Figure 4.8: Scoring strategy used for annotation of ARGs. **A)** Majority voting score obtained from the total number of workers. **B)** Expertise and confidence normalized from 0 to 1 scores. **C)** Strategy to compute the score for the trust validation filter.

4.2.10 Annotation score

ARGminer scores the curation of ARGs based on the majority voting strategy described by Prill. et. al., [165] weighted by the trust validation-filter score (V), curator's confidence (C) and expertise (E). To describe these values, let us assume we want to score the names of a gene g_i, that has been inspected by n crowdsourcing curators who had available a total of L different gene names to choose from. The majority voting strategy, the expertise, the confidence and the trust validation scores can be arranged in a matrix where columns represent the crowdsourcing workers and rows represent each label (See **Figure 4.8**). Thus, to measure the score for each labels, we need to sum up the row scores and divide them by the total number of workers. The final annotation

score for the label L_i is equal to the multiplication of the individual scores ($A_i = M_i * E_i * C_i * V_i$).

4.3 Results and Discussion

4.3.1 Nomenclature prediction

A machine learning model to assist annotators to identify the correct ARG nomenclature was embedded into the ARGminer platform to assists curators to determine the gene name nomenclature. To train the model, 17 different gene name shapes with at least 10 genes were identified (see **Table 4.1**). Thereafter, to assess the performance of the nomenclature prediction module in ARGminer, precision and recall over the validation set were computed. Precision corresponds to the ratio between the number of correctly predicted gene names over the total number of genes in the validation data set, while recall is the ratio that corresponds to the number of correctly predicted gene names over the total number of genes with this name. The model achieved a high precision of 0.87 ± 0.02 and recall of 0.87 ± 0.02 in the validation set composed of 452 entries that were not used during the training process. **Table 4.2** shows examples of predicted gene names along with their input. For instance, the UniProt entry D3JX00 has been reported by different resources as aadA2, ant2ia, ANT3-DPRIME, and AadA2, the system recommends the gene name to have the shape xxxXN with a 63% match.

Nomenclature	ARG counts	ARG Example
XXX-N	1153	CMY-2
xxxX	338	ileS
XXX-X-N	174	CTX-M-124
XxxXN	102	FosA4
XxxX	86	OqxA
xxxXN	65	dfrB6
xxxXX	53	vanXA
XXX(N')-Xx	37	AAC(6')-Ip
xxx	29	cat
XXXX-N	25	HERA-1
xxxN	24	tet32
XXX-N-N	23	OXY-1-6
XXX(N)-Xx	17	ANT(6)-Ia
xxxx	16	mdtl
Xxx(N)	15	Erm(35)
XXX(N')-XXx	12	AAC(6')-IIa
xxx-N	11	arr-8

Table 4.1: Nomenclature shapes detected in CARD. The table shows shapes that have at least 10 genes.

Training text	Label
CTX-M beta-lactamase cephalosporin antibiotic inactivation bl2be_ctxm ceftazidime cephalosporin_i cephalosporin_ii cephalosporin_iii monobactam penicillin betalactams CTX beta_lactam blaCTX-M-15 beta-lactam blaCTX-M-142 beta-lactam CTX-M-142 class A extended-spectrum beta-lactamase CTX-M-142	XXX-X-N
vanY glycopeptide resistance gene cluster glycopeptide antibiotic antibiotic target alteration vanyg vancomycin Glycopeptides VANYG glycopeptide vanY-B glycopeptide VanXY_C2 vancomycin vanG MULTISPECIES: D-Ala-D-Ala carboxypeptidase VanY-G1	xxxXXN
major facilitator superfamily (MFS) antibiotic efflux pump tetracycline tetracycline antibiotic efflux pump complex or subunit conferring antibiotic resistance antibiotic efflux tetc tetracycline Tetracyclines TETA tetracycline tet30 tetracycline tet(30) tetracycline tetA MULTISPECIES: tetracycline efflux MFS transporter Tet(30)	xxx(N)

Table 4.2: Examples of entries in the data set for the nomenclature predictor. The training test consists of merging the information from different databases into a long string whereas the label corresponds to the gene name shape.

4.3.2 Crowdsourcing curators

To assess the effectiveness of the crowdsourcing approach for ARG annotation, we evaluated three groups of curators with the following attributes:

1. A set of crowdsourcing curators from MTurk, referred to as **AMT-Free**. In this scenario, curators were paid $0.10 for each annotation, with the trust validation filter disabled to examine the reliability of the general crowd. Therefore, curators could input anything as feedback without restriction. A total of 100 annotations were requested from MTurk for this test.
2. A second batch of crowdsourcing curators from MTurk, referred to as **AMT-Val**. In this case the trust validation filter was enabled. The main purpose of this experimental group was to measure the effectiveness of the trust validation filter. In this scenario, a total of 200 annotations were requested from MTurk.
3. A group of users with general microbiological knowledge, with varying levels of experience in ARG research, referred to as **LAB**. This group consisted of Masters and Ph.D. students from a microbiology class at Virginia Tech. They completed this work as an assignment and did not receive any monetary reward. Here the annotations were performed with the validation filter on, and each curator was requested to perform 15 annotations (540 microtasks in total). The goal of the LAB scenario was to compare its performance against the non-expert community of MTurk (AMT-Val, AMT-Free).

4.3.3 Effectiveness of the trust validation filter

Spammers are curators that intend to obtain monetary reward by submitting invalid information, which is a major confounder to effective crowdsourcing. In the present study, although the ARGminer web site provides curators with detailed instructions about how to handle the annotation process, many of the **AMT-Free** curators submitted misleading or unrelated feedback. For the antibiotic category annotation task, curators must choose the antibiotic class to which they believe the gene belongs from a dropdown menu that contains a list of antibiotic classes. Results indicated that many **AMT-Free** curators simply selected the first option on the dropdown menu (aminoglycosides class), most likely without reading the evidence section of the web page. Thus, it was observed that most of the antibiotic class annotations under the **AMT-Free** group were labeled as aminoglycosides. This is a serious hurdle to accurate database curation and indicates the need for a real time control that guarantees accuracy of the annotation. In terms of performance, as expected, the AMT-Free group achieved very low scores for all annotations (**Figure 4.2**). However, not all curators were spammers. It was observed that a few curators who performed more than ten microtasks also responded correctly and consistently to their observations and evidence. After integration of the trust validation filter, MTurk curators were monitored on their feedback (**AMT-Val** group) (see **Figure 4.9**). As a result, the performance of the **AMT-Val** curators improved significantly for all fields (antibiotic class, ARG name, and ARG mechanism) when compared to the **AMT-Free** group (P=1e-10 for annotation score distributions). Under the new restriction, MTurk curators were not allowed to continue with the microtask until their annotation was valid. Under this test, all nonsense input was eliminated and all annotations from the **AMT-Val** group corresponded to actual ARG evidence prompted by the web site. These results demonstrate the effectiveness of the trust validation filter for the control of spam annotations. In addition, it was imperative to test the performance of the MTurk curators against domain knowledge users. The main goal of this test case was to investigate whether a nonexpert crowd community (**AMT-Free** and **AMT-Val**) can perform a complex task with comparable outcomes as those of a group of curators with domain-knowledge (**LAB**). As expected, the LAB curators achieved a much higher average score (0.146) than the AMT-Free curators (0.06), but, surprisingly, a rather similar score to the AMT-Val curators was observed (0.114). This demonstrates that crowdsourcing is indeed a powerful alternative to manual inspection and annotation of ARGs by domain experts. As expected, MTurk annotations (AMT-Val) were characterized by a higher variance compared to the LAB group in all annotation fields, but the two distributions were not significantly different (Kolmogrov-Smirnov test: $P > 0.05$).

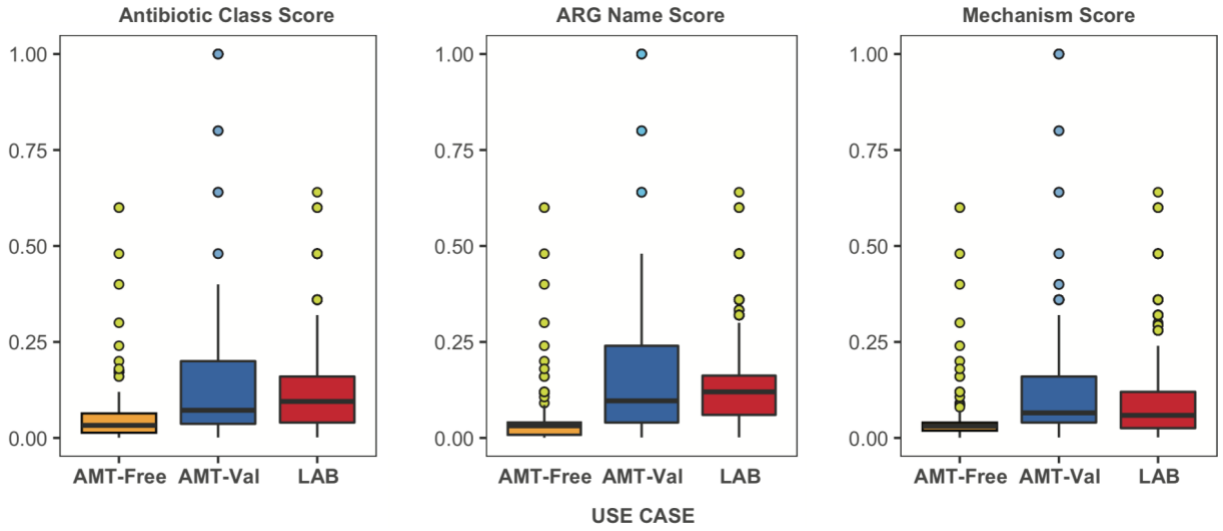

Figure 4.9: Annotation score of the three crowdsourced use cases (AMT-Free: Amazon MTurk curators without the true validation filter, AMT-Val: Amazon MTurk curators with the validation filter enabled and LAB: a group of curators with general microbiology domain knowledge and some antibiotic resistance knowledge. AMT-Val displayed the highest variance. However, this distribution was closer to that obtained by the curators with domain knowledge. Scores from the AMT-Free curators were the lowest among the three scenarios, indicating the ineffectiveness of the crowdsourcing annotation when the curator's input was not validated.

4.3.4 Effectiveness of the scoring strategy

To evaluate the quality of the scoring strategy, four genes were selected among the total set of curated genes and examined in greater detail, as illustrated in **Figure 4.10**. For instance, the UniProt entry A0A0D0NPG2 is a bifunctional polymyxin resistance protein, ArnA, involved in several biological processes including coenzyme binding, UDP-glucuronic acid dehydrogenase activity, lipid A biosynthetic process, and response to antibiotic. This protein builds up the UDP-L-4-formamido-arabinose attached to lipid A complex, which is required to confer resistance against polymyxin and cationic peptides [166].

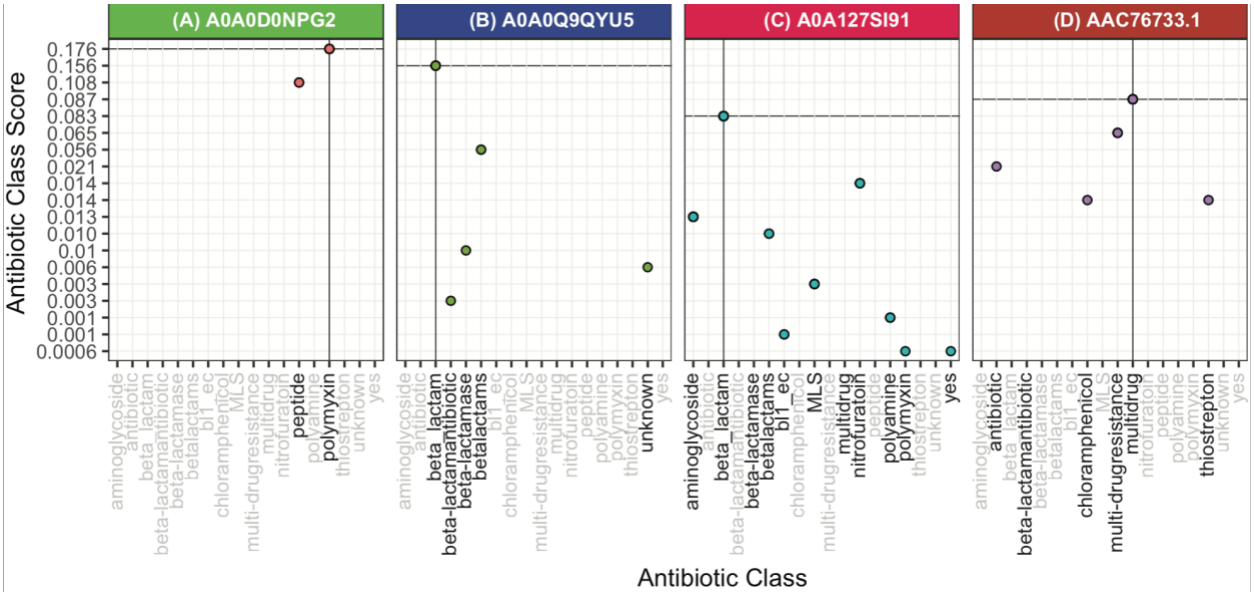

Figure 4.10: Distribution of the antibiotic class annotation by the crowdsourcing curators using the annotation score. X axis corresponds to the antibiotic resistance categories, where black labels indicate the categories reported by the curators and the top of each box corresponds to the ARG identifier.

From the crowdsourcing classification, both peptide and polymyxin antibiotic classes were identified, where polymyxin was characterized by a slightly higher score (**Figure 4.10A**). A closer look at the evidence from the antibiotic resistance databases (CARD, ARDB, and MEGARes) reveals a consensus of the gene towards the polymyxin antibiotic class. The evidence from the antibiotic resistance databases strongly suggests that the gene entry A0A127SI91 corresponds to a bl1-EC beta lactamase gene. **Figure 4.10C** illustrates different crowd classifications (including all evaluation scenarios). Note that beta-lactam is the dominant class with the highest annotation score. However, as a consequence of disabling the trust validation filter, several unrelated categories were accepted, such as aminoglycoside, MLS, multidrug, nitrofurantoin, polyamine, polymyxin, and even the word "yes". One particularly interesting observation is the remarkable similarity among valid annotations. For instance, in **Figure 4.10D**, the gene AAC76733.1 was correctly assigned to multidrug as its best classification and to the "multi-drug resistance" category as its second best classification. Such small semantic differences are not detected by the trust validation filter. Therefore, under the validation interface, the administrators of ARGminer have the ability to validate or reject the annotations if needed. **Figure 4.10B** shows that most curators assigned the gene A0A0Q9QYU5 to the beta lactamase category. Note that the suggested name "beta_lactam" is the highest scored among all choices.

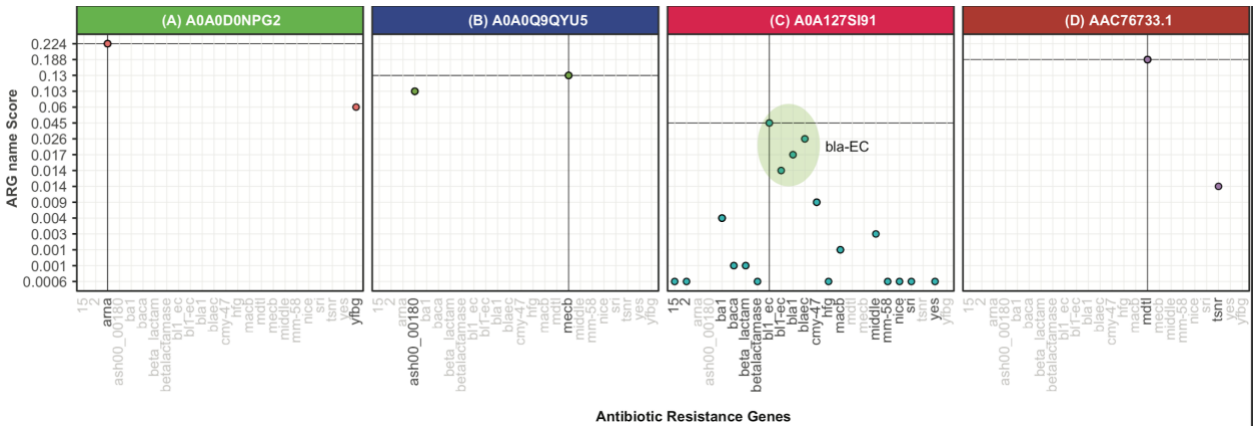

Figure 4.11: Distribution of the prediction of ARG names. ARG names are represented on the x axis and the y axis indicates the corresponding annotation score. The top of each box corresponds to the ARG identifier.

Figure 4.11 shows the crowdsourced score for the ARG name classification. As seen for the antibiotic category annotation, there are cases where the annotations are semantically highly similar. For instance, the gene A0A127SI91 was tagged as bl1_ec, bl1-ec, or blaec, all corresponding to the bla1-EC gene name (**Figure 4.11C**). Note that all these labels were ranked higher than the other gene names (macb, baca, ba1) and all the unrelated tags such as "mm-58", "15", "yes", and "middle". Also, all unrelated annotations were ranked low by the scoring strategy.

Although identification of the antibiotic category for the gene A0A0Q9QYU5 was straightforward, the detection of its gene name is challenging, primarily because the metadata of this entry does not include the gene name and because the identity of its best hit alignments is below 30%. This indicates that the gene has a potential homology to known ARGs. Two ARG databases (CARD and MEGARes) show a significant best hit e-value (<1e-22) over the mecB gene. For this example, 50% of the curators annotated the gene as mecB whereas the other 50% annotated it as ash00_000180. Also, curators yielded a higher confidence for the *mec*B gene (2.6 average confidence score) compared to the ash00_000180 (2.3 average confidence score). As a result, *mec*B achieved a slightly higher score. To document any uncertainty, ARGminer recommends that users retain the original label if the evidence is not convincing. For the other examples (**Figure 4.11A** and **4.11D**), the crowd classified the gene names according to the observed evidence.

4.3.5 Annotation analysis

To assess the accuracy of the crowdsourcing annotation, genes that were inspected by fewer than 10 curators were removed from the total pool of classified genes. Then, a total of 35 genes were identified and manually curated by three domain experts according to antibiotic class and gene name. It was found that experts achieved an annotation pairwise correlation of 0.96 ± 0.02,

indicative of an almost perfect classification agreement. Genes that were classified to the same label by at least two experts were used as the gold standard data set. This benchmark was then used to measure the performance of the crowdsourcing curators where labels were selected based on the greatest annotation score.

The crowdsourcing classification of the antibiotic classes was essentially just as accurate as the expert annotation (94% Positive Predictive Value - PPV). In other words, 33 out of 35 genes labeled via crowdsourcing matched the expert classification. The genes for which the curators failed to identify the correct antibiotic class were a quinolone ARG annotated as multidrug (YP_001693238) and a multidrug gene annotated as quinolone (NP_358469.1). The classification of the ARG names proved to be a challenging task. Indeed, experts did not fully agree about the correct name of five ARGs. However, only one of those conflicting genes was assigned a different classification assigned by all three experts. This gene corresponded to a macrolide gene (AFU35065.1), which was tagged as Isa, Isa-A, and Isa-E by the three experts. Thus, this gene entry was removed for the gene name analysis comparison and the final control data set contained 34 genes. When comparing the gene name annotation from the crowdsourcing curators, their prediction had a 97% PPV. This indicates that only one gene was not correctly classified by the crowd (J2LT98). By examining the details of this gene in ARGminer, all three ARG databases agreed that the gene belonged to the SHV group, with markedly high scores. However, CARD labeled it as the SHV variant 1 (SHV-1), ARDB labeled it as variant 2 (SHV-2), and MEGARes labeled it as the SHV group, without specification of a variant. An interesting aspect with respect to this particular ARG is that variants are defined by specific amino acid modifications [167], thus these genes are highly similar and identifying the correct variant by using sequence alignment is a particularly difficult task. This aspect has the potential to confuse curators when classifying genes that are highly similar. Interestingly, by examining the crowd results, curators were able to discard the SHV variant 2 (99.3% identity), but they were not able to differentiate between the SHV variant 1 and the SHV group (both have the same score). These results suggest that crowdsourcing curators are able to follow the correct track, even in the face of complex tasks. Interestingly, the gene name prediction model assigned the gene name nomenclature to XXX-N with a probability of 0.79, which corresponds to the nomenclature followed to name SHV beta-lactamase genes. Because of the risk of propagation errors, the updating process of ARGs requires administrator approval for new database releases in the ARGminer web site. Overall, the crowd exhibited performance comparable with the experts, but in much less time. These results suggest that crowdsourcing annotation is a strong alternative to manual classification and validation of ARGs by domain experts.

4.3.6 Expertise and confidence

ARGminer asks users to rate their own expertise in the analysis of ARGs on a scale of 1 to 5. **Figure 4.12A** shows the distribution of the expertise score against the correct or incorrect annotations for the antibiotic category classification (including all scenarios: AMT-Free, AMT-

VAL, and LAB). Surprisingly, it is clear that having expert knowledge does not really make a difference in the quality of the classification. Indeed, because of the open nature of AMT, most of the curators are not experts and have little knowledge about ARGs. From Figure 6A, it is also evident that the proportion of correct annotations was higher compared to the incorrect classifications (the size of the dot indicates the number of annotations). This result suggests that accurate detection of the correct antibiotic resistance category does not necessarily require domain expertise. On the other hand, curators were also required to rate their confidence in the annotation. **Figure 4.12B** shows that self-rated confidence is a strong predictor of the quality of the annotation. The distribution of the confidence score indicates that higher confidence correlates with more accurate results. For instance, 95% of the curators who rated their confidence with 5 stars achieved correct annotation and only 5% missed. This strongly suggests that the confidence score is a superior indicator of correct annotation over the expertise score.

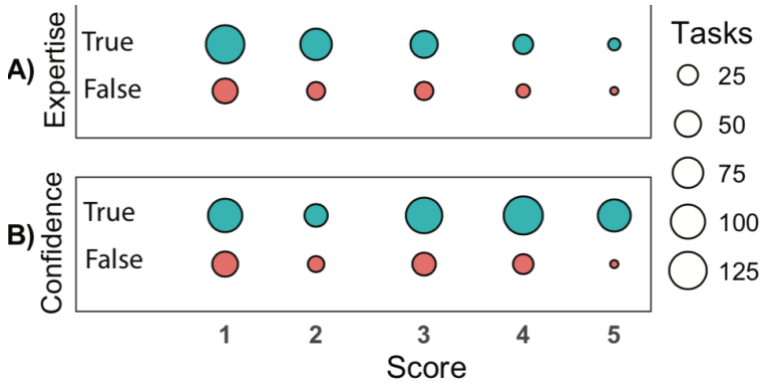

Figure 4.12: Expertise and confidence levels of the curators. The size of the points indicates the number of tasks; the x axis corresponds to the score level and the y label shows the expertise and confidence parameters. Color depicts correct and incorrect classifications.

4.4 Conclusions

Here we develop, launch, and validate a new web platform, ARGminer, a powerful system that harnesses the power of crowdsourcing for advancing robust and comprehensive curation of ARGs. ARGminer enables easy access to key relevant information pertaining to ARGs, including metadata, evidence of ARGs being carried by pathogens, and the possibility of ARGs being mobilized by MGEs. Further, it enables a simple, but powerful, tool for the curation of ARGs designed to provide accurate information represented in a noncomplex way that can be validated by users without the requirement of domain knowledge. Results demonstrated not only that crowdsourcing curators yield curations as accurate as experts, but they are also more efficient than

ARG-domain experts. Thus, ARGminer opens the possibility of a truly comprehensive, accurate, and perpetually up-to-date publicly available ARG database.

CHAPTER 5 ANNOTATION OF LONG NANOPORE READS

Direct and indirect selection pressures imposed by antibiotics and co-selective agents and horizontal gene transfer are fundamental drivers of the evolution and spread of antibiotic resistance. Therefore, effective environmental monitoring tools should ideally capture not only antibiotic resistance genes (ARGs), but also mobile genetic elements (MGEs) and indicators of co-selective forces, such as metal resistance genes (MRGs). A major challenge towards characterizing the potential human health risk of antibiotic resistance is the difficulty in identifying ARG-carrying microorganisms, of which human pathogens are arguably of greatest risk. Historically, short reads produced by next-generation sequencing technologies have hampered confidence in assemblies for achieving these purposes. Here, we introduce NanoARG, an online computational resource that takes advantage of the long reads produced by nanopore sequencing technology. Specifically, long nanopore reads enable identification of ARGs in the context of relevant neighboring genes, thus providing valuable insight into mobility, co-selection, and pathogenicity. NanoARG allows users to upload sequence data online and provides various means to analyze and visualize the data, including quantitative and simultaneous profiling of ARGs, MRGs, MGEs, and putative pathogens. NanoARG is a Web platform dedicated to the analysis of ARGs from nanopore sequencing metagenomes and provides context of co-located genes, including ARGs, MGEs, MRGs, and taxonomic markers. A user-friendly interface allows users the analysis of long DNA sequences (including assembled contigs), facilitating data processing, analysis, and visualization. NanoARG is publicly available and freely accessible at http://bench.cs.vt.edu/nanoARG.

5.1 Introduction

Antimicrobial resistance (AMR) compromises the ability to prevent and treat infectious disease and represents a highly significant and growing global public health threat [168]. It is currently estimated that the annual number of deaths worldwide due to antibiotic resistance will top ten million by 2050 [28]. In response, numerous national and international agencies have called for expanded monitoring both in the clinic as well as in environmental settings. In particular, environmental monitoring can provide insight into not only human and agricultural inputs of antibiotic resistant bacteria and antibiotic resistance genes (ARGs), but also factors contributing to the evolution and spread of resistant pathogens. For instance, various environmental compartments, such as wastewater treatment plants, livestock lagoons, and amended soils, can act as "environmental reactors," in which resistant bacteria discharged from domestic, hospital, industrial, and agricultural waste streams have the opportunity to interact with native aquatic and soil bacteria in the presence of selection pressures to potentially give rise to new resistant forms [140, 169]. Humans may subsequently be exposed to resistant organisms via consumption of food crops affected by biological soil amendment or irrigation, as well as through contact with treated and untreated water used for recreational, hygienic, and potable purposes [29, 170].

Molecular-based monitoring presents many advantages over culture-based techniques for tracking antibiotic resistance in the environment. This is particularly true with respect to the potential to recover rich information regarding the carriage and movement of ARGs within complex microbial communities. Culture-based techniques are time consuming and only provide information about one target species at a time, thus potentially overlooking key microbial ecological processes that contribute to the spread of AMR. Thus, directly targeting ARGs as "contaminants" of concern that transcend bacterial hosts has gained popularity. In particular, horizontal gene transfer (HGT) [171] plays a critical role in the rise of new resistant strains and the dissemination of AMR in microbial ecosystems [172]. Intercellular transfer of ARGs among bacteria is facilitated via mobile genetic elements (MGEs) including integrons, transposons, or plasmids [173]. Integrons are key genetic elements of interest as they facilitate capture of multiple ARGs and are often embedded in MGEs, thus effectively functioning as vehicles for dissemination of multidrug resistance [162]. The mechanisms involved in HGT include conjugation, transformation, transduction, and homologous recombination, where DNA is incorporated by transposition, replication, and integration [173].

Multi-drug resistance has emerged as a major clinical challenge. For example, methicillin resistant *Staphylococcus aureus* (MRSA) is responsible for major hospital infections, with few options for treatment, especially when resistant to vancomycin [174]. More recently, New Delhi Metallo beta lactamase (*bla*NDM-1) has emerged as a major concern, as it encodes for resistance to powerful last-resort carbapenem antibiotics and is carried on a highly mobile genetic element associated with multi-drug resistance that has been detected in several different pathogenic species, including *Escherichia coli*, *Klebsiella pneumoniae*, *Providencia rettgeri* and *Acinetobacter baumannii* [175-177]. This example emphasizes that, ideally, monitoring technologies should provide a rapid and robust characterization of ARGs and their likely association with MGEs, multi-drug resistance, and carriage by pathogen hosts. In this regard, shotgun metagenomic sequencing techniques have emerged as a promising tool for the characterization of the diverse array of ARGs found in different environments [30, 169, 178, 179]. In particular, high-throughput next-generation DNA sequencing technologies, such as the Illumina platform [132] and 454 pyrosequencing [180, 181], have enabled a new dimension to ARG monitoring in the environment.

While providing unprecedented amounts of sequence information (360,081 metagenomes processed on MG-RAST [182], a total of 20,120 samples on EBI-metagenomics [5], and 3,038 on MetaStorm [19]), a major drawback of these technologies is the very short DNA sequence reads produced, at most a few hundred nucleotides long. Nonetheless, next-generation DNA sequencing is growing in use as a powerful means of profiling ARG occurrence in various environments. ARGs can be identified by direct annotation through comparing sequences against available ARG databases. This enables relative quantitative comparisons, including relative abundance calculations (e.g., normalization to 16S rRNA genes or total ARGs). Alternatively, short reads can be assembled into longer contigs for assembly-based annotation, which can improve resolution

in identifying ARGs and can also provide information about neighboring genes. Both approaches have limitations. The first can only be used to detect previously-described ARGs that populate available databases [183] and requires determination of an arbitrary DNA sequence identity cutoff [20]. This process generally undermines the possibility to identify novel ARGs, although a novel similarity based method was recently proposed to annotate ARGs with low similarity to existing database ARGs [25]. Assembly, on the other hand, requires deeper and more costly sequencing along with greater computational resources [184] and still can produce incorrect contigs and chimeric assemblies [185]. For these reasons, it is important to be cautious in interpreting results derived from the assembly of short sequence reads because of the possibility of assembly errors and the lack of standard means to estimate confidence in assembly accuracy [75, 186, 187]. Also, the quantitative value of data is lost following assembly.

In 2014, Oxford Nanopore Technologies (ONT) released the MinION nanopore sequencer, which provides long sequence reads averaging 5kb in length [38] and even upwards of 100kb [188]. A major disadvantage of nanopore technology, however, is the high error rate, estimated by Jain et al. (2016) to be below 8% [189]. However, this error rate represents a marked improvement over an earlier estimated error rate of 38% [190], with a general trend towards reduced error rates with the help of read correction algorithms [8]. It has been shown that nanopore technology can produce highly accurate assemblies, in the range of 95% when applied to whole-genome sequencing [191-193]. Nanopore sequencing has also been applied for shotgun metagenomics, including identification of viral pathogens [194], assessment of microbial diversity in extreme environments [195], and detection of ARGs in various environments [103, 196-200]. To date, nanopore sequencing has not been applied for the purpose of metagenomic profiling of ARGs in environmental samples.

Long length nanopore reads offer a unique opportunity to explore the context of ARGs in terms of co-occurrence and potential for mobility. Unlike *de novo* assembly of short reads into longer contigs, while might produce chimeric sequences [201], nanopore sequencing inherently yields long sequences, thus reducing the potential for chimeras. Therefore, nanopore sequencing has potential to become a powerful tool for the identification of the coexistence of ARGs, MGEs, and MRGs. Such an approach could substantially advance environmental monitoring approaches, providing insight into the potential dissemination of AMR through co-occurrence and co-selection of ARGs and other relevant genes and genetic elements [69, 97, 118]. The co-occurrence of ARGs and MGEs also enables tracking of evidence of genetic events of interest, such as HGT [200].

Here, we introduce NanoARG, a user-friendly online platform that enables comprehensive profiling of ARGs in environmental samples using nanopore sequencing data. In addition to comprehensive ARG profiling, NanoARG also provides identification of MRGs, MGEs, taxonomic markers, and sequences with high similarity to known pathogens, along with interactive visualization of linkages among these various elements on the same DNA strand. To demonstrate

the potential of NanoARG for environmental ARG profiling, several MinION nanopore sequencing libraries, including environmental and clinical samples, were analyzed. The Web service is freely available at http://bench.cs.vt.edu/nanoARG/. It requires a user login and subscription to upload and process nanopore sequencing data.

5.2 Materials and Methods

5.2.1 Web Service and Pipeline

Figure 5.1 illustrates the NanoARG architecture. The workflow has three major components: 1) a web interface, where users can upload data and monitor the progress of the analysis (**Figure 5.1A**); 2) a REpresentational State Transfer (RESTful) application program interface (API), which monitors and sends the raw MinION nanopore sequencing data to a computing cluster for processing (**Figure 5.1B**); and 3) a backend platform for retrieval of results and downstream analyses (**Figure 5.1C**), such as taxonomic annotation, gene co-occurrence analysis, human pathogen-like sequence detection, network analysis, and multiple sample comparisons. The nanopore reads are screened against databases currently available using different 'omics tools, both of which will be updated in the future when an improved version is available. Results are stored as JavaScript Object Notation (JSON) files. Metadata and user information are encrypted and stored in a Mongo database. The workflow runs on a large distributed system in the Advanced Research Computing (ARC) center at Virginia Tech. The cluster is managed by the qsub queuing system [202].

The Web service provided by NanoARG includes several features to facilitate analysis of environmentally-derived metagenomic data obtained via nanopore sequencing. Users can submit data to the NanoARG Web service using a simple graphical user interface (**Figure 5. 2A**). In the current version of NanoARG, data submitted to the system is stored privately. To start using the service, users are required to register an account with their email address, which allows them to manage and control submitted samples and projects. Users can voluntarily share their projects with other users by sharing additional email addresses. To create a project, a few parameters, such as name, description, and biome type (**Figure 5. 2B**), are required. Inside each project, users can add new samples, run new analyses, or remove or rerun existing samples (**Figure 5. 2C**).

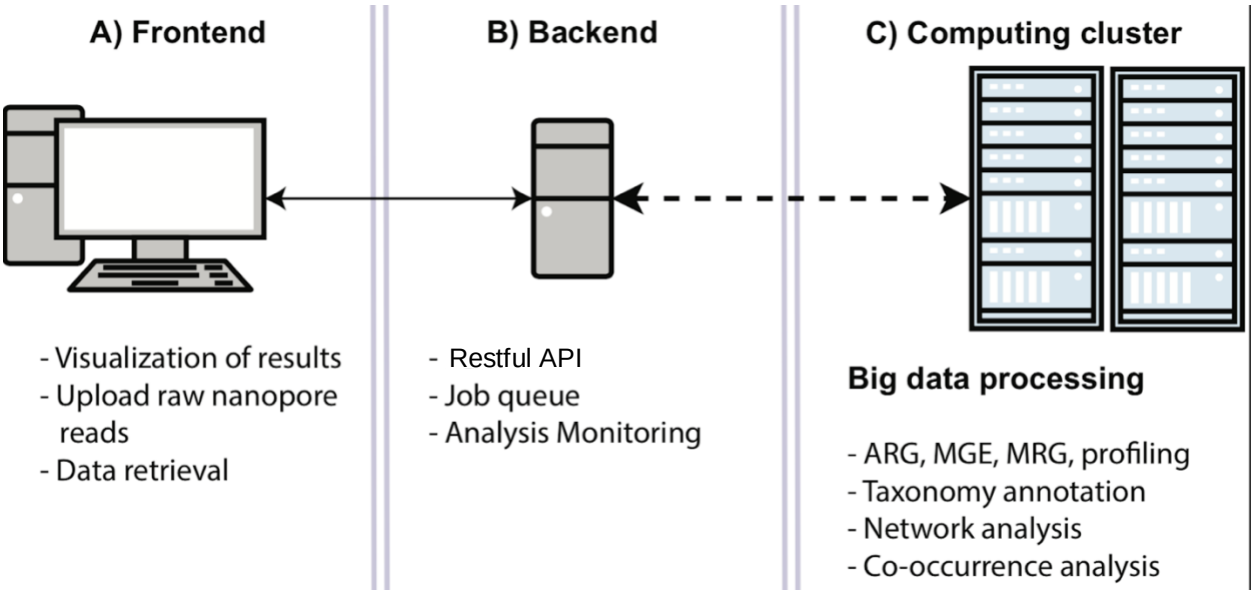

Figure 5.1: NanoARG architecture. **A)** The Frontend is the link between users and the analytical tools, allowing raw data upload and results visualization. **B)** A backend RESTful API manages the data, triggers the analysis, and monitors the status of the analysis. **C)** The computing cluster module processes the data and executes ARG, MGE, MRG and taxonomic profiling.

NanoARG provides several types of visualizations to interpret the results and allows users to download results (e.g., absolute and relative abundances, co-occurrence network associations, taxonomy annotations, and ARG context patterns) in a tabular format containing the fields required for tuning the results (E-value, identity percentage, and coverage). These tables can be used for further processing and statistical analysis. The NanoARG Website was developed using the Google Angular 5 framework (https://angular.io), the back-end was developed with the Node.js framework (https://nodejs.org/en/). Finally, the computing pipeline was developed using the Luigi framework, allowing the monitoring and rescheduling of jobs that failed during execution (https://github.com/spotify/luigi).

5.2.2 Required Data Types

NanoARG requires users to upload nanopore reads in FASTA format [203], thus requiring that the users have already preprocessed the raw fast5 files from the nanopore sequencing device. This step can be done using a base-calling program such as Albacore [204], Metrichor [38], or Nanocall [205], with a sequence extractor toolkit such as poretools [206]. Barcode recognition and read sorting by barcodes can be conducted along with base calling. Before submitting data to the system, users must provide simple metadata consisting of sample name, biome, location, and comments and can also manually enter details about DNA extraction methodology, if so desired. Then, following four simple steps (insert metadata, upload files, set up parameters, and execute), users can submit the data and initiate analysis (**Figure 5. 2A**).

5.2.3 Data Processing

Once the data is uploaded to the computing cluster, it is processed by several modules that perform a set of tasks to obtain annotation profiles for ARGs, MGEs, MRGs, and associated taxa (**Figure 5. 3**). The status of the analysis can be easily monitored through the user interface (**Figure 5. 2C**).

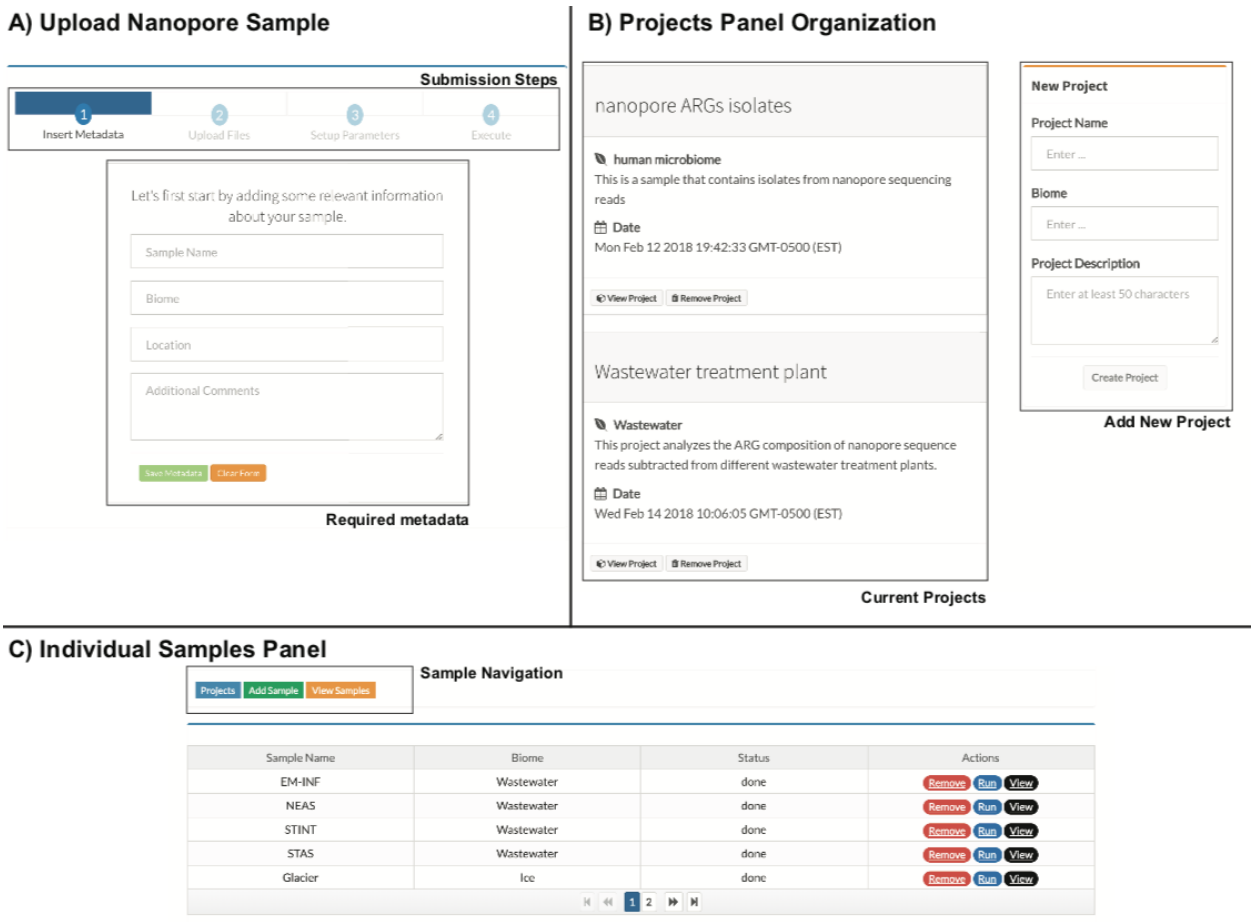

Figure 5.2: User Interface. **A)** Steps and metadata required to upload samples to NanoARG. **B)** Projects are organized based on the creation date and visualized as a timeline post. **C)** List of samples under a project displaying basic metadata (Biome), the monitor variable (Status) and the three actions that can be performed by users.

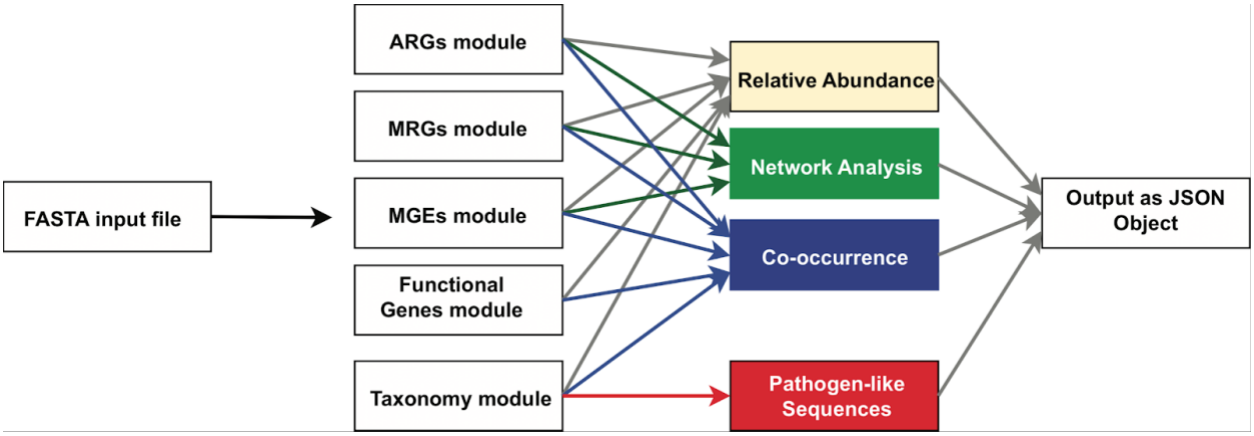

Figure 5.3: General overview of the NanoARG pipeline. FASTA input reads are processed by five modules to annotate reads according to ARGs, MRGs, MGEs, other functional genes and taxonomic affiliation. Annotations are then processed through several stages to achieve the desired analysis (relative abundance, network analysis, co-occurrence, and putative pathogens). All analyses are packed into a JavaScript Object Notation (JSON) file that can be easily streamed using an http request.

5.2.4 Clustering of Local Best Hits for Annotating ARGs, MRGs, and MGEs

Traditionally, the analysis of long sequence reads, such as assembled contigs, is achieved by first identifying open reading frames (ORFs) within the sequences [19, 144, 207, 208] and then searching (e.g., by utilizing BLAST) the ORFs against a database for functional annotation. While nanopore sequences are analogous to long contigs, the high sequencing error rate can limit detection of ORFs. Therefore, NanoARG deploys DIAMOND [40] to align reads against the corresponding databases. Then, it clusters all the local best hits into regions, and determines the annotation of each region using either the best hit approach or the DeepARG prediction [25], as shown in **Figure 5.4**. Specifically, DIAMOND [40] is run with permissive parameters (E-value 1e-5, identity 25%, coverage 40%, and --nk 15000), while bedtools [209] is used to cluster the local best hits in each read into regions. **Table 5.1** describes the databases, methods, and parameters used in nanoARG. The resulting regions/clusters are then annotated for ARGs, MRGs, and MGEs, as detailed below.

Module	Database	Method	Parameters
ARGs	deepARG-db	deepARG-LS	--iden 25 --prob 0.5 --cov 0.4
MGEs	NCBI-NR + I-VIP	Diamond	--evalue 1e-5 --iden 25 --nk 15000
MRGs	BacMet	Diamond	--evalue 1e-5 --iden 25 --nk 15000
Taxonomy	Bacteria, Aarchaea, Viruses, Human	Centrifuge	default
Pathogens	ESKAPE + WHO	Pattern matching to NCBI Taxa ID	-

Table 5.1: NanoARG modules, parameters and methods

5.2.5 ARG Module

Following the clustering procedure of the local best hits to identify putative regions of interest (**Figure 5. 4**), NanoARG uses the DeepARG-LS model, a novel deep learning approach developed by Arango-Argoty et al. [25] to detect and quantify ARGs within the regions. A fundamental advantage of the DeepARG model is its ability to recognize ARG-like sequences without requiring high sequence identity cutoffs, which is especially useful for nanopore sequences with high sequencing error rates. The DeepARG-LS model is applied with permissive parameters, specifically, an identity cutoff of 25%, a coverage of 40%, and a probability of 0.5, to predict that a region corresponds to an ARG.

Abundance of ARG classes and groups is estimated by the copy number of ARGs. To enable comparison of ARG abundance across samples, analogous to the approach described by Ma et al. [207], the copy number of ARGs is normalized to the total gigabase pairs (Gbp) of the sample to obtain the relative ARG abundances:

$A_i = \frac{C_i}{C_g}$ (1),

where C_i corresponds to the total count of ARG i (copies of the ARG) and C_g corresponds to the size of the data set in Gbp, that is, $Cg = \Gamma/\mu_g$, where Γ is the total number of nucleotides in the library and $\mu_g = 1\ x\ 10^9$ corresponds to 1 Gbp.

5.2.6 MRG Module

To annotate MRGs, NanoARG queries the BacMet database [210]. Following clustering of the local best hits to identify putative regions of interest (**Figure 5.4**), NanoARG identifies and

categorizes clusters to MRGs according to their best hits. Absolute (copy number) and relative abundances of MRGs are computed using **Equation (1)**.

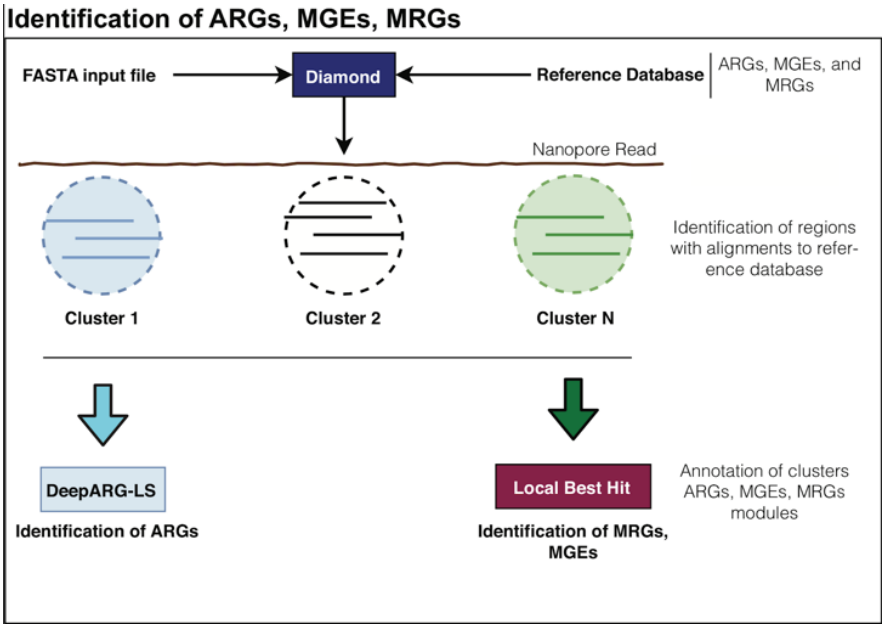

Figure 5.4: Annotation pipelines. **A)** Identification of ARGs: Input nanopore reads are aligned to the DeepARG database using DIAMOND. Alignments are clustered based on their location and annotations are performed using the DeepARG-LS model. **B)** Local Best Hit Approach: Identification of the functional genes within the nanopore reads. Alignments are clustered based on their location and the best hit for each cluster is selected. Resulting alignments are filtered out based on sequence alignment quality.

5.2.7 MGE Database and Annotation Module

MGEs were extracted using the non-redundant database from National Center for Biotechnology Information (NCBI) by using a keyword search [92]. Thus, genes related to any of the following keywords — transposase, transposon, integrase, integron, and recombinase — were labeled as associated MGEs. In addition, a set of integrases and class 1 integrons (*Int*I1) were added from the integron-integrase (I-VIP) database [211]. All sequences were clustered using CD-HIT [212] with an identity of 90%. The resulting MGE database consists of 227,640 genes. Similar to the annotation strategy adopted for MRGs, nanopore reads are annotated using the MGE database and relative abundance of MGEs is computed using **Equation (1)**.

5.2.8 Taxonomic Annotation Module

Nanopore reads are classified according to taxonomic lineage using Centrifuge [213], a fast and accurate metagenomic classifier that uses the Burrows-Wheeler transform (BWT) and FM-index. Centrifuge is executed with default parameters (--min-hitlen 25 -f -k 50). Taxonomic relative abundance is estimated by Centrifuge using an expectation maximization (EM) algorithm similar to the one used in Cufflinks [214] and Sailfish [215]. This allows the abundance estimation to be sensitive to genomes that share nearly identical genomic regions. Therefore, each nanopore read is assigned to a particular taxonomic lineage. In addition, nanopore reads not successfully processed by Centrifuge are labeled as unknown.

5.2.9 Co-occurrence of ARGs, MGEs, and MRGs

To support users in exploring the co-occurrence of ARGs, MGEs, and MRGs in nanopore data sets, NanoARG reports all reads that contain at least one ARG, along with its neighboring genes. This data is presented in a tabular format, where each entry contains the start position, end position, gene coverage, percent identity, e-value, strand (forward or reverse), and taxon corresponding to each read. Furthermore, NanoARG provides a gene map that depicts the gene arrangement, which is useful for visualizing the gene's co-occurrence and context. Overall co-occurrence patterns are depicted as a network, where nodes represent genes, node sizes represent the number of occurrences, edges between nodes represent genes' co-occurrence, and edge thickness depicts the number of times the co-occurrence pattern is observed in the data set. Links among nodes are added according to their co-occurrence among the nanopore reads. The network is rendered using cytoscape.js [216].

5.2.10 World Health Organization Priority Pathogens

The WHO listed a set of pathogens that are of particular interest with respect to the spread of antimicrobial resistance [217]. This list consists of three priority tiers, namely, critical, high, and medium, as described in **Table 2**. Similarly, the ESKAPE database houses multidrug bacterial pathogens that are critical to human health [218]. These two resources are employed by NanoARG to identify the potential presence of critical pathogens in the nanopore sample. Briefly, nanopore reads are matched against sequences available for critical pathogens by examining the NCBI taxonomic identifier downloaded from the NCBI taxonomy Website. Note that NanoARG refers to these hits as "potential" pathogens because the presence of true pathogens cannot be confirmed without higher resolution methods, such as whole genome sequencing and viability confirmation.

Type	Species	Antibiotic
Critical	*Acinetobacter baumannii*	carbapenem
	Pseudomonas aeruginosa	carbapenem
	Enterobacteriaceae	carbapenem, ESBL-producing
High	*Enterococcus faecium*	vancomycin
	Staphylococcus aureus	methicillin, vancomycin
	Helicobacter pylori	clarithromycin
	Campylobacter spp.	fluoroquinolone
	Salmonellae	fluoroquinolone
	Neisseria gonorrhoeae	cephalosporin, fluoroquinolone
Medium	*Streptococcus pneumoniae*	penicillin
	Haemophilus influenzae	ampicillin
	Shigella spp.	fluoroquinolone

Table 5.2: Twelve species of pathogenic bacteria prioritized by the World Health Organization (WHO) as representing substantial antibiotic resistance concern. WHO classification is based on the three categories according to the impact on human health and need for new antibiotic treatments.

5.2.11 Application of NanoARG to Nanopore Sequencing Data sets

To demonstrate NanoARG's capability for profiling ARGs in the context of other relevant genes, four DNA extracts obtained from the sewage and activated sludge of three different wastewater treatment plants (WWTPs) were sequenced using the MinION nanopore sequencing platform and analyzed together with four publicly-available nanopore metagenomic data sets and analyzed using NanoARG (see **Table 5.3**).

Samples	Biome	Sample labels	Number of Reads	Reference	Type of Sample
Hong Kong Activated Sludge	Wastewater	HK_AS	3,307,368	This study	complex microbial community
Hong Kong Influent	Wastewater	HK_INF	2,724,813	This study	complex microbial community
Switzerland Influent	Wastewater	CHE_INF	687,835	This study	complex microbial community
India Activated Sludge	Wastewater	IND_INF	1,925,639	This study	complex microbial community
Artic Glacier Extreme Metagenome	Glacier	GEM	344,966	Edwards, 2016	complex microbial community
Heavily infected Urine	Human associated	HIU	36,510	Schmidt, 2017	enriched microbial community
Hospital Fecal Sample	Human associated	HFS	67,658	van der Helm, 2017	enriched microbial community
Lettuce Spiked *Salmonella*	Plant surface	LSS	211,806	Hyeon, 2018	enriched microbial community

Table 5.3: Sample collection, metadata and total number of reads for all validation samples.

5.2.12 Nanopore Sequencing of WWTP Samples

Four WWTP samples (two influent sewage, and two activated sludge) were collected from three WWTPs located in Hong Kong (HK_INF and HK_AS), Switzerland (CHE_INF), and India (IND_AS). Samples were preserved, transported, and subjected to DNA extraction using a Fast DNA SPIN Kit for Soil (MP Biomedicals) as described in Li et al. [219]. DNA was purified with the Genomic DNA Clean & Concentrator kit (Zymo Research, Irvine, CA) and its concentration was quantified with the Qubit dsDNA HS Assay Kit (Thermo Fisher Scientific). DNA for each sample was pooled from triplicate extractions with equal mass. Pooled DNA was further purified and concentrated to meet the quality and quantity requirement for library preparation. The purity of DNA was then checked using a NanoPhotometer Pearl (Implen, Westlake Village, CA) via the two ratios of A260/280 and A230/260. Each DNA sample (1000 ng) was prepared individually for sequencing using the 1D Native Barcoding Genomic DNA kit (with EXP NBD103 & SQK-LSK108; Oxford Nanopore Technology) following the manufacturer's protocol. Each sample was sequenced with a R9.4 flow cell for 24-48 hours without local base calling. Sequence reads were base called using Albacore (v 1.2.4).

5.3 Results and Discussion

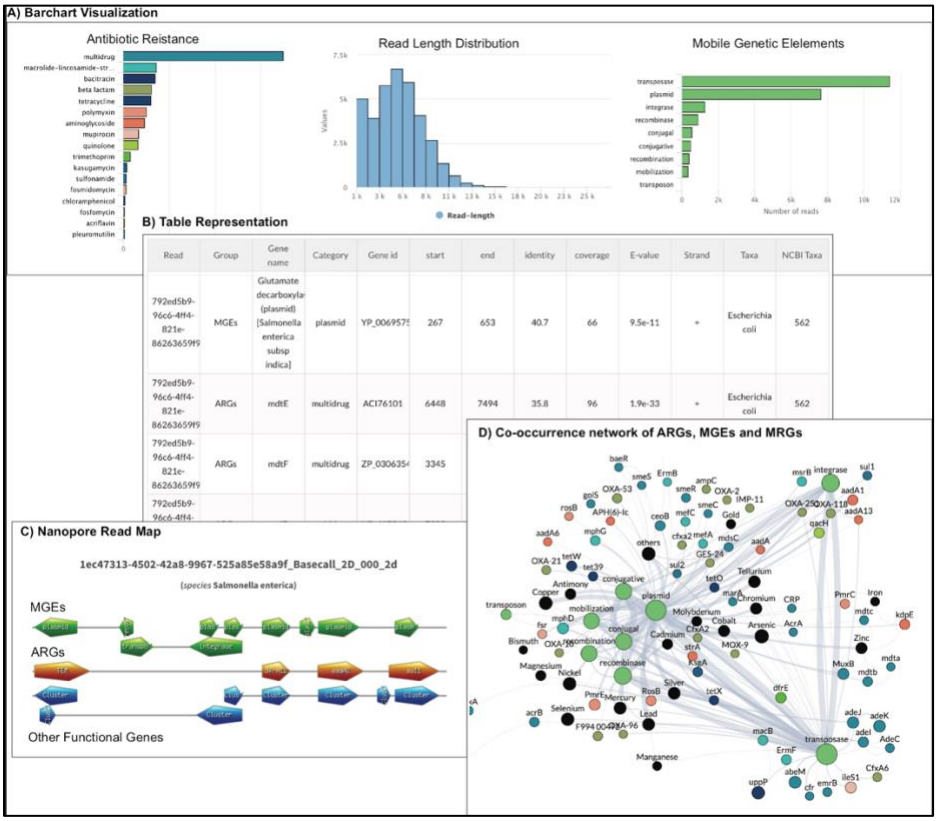

Figure 5.5: Visualization of NanoARG report. **A)** Absolute abundances (read counts) are shown as barcharts as well as read length distribution and taxonomic counts. **B)** Tabular data: Results are also shown in tables containing all the relevant information for each annotation (e.g., E-value, coverage, identity, strand (forward, reverse), and taxonomy, and group). **C)** Nanopore Read Map: This visualization organizes the gene matches in a linear format showing the co-occurrence patterns for each nanopore read with at least one ARG. **D)** Co-occurrence Network of ARGs, MGEs, MRGs: This interactive visualization allows users to drag and drop nodes to visualize the co-occurrence patterns in the sample.

NanoARG is an online computational resource designed to process long DNA sequences for the purposes of annotating co-location of ARGs, MGEs, and MRGs, and to identify their taxonomic hosts. Publication-ready figures and tables derived from these annotations can be directly produced, thus facilitating various dimensions of environmental monitoring and sample comparison.

5.3.1 Visualization and Data Download

The NanoARG service provides a range of visualization options; including bar charts (**Figure 5. 5A**), tables (**Figure 5. 5B**), gene mapping charts (**Figure 5. 5C**), and co-occurrence networks (**Figure 5. 5D**) that display individual and combined analyses of ARGs, MGEs, and MRGs.

Results can be downloaded from the tables and configured to include all data, without any filtering. This enables users to deploy their own filtering criteria and customize analyses.

5.3.2 Effect of Error Correction in the Detection of ARGs

To examine the effect of error correction in the detection of ARGs by NanoARG, HFS sample nanopore sequences were analyzed with and without error correction. The complete data set (Library B) was downloaded from the poreFUME repository, including the raw nanopore reads (HFS-raw) along with the corrected reads after the poreFUME pipeline (HFS-poreFUME). In addition, the raw nanopore reads were also corrected (HFS-CANU) using the correction module from the CANU assembler. These three data sets were submitted to the NanoARG pipeline for annotation.

Figure 5.6A shows that the alignment bit score of all the ARGs is increased after read correction by both CANU and poreFUME algorithms compared to the raw uncorrected reads. Here "high coverage" ARGs are those ARGs with ≥10 read hits whereas "low coverage" ARGs have fewer hits. For the CANU-correct algorithm, the bit scores of "high coverage" ARGs such as CTX-M, TEM, *aad*A, aac(6')-I, and *erm*B ARGs were significantly improved (**Figure 5.6B-D**) compared to the raw reads. Similarly, the bit scores of "low coverage" ARGs, such as CARB, *erm*F, *fos*A3, *mel*, and *tet*Q, also showed an improvement after read correction (**Figure 5.6E-G**).

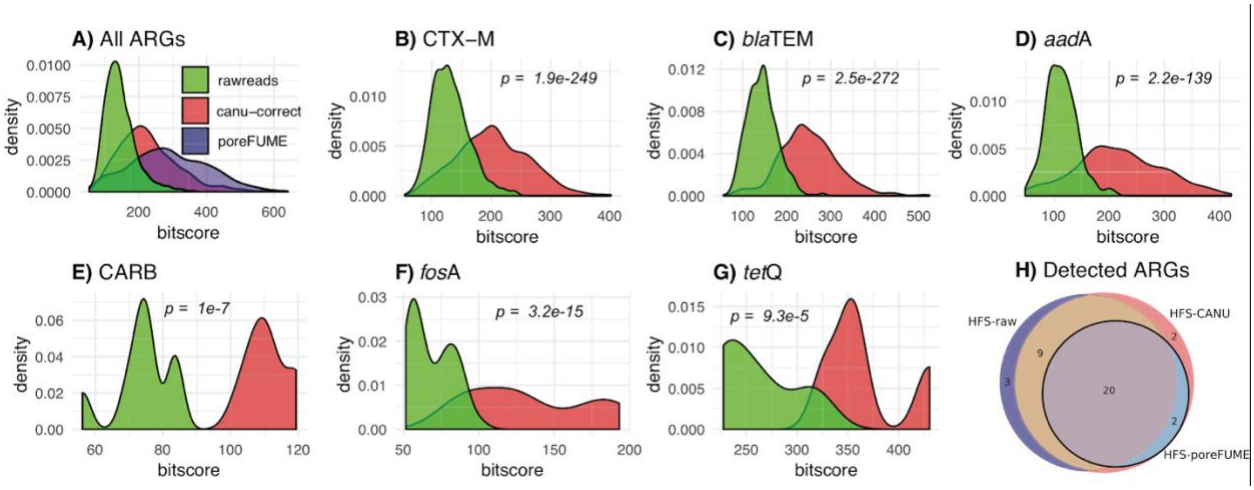

Figure 5.6: Comparison of error correction approach applied to a functional metagenomic sample. Comparison against raw reads and error corrected reads using CANU correct and poreFUME. P-values were computed between the different distributions using a T-Test. **A)** Bit score distribution of all ARG alignments. **B-D**) Comparison between raw and corrected reads using CANU correct for ARGs with high depth. **E-F)** Bit score distribution for raw and corrected reads for low depth ARGs. **G)** Venn diagram showing discovered ARGs by raw and corrected reads by CANU and poreFUME. *Because poreFUME could not run due to library dependency errors, **Figure 6B-G** contain the transition of quality distribution when comparing CANU-correct and the raw reads

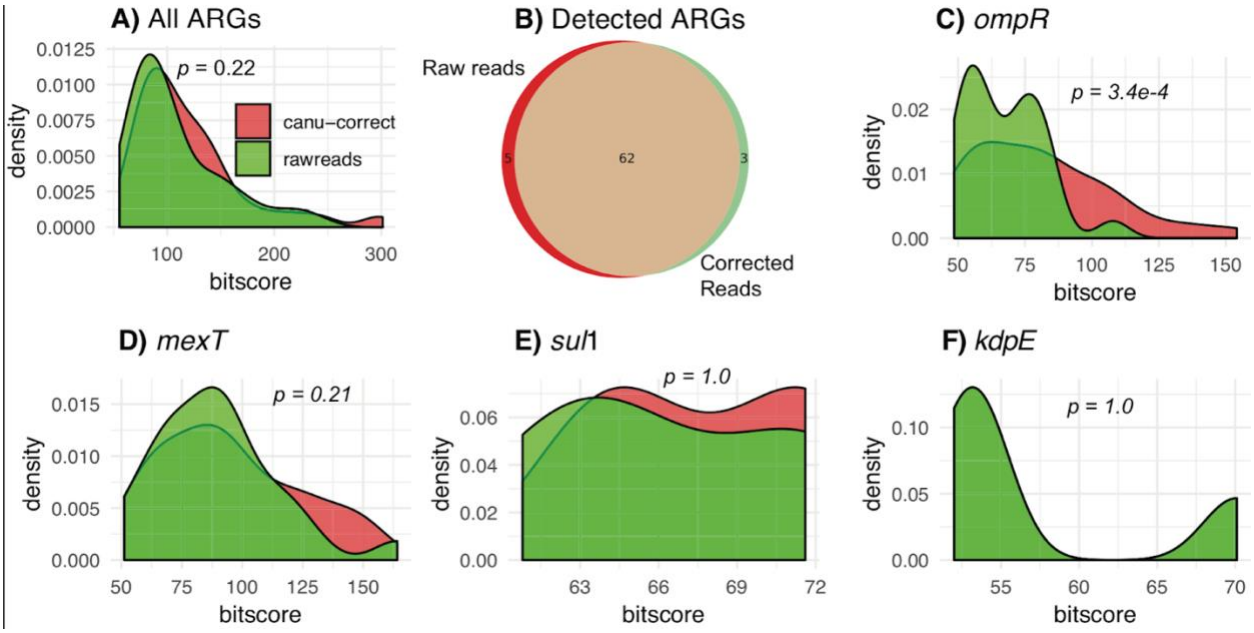

Figure 5.7: Effect of error correction on analysis of an environmental sample (WWTP influent). **A)** Bit score distribution for all ARGs detected by NanoARG using the raw and CANU corrected reads. **B)** Venn diagram showing the intersection of detected ARGs from raw and corrected reads. **C-D)** Examples of the effect of correction in individual ARGs with high number of hits comparing the raw and corrected reads. **E-F)** Effect of correction in ARGs with few hits from the raw and corrected data sets.

Figure 5.6H depicts the intersection of ARG annotation by NanoARG among the three data sets (HFS-raw, HFS-CANU, and HFS-poreFUME). ARGs with a minimum coverage of 80% and an identity greater than 30% were used for this comparison. Altogether, 22 unique ARGs were detected in the HFS-poreFUME data set, 32 in the HFS-raw data set, and 33 in the HFS-CANU data set. Out of the 22 ARGs detected in HFS-poreFUME, two ARGs (*abe*S and CARB) were not identified in the HFS-raw sample. Further examination revealed that these genes were actually detected in the HFS-raw data set, but were removed after applying the filtering criteria described above. These two genes were also detected following the error correction step (HFS-CANU); indeed, all ARGs that were detected in HSF-poreFUME were also identified after applying the error correction algorithm with CANU. Although there were three uniquely identified ARGs in the HFS-raw data set (FosC2, LuxR, and *emr*K) and four uniquely identified ARGs after CANU correction (CARB, OXY, *abe*S, and *van*H), the results show that there was a transition in the annotation from raw to corrected reads. Thus, reads were reassigned to other ARGs with higher alignment and classification scores. For instance, raw reads containing the CTX-M gene were reassigned to the OXY gene with higher alignment scores in the HFS-CANU data set. The CARB gene was detected in both HFS-raw and HFS-CANU data sets. However, the coverage of this gene in the HFS-raw data set was below the 80% cutoff used for the analysis and therefore was removed from the list, whereas it was successfully detected on the HFS-CANU data set, showing an

improvement in the alignment coverage. The reads containing the *fos*C2 gene in the HFS-raw sample were reassigned to the *fos*A gene in the HFS-CANU data set with higher alignment bit scores (73 to 126.3, respectively). Interestingly, the *van*H gene was detected exclusively on the HFS-CANU data set. These results show that the correction step enhances the detection of ARGs in MinION nanopore sequencing samples.

To validate the read correction approach on a more complex sample than HFS, one WWTP sample (CHE_INF) subjected to direct shotgun metagenomic sequencing was selected for further validation of the effect of the error correction algorithm. The metagenomic data set was processed using CANU correct and submitted along with the raw data sets to NanoARG for annotation. poreFUME was not performed for this analysis because of dependency errors present during execution of the pipeline. **Figure 5.7A** shows the bit score distribution of the ARG alignments for both raw and corrected reads. Notably, the correction algorithm did not significantly improve (p=0.22) the overall ARGs bit score of the alignments for this more complex sample. **Figure 5.7B** shows the intersection of the detected ARGs for the WWTP sample with and without correction. Among the majority of ARGs detected by NanoARG in both raw and corrected reads, three were detected after read correction, but not in the raw reads (OKP-A, *bcr*A, and *otr*C). To observe the effect of coverage depth for each ARG, a closer examination of the individual ARGs did not indicate enhancement of alignment scores for genes with the greatest number of hits, such as *omp*R and *mex*T (**Figure 5.7C-D**), or for ARGs with low numbers of hits, such as *sul*1 and *kdp*E (**Figure 5.7E-F)**. Because the overlap between the ARGs detected in the raw and corrected reads is greater than 95% (**Figure 5.7B**), NanoARG was not further configured to perform error correction and allows users to decide whether to upload raw reads, corrected reads, or assembled contigs. Users can find information about error correction and how to perform it using CANU on the NanoARG website.

To check the effect of time and consistency for the discovery of ARGs in nanopore samples using NanoARG, several data sets from the LSS sample were analyzed, including comparison of nanopore- versus Illumina-derived and whole-genome versus shot-gun data sets. Specifically, a study of lettuce spiked with *Salmonella enterica* (LSS) consisted of the following data sets: LSS-WGS (whole-genome sequencing), LSS-M (shotgun metagenomics), LSS-1.5hN (nanopore sequencing after 1.5 hours) and LSS-48hN (nanopore sequencing after 48 hours). To facilitate comparison, the short reads from LSS_WGS and LSS-M were first assembled using spades [187] with default parameters. Assembled scaffolds were subsequently submitted to NanoARG for annotation. The MinION nanopore sequencing libraries were first error corrected using CANU correct algorithm prior to submitting to NanoARG. To evaluate the accuracy of ARG detection, alignments were compared relative to a threshold identity cut-off greater than 80% and an alignment coverage greater than 90% from the LSS-WGS sample. A total of 28 ARGs passed these filtering criteria, and further analyses were benchmarked against these 28 ARGs, assuming a high level of confidence in their identity. Out of these 28 ARGs, two genes (*mdt*B and *bcr*) were not

detected in the Illumina shotgun metagenomic data set (LSS-M). When comparing the 28 benchmark ARG set against the 1.5h nanopore LSS-1.5hN sample, only four ARGs were detected (aac(6')-I, *mdf*A, *mdt*G, and *mdt*M) in the nanopore data set. This result suggests that although nanopore sequencing offers a real-time alternative, the detection of specific ARGs would still require several hours. Still, when examining the 48h nanopore sample (LSS-15hN), 25 out of the 28 benchmark ARGs were discovered. Interestingly, *mdt*B, one of the three undiscovered benchmark ARGs (*mdt*A, *mdt*B, and *mdt*C) from the LSS-48hN was not found by either the Illumina shotgun metagenomics sample (LSS-M) or the nanopore samples. These three ARGs were noted to pertain to the same antibiotic resistance mechanism. Overall, this analysis demonstrates general consistency of detection of ARGs in Illumina and nanopore sequencing libraries using NanoARG.

5.3.3 Application of NanoARG to Nanopore Sequencing Data

NanoARG provides users with a master table that contains the absolute and relative abundances of ARGs, MRGs, MGEs, and taxonomy annotations for each sample under a particular project. Relative abundances are computed as described in Equation 1. Key attributes of this table are summarized in the following subsections, using eight nanopore sequencing data sets as examples.

5.3.4 ARG Abundance

WWTP samples contained the greatest number of reads (> 687,835), whereas human-derived samples (HIU, HFS) were comprised of far fewer reads (<67,658) (See **Table 5.3** for details). **Figure 5.8** shows relative abundances of ARGs in the eight data sets. HFS contained the highest relative ARG abundance, likely due to the sample preparation approach that intentionally targeted genomic content associated with antibiotic resistance [196]. Comparatively, the direct shotgun metagenomic sequenced environmental samples had much lower ARG relative abundance. Among the WWTP samples, HK Influent and HK Effluent ranked the greatest in terms of relative abundance of ARGs.

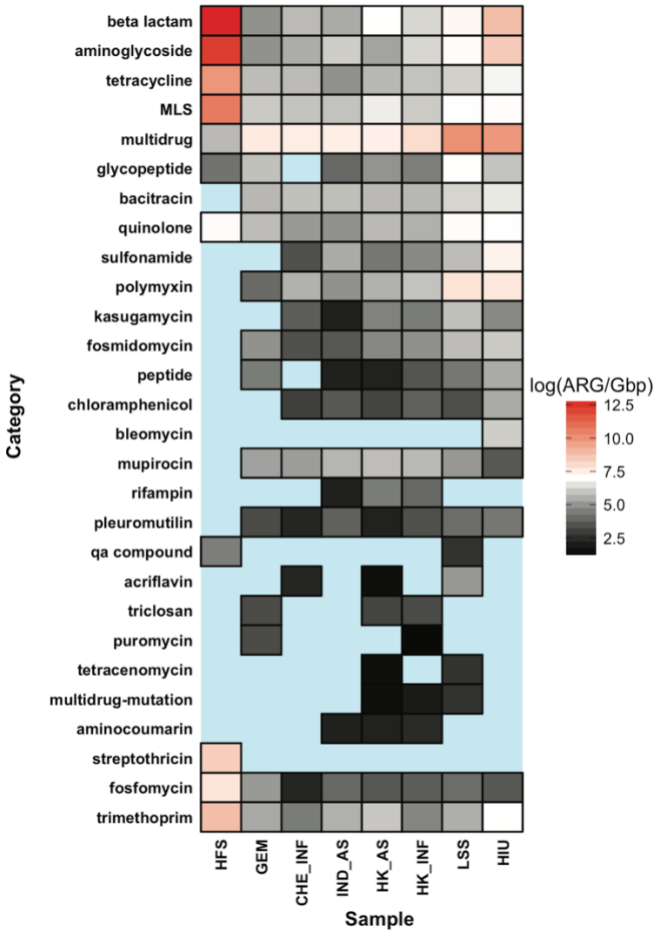

Figure 5.8: Total relative abundance of ARGs from the four validation samples, each representing distinct biomes. WWTP samples are zoomed in to aid discrimination of ARG content.

In considering specific subcategories of resistance, the HFS sample contained the greatest relative abundances of beta lactamase, aminoglycoside, tetracycline, trimethoprim, fosfomycin, streptothricin, quinolone, and MLS antibiotic classes (**Figure 5.8**). Note that these categories were also prominent in the WWTP and glacier samples but to a lesser extent than in HIU and the LSS samples. In addition, although the multidrug category is highly abundant in HIU and LSS, it has the lowest relative abundance in the HFS sample. Interestingly, although HFS contained the highest relative abundance of total ARGs, the WWTP samples had the highest diversity of antibiotic resistance classes measured as the number uniquely identified antibiotic types (**Figure 5.8**). For instance, *sul1* was one of the most prevalent ARGs detected in WWTP samples [220]. However, *sul*1 was not found in the GEM sample. This is consistent with the *sul*1 gene being an anthropogenic (result of human activity) marker of antibiotic resistance [36, 221]. Similarly, GEM has lower diversity of beta lactamase genes (4 beta lactamase ARGs) than the WWTP environments (25-237 beta lactamase ARGs). ARGs from acriflavine, triclosan, aminocoumarin, tetracenomycin, rifampin, and puromycin antibiotic classes were only detected in the WWTP and

LSS samples. HK_INF and HK_AS indicated the highest relative abundance of ARGs compared to IND_AS and CHE_INF (**Figure 5.9A**). Particularly, the HK_AS sample showed a decrease compared to HK_INF in the abundance of multidrug and aminoglycoside resistance genes but an increase in the beta-lactamase, *MLS*, and trimethoprim antibiotic types.

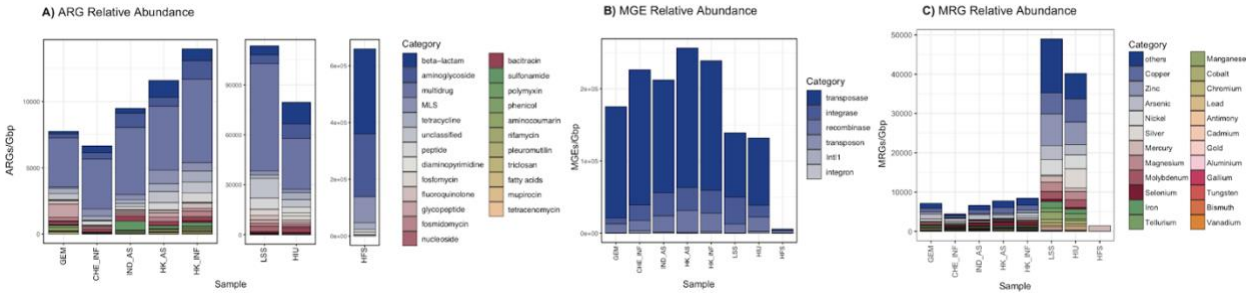

Figure 5.9: Relative abundance of antibiotic resistance classes for all biomes. Each point corresponds to a particular antibiotic, biome pair. Size and color represent the copy number of ARGs divided by 1 Gbp on a logarithmic scale.

5.3.5 MGE Abundance

For its MGE reference database, NanoARG curates a collection of genes related to mobility, including transposases, integrases, recombinases, and integrons, in addition to a curated database for the class 1 integron *intI*1 [211]. Transposases are the prominent MGEs across all samples (**Figure 5.9B**). Interestingly, the HFS sample shows the lowest relative abundance of MGEs. The *Salmonella*-spiked sample along with the heavily infected urine sample, shows a lower MGE relative abundance compared to the environmental samples (WWTP and glacier). Note that the glacier sample, GEM, contained the lowest MGE abundance compared to the WWTP samples. Interestingly, GEM also has the lowest diversity of MGEs (integrases, transposases, and other MGEs) when compared to other samples. This suggests that there may be a lesser degree of HGT in relatively pristine environments, such as glaciers, than in heavily anthropogenically-influenced environments, such as WWTPs. Further, the class 1 integron *int*I1, which has been proposed as an indicator of anthropogenic sources of antibiotic resistance [162], is also consistent with this trend. The integron *intI*1 was detected in all samples, except in the GEM sample, likely because glaciers are under less anthropogenic pressure such as antibiotics usage or wastewater discharges. In addition, *int*I1 in the HIU sample was ranked to be the highest in relative abundance, which is expected given the clinical context of this sample.

5.3.6 MRG Abundance

MRG profiles were markedly distinct when comparing trends among samples relative to ARG profiles. The HFS sample has the lowest number of MRGs, with only *merP* and *merT*, part of the mercury transport mechanism [210] (**Figure 5.9C**). In contrast, LSS and HIU samples carried the

highest relative abundance of MRGs. The lack of MRGs in HFS could be the result of the sample preparation or lack of direct selection pressures relevant to MRGs. Notably, the HFS sample carried high beta lactamase, aminoglycoside, tetracycline, and MLS abundance, contrasting with low multidrug relative abundance. WWTP samples showed a different trend compared to MGEs and ARGs. The CHE_INF sample has the lowest relative abundance of MRGs compared to other WWTP samples. Although CHE_INF has also the lowest ARG relative abundance, its MRG abundance was less than half that of any other WWTP sample, suggesting that the CHE_INF sample had less exposure to heavy metal compounds.

5.3.7 Taxonomy Profile

The HIU sample indicated *Escherichia coli* as the dominant species, which is expected given that a strain of MDR *E. coli* had been spiked into the urine prior to DNA extraction and analysis [197] (see **Figure 5.10D**). Similarly, *Salmonella enterica* was found to be most abundant in the food sample metagenome (LSS), consistent with known *S. enterica* contamination of this sample [222]. The results of the HFS sample provide the opportunity to evaluate how the NanoARG taxonomic profiling performs with distinct approaches of library construction. Specifically, the HFS study [196] was designed to maximize chances of ARG detection, not to profile taxonomy. Thus, it makes sense that the nanopore taxonomy profile consists largely of *E. coli*, the expression host, and other taxa that likely represent the original source of the transformed ARGs, e.g., *Klebsiella pneumoniae*, *Serratia marcescens*, and *Enterococcus faecium* (see **Figure 5.10B**). A surprise with respect to the species distribution in the WWTP samples was substantial detection of human DNA (see **Figure 5.10E-H**). In one of the influent samples, *Homo sapiens* was the dominant species (see **Figure 5.10F-G**). This host DNA is also observed to a lesser extent in the spiked samples (LSS, HIU). Surprisingly, the HFS sample did not contain detectable human DNA, suggesting that the technique employed in this study to specifically enrich ARGs during library preparation was successful for enriching ARGs.

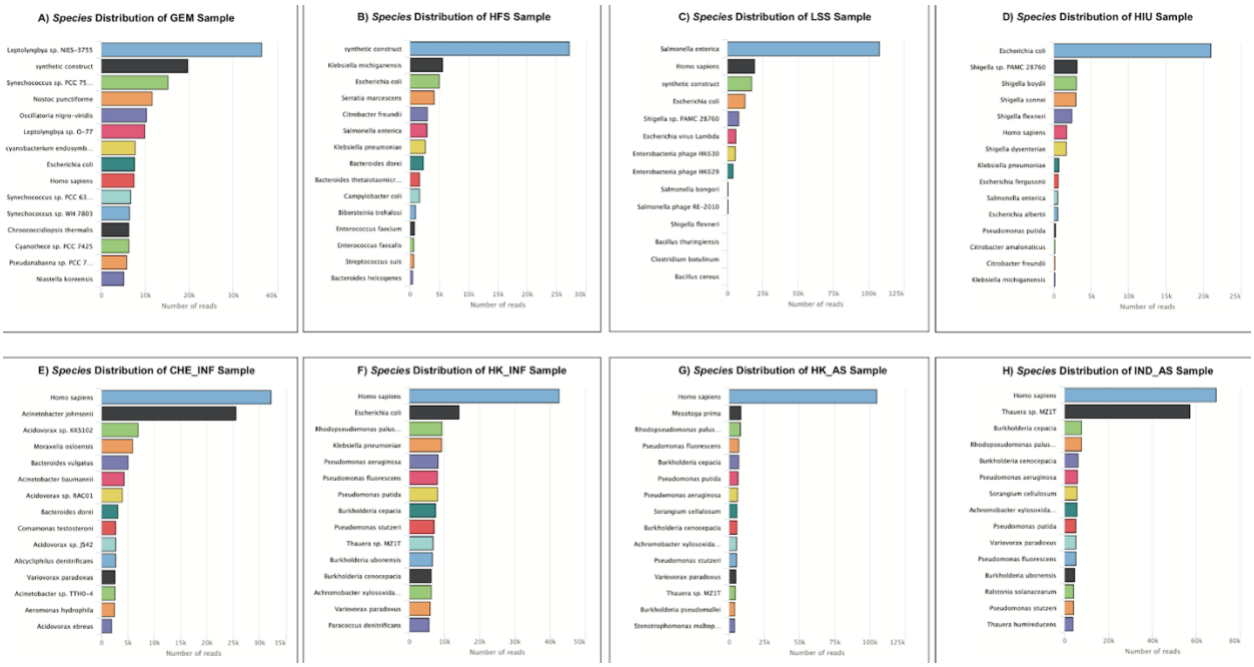

Figure 5.10: Relative abundance computed as copy of genes per 1Gpb of **A)** Antibiotic resistance classes, **B)** MGEs, **C)** MRGs.

5.3.8 ARG Neighboring Gene Analysis

Long nanopore sequences allow the inspection of ARG linkage patterns and the context of neighboring genes. For instance, **Figure 5.11** shows that the sulfonamide ARG *sul1* appears in different contexts depending on the WWTP sample and its host. Also *sul1* is almost exclusively co-located together with *integrase* or *recombinase*, along with genes that have been found in plasmids, consistent with the theory that *sul1* is an indicator of HGT. *sul1* was commonly observed together with an *integrase* or *recombinase* gene, followed by an aminoglycoside (*aadA*) gene, a determinant of quaternary ammonium compound resistance gene (*qacE*), which is also consistent with prevailing understanding of typical class 1 integron operon architecture [223]. Interestingly, this pattern seems to be modified in *E. coli* from two of the activated sludge environments (HK and IND), where the *integrase* or *recombinase* and the *aadA* region is interrupted by the insertion of a beta lactamase (*OXA*) gene. This linkage pattern differs from the one observed in *Hydrogenophaga* sp. *PBC* from the CHE influent. This *sul1* gene analysis is only one example of how NanoARG facilitates the inspection of colocation of ARG together with other genes of interest on the same DNA strand. Users can dig deeper to identify other patterns of interest and discover signals of ARG dissemination. The full co-occurrence result can be downloaded for further analysis.

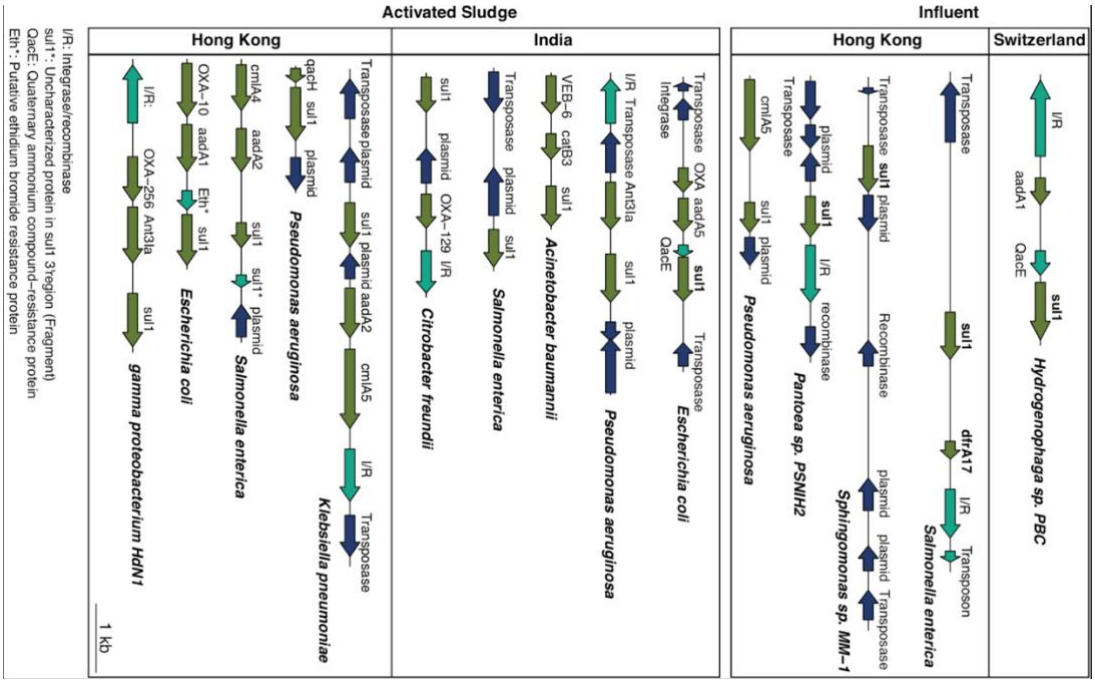

Figure 5.11: Taxonomic distribution of validation samples representing distinct biomes. A) Phylum distribution of WWTP samples. B-H) bar plots with the total number of reads classified at the *Species* taxonomy level for each validation sample.

Figure 5.12 shows the ARG co-occurrence network for all samples. ARGs are linked if they co-occur within the same read and ARGs that appear only once are not shown. GEM, with a small number of ARGs belonging to only multidrug and trimethoprim classes, has no ARG co-occurrence (**Figure 5.12A**). The WWTP samples show a common pattern of co-occurrence between beta-lactamases and aminoglycoside genes, indicating the high potential of these genes to be carried simultaneously. The HFS sample was dominated by aminoglycosides and beta lactamase genes, whereas LSS was dominated by multidrug genes and glycopeptides genes.

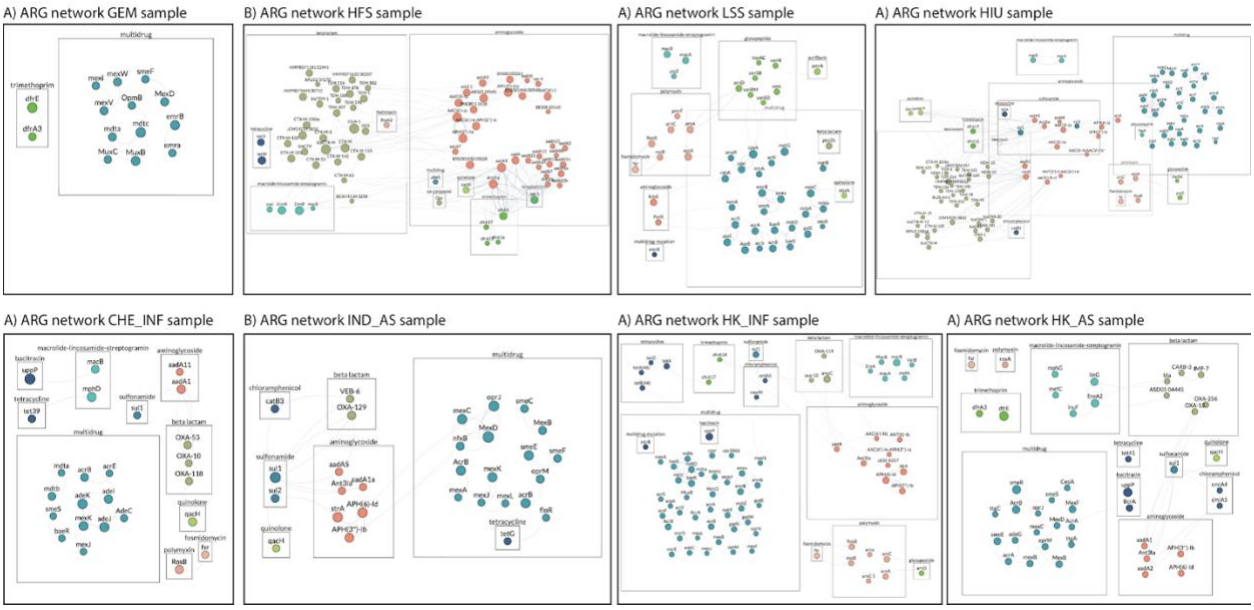

Figure 5.12: ARG patterns and contexts. Different patterns of ARGs for the WWTP samples (influent and activated sludge). I/R: integrase/recombinase, *sul*1*: Uncharacterized protein in *sul*13' region. *aqcE:* Quaternary ammonium compound-resistance protein, Eth*: Putative ethidium bromide resistance protein.

5.3.9 Critical Bacterial Pathogens

Another important feature of NanoARG is the ability to putatively identify pathogens based on similarity to available DNA sequences in databases (see **Table 5.2**) and to assess their association with ARGs. For instance, DNA sequences corresponding to two of the three pathogens classified as having "critical importance" by the World Health Organization, *Acinetobacter baumannii* and *Pseudomonas aeruginosa*, were detected in all WWTP samples (see **Table 5.4**). In contrast, DNA sequences corresponding to *Enterobacteriaceae* (carbapenem-resistant pathogen) were only detected in one WWTP sample (HK_INF). In addition, the HK_INF sample contained DNA sequences with high similarity to *Neisseria gonorrhoeae*. *Pseudomonas aeruginosa* was estimated to be the most abundant pathogen in the "critical" category across all samples and is particularly abundant in the IND_AS sample. No pathogen-like DNA sequences were found in the GEM sample, consistent with our expectation of a relative lack of anthropogenic influence. NanoARG clearly holds promise as a tool for screening for the potential presence of pathogens pertaining to various levels of priority. Further, the potential for putative pathogens to carry ARGs, MRGs, and MGEs can be readily assessed. However, it is important to emphasize that further culture-based and molecular-based analysis are required as follow up to confirm the presence of viable and virulent pathogens.

Pathogen-like sequences	**CHE_INF**	**IND_INF**	**HK_INF**	**HK_AS**	**GEM**
Acinetobacter baumannii	**3 (4)**	**4 (6)**	**12 (16)**	**6 (6)**	0 (0)
Pseudomonas aeruginosa	**7 (6)**	**58 (74)**	**12 (13)**	**7 (11)**	0 (0)
Enterobacteriaceae	0 (0)	0 (0)	**2 (2)**	0 (0)	0 (0)
Enterococcus faecium	0 (0)	0 (0)	0 (0)	**1 (1)**	0 (0)
Staphylococcus aureus	0 (0)	0 (0)	0 (0)	**1 (1)**	0 (0)
Helicobacter pylori	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)
Campylobacter spp	0 (0)	0 (0)	0 (0)	**1 (1)**	0 (0)
Salmonellae	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)
Neisseria gonorrhoeae	0 (0)	0 (0)	**3 (6)**	0 (0)	0 (0)
Streptococcus pneumoniae	0 (0)	0 (0)	**1 (1)**	0 (0)	0 (0)
Haemophilus influenzae	0 (0)	0 (0)	**1 (1)**	0 (0)	0 (0)
Shigella spp	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)

Table 5.4: List of critically-important bacterial pathogens putatively identified in the WWTP samples. * Notation: Number of reads (number of ARGs).

5.3.10 NanoARG Usage Recommendation

Note that the various analyses provided by NanoARG are not restricted to nanopore sequencing reads. In fact, NanoARG can be applied to any set of long DNA sequences (>1000bp long). For instance, sequences from different technologies such as PacBio long-read sequencing or assembled contigs from short sequencing reads can be directly processed in NanoARG. Depending on specific research needs, different studies may have different requirements, e.g., some require more stringent criteria, whereas others less. Thus, to allow for flexibility and customization, NanoARG provides users results produced by relaxed annotation parameters so that they can filter the results further to meet their specific needs.

5.4 Conclusions

NanoARG is a public Web service dedicated to the analysis of ARGs from nanopore MinION metagenomes and is the first, to our knowledge, configured for analysis of environmental samples. While the platform was specifically developed for the analysis of environmental metagenomes generated from nanopore sequencing technologies, here we demonstrate that it also has broad

potential for other types of data sets. As validated here using a combination of publicly-available and in-house DNA sequence libraries, NanoARG can be used to profile ARGs in any biome while also providing context of other co-located genes, such as MGEs, MRGs, and taxonomic markers. NanoARG provides a user-friendly interface for analysis of any set of long DNA sequences (including assembled contigs), facilitating data processing, analysis, and visualization. Unlike other services dedicated exclusively to antimicrobial resistance (e.g., WIMP- https://nanoporetech.com/), NanoARG offers analysis of MRGs and MGEs while also enabling taxonomic annotation, identification of pathogen-like DNA sequences, and network analysis for assessing corresponding co-occurrence patterns. Further, integration with deep-learning based DeepARG facilitates a local strategy for annotating genes from long nanopore reads. Specifically, implementation of permissive parameters allows high flexibility for the detection of homologous genes, which helps overcome high error rates characteristic of nanopore sequences.

CHAPTER 6 SPEEDING UP METAGENOMICS ANNOTATION

A functional profile of metagenomic samples allows the understanding of the role of the microbes in their environment. Such analysis consists of assigning short sequencing reads to a particular functional category. Normally, manually curated databases are used for functional assignment where genes are arranged into different classes. Sequence alignment has been widely used to profile metagenomic samples against curated databases. However, this method is time consuming and requires high computing resources. Although several alignment free methods based on k-mer composition have been developed in recent years, they still require a large amount of memory. In this paper, MetaMLP (Metagenomics Machine Learning Profiler) a machine learning method that encodes sequences as numerical vectors (embeddings) and uses a simple one hidden layer neural network to profile functional categories is proposed. Unlike other methods, MetaMLP enables partial matching by using a reduced alphabet to build sequence embeddings from full and partial k-mers. MetaMLP is able to identify a slightly larger number of reads compared to DIAMOND (one of the fastest sequence alignment method) as well as to perform accurate predictions with 0.99 precision and 0.99 recall. MetaMLP can process 100M reads in around 10 minutes in a laptop computer, which is 50 times faster than DIAMOND. MetaMLP is free for use, and available at https://bitbucket.org/gaarangoa/metamlp/src/master/.

6.1 Introduction

The wide and rapid adoption of next generation sequencing techniques (NGS) such as metagenomics in the analysis of microbial diversity, antibiotic resistance, and other functional profiling analysis creates a gap between scalability and processing efficiency. In other words, large amounts of data require the design of computational tools that are both accurate and fast. Sequence comparison algorithms such as BLAST [224], FASTA [203], HMMER [225], and PSI-BLAST [226], were created with the aim to find correspondence of the sequence distribution in two or more sequences. BLAST is to date the most popular and trusted tool for sequence alignment. However, it is well known that BLAST does not scale well when comparing millions of sequences. The reason is that BLAST uses a computationally demanding strategy consisting of a seed and extend algorithm [227]. Although, sequence alignment is considered the gold standard approach for sequence analysis, there are several cases where this technique can produce dubious results [228]. For instance, alignment-based methods assume that homologous sequences share a certain degree of conservation. Although this assumption is considered to be true when analyzing conserved domains, organisms such as viruses that exhibit a high degree of mutation, challenge this principle. When analyzing short sequences (e.g., Illumina sequencing reads), a high similarity does not always guarantee that the read can be assigned to an unique origin gene [136]. In the opposite case, genes that share less than 30% identity over their full length can potentially belong to the same gene family and perform the same functions [84].

DIAMOND [40], BLAT [229], USEARCH [230], and RAPSearch [231] are alternatives to BLASTX that can run much faster but with some loss of sensitivity. Particularly, the dramatic speed up of DIAMOND (20,000X) is achieved by using a double indexing strategy, spaced seeds (longer seeds where not all positions are used) and a reduced alphabet. In detail, DIAMOND implements a seed and extend algorithm that first indexes both query and reference sequences. Then, the list of seeds in both the query and reference are linearly traversed to determine all the matched seeds with their locations. Finally, seeds are extended by using the Smith-Waterman algorithm [232].

Alignment-free methods have been proposed as an alternative to quantify the sequence similarity without performing any sequence alignment [39, 228]. These methods do not use the seed and extend paradigm. Therefore, their computational complexity is often linear in time and only depends on the query sequence length. In next-generation sequencing, several alignment-free strategies have been developed for different applications, including transcript quantification (kallisto [233], sailfish [215], Salmon [234], RNA-Skim [235]), variant calling (ChimeRScope [236], FastGT [237]), *de novo* genome assembly (minimap [238], MHAP [239]), and the profiling of metagenomics taxonomy by using a k-mer matching approach (Kraken [240], Mash [241], CLARK [242], stringMLST [243]).

The word embedding technique is one of the most successful learning methods applied in natural language processing (NLP), where a word is encoded in a numerical vector. For instance, the Word2vec technique [160] uses a shallow two-layer neural network to train and aggregate word embeddings by using the continuous bag of words (CBOW) approach. Thus, identifying semantic associations between a target word given its context. The concept of using word vectors for representing protein or DNA sequences is not new and has been explored before. For instance, DNA2Vec [244], explores the associations between variable length k-mers to generate an embedding space that proved to correlate with sequence alignment. Yang, et. al. [245] explores the performance of word embeddings for classification of protein functions compared with classical representation techniques. Yang, et. al. demonstrated that k-mer embeddings outperformed other techniques. However, in both studies, embeddings are learned in an unsupervised way. This means that the embeddings are learned first and then the classifier is built by using those embeddings. Here, MetaMLP (Metagenomics Machine Learning Profiler), an alignment-free method that uses word embeddings to represent target protein databases is proposed for the functional profiling of metagenomic samples. The strategy behind MetaMLP relies on a slightly modification of the CBOW model where the “target word” is replaced by the label of the sequence. Thus, MetaMLP builds a supervised embedding representation by using k-mer and fragmented k-mers as context words. Therefore, MetaMLP is a novel strategy that uses a combination of hash indexing, six open reading frame translation, a reduced amino acid alphabet, and an embedding representation to process metagenomic data. In addition, MetaMLP was built up on top of the C++ FastText [43] library and is composed of two main stages: MetaMLP-index

that process protein sequences to build a machine learning model and MetaMLP-classify to annotate reads from metagenomic DNA sequence libraries.

6.2 Materials and Methods

The overall structure of MetaMLP is shown in **Figure 6.1** and consists of two main components: **A)** an indexing stage that process protein reference sequences into a word vector representation to train a classifier and **B)** a prediction stage that process short sequencing reads and classifies them into one to the predefined classes from the reference database.

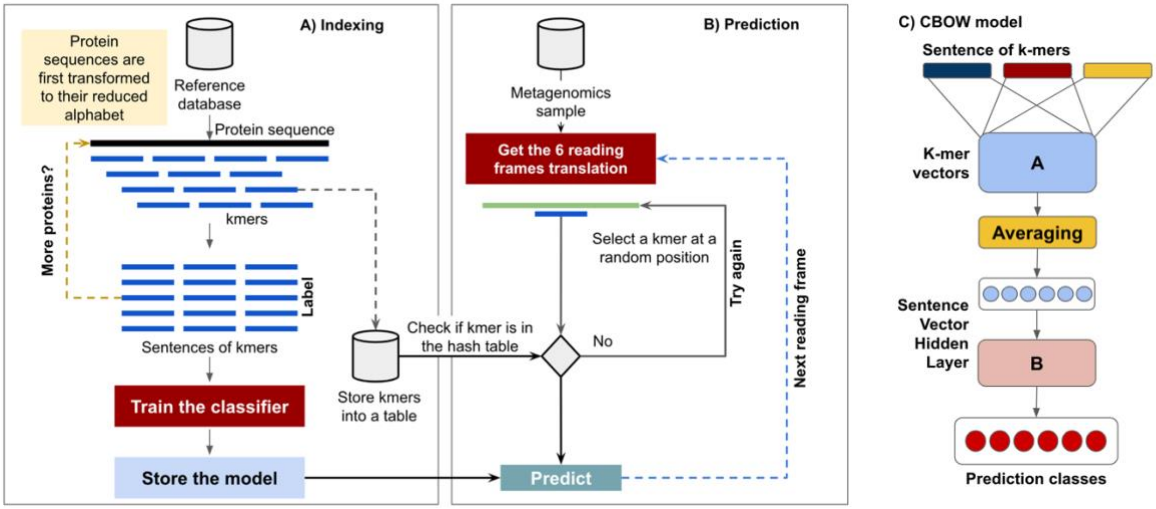

Figure 6.1: Overview of MetaMLP. A) Indexing reference databases where proteins are used to train the machine learning model. B) Once a model is trained, it will be used later to profile short sequencing reads to produce a relative abundance profile and the individual predictions for each read. C) CBOW model used in MetaMLP

6.2.1 Indexing Protein Reference Databases

6.2.1.1 Reference Database Preprocessing

To increase the chances of detecting sequences with mismatches, the reference proteins are first transformed into their equivalent 10 amino acid alphabet version using the murphy.10 alphabet representation used in rapsearch (A [KR] [EDNQ] C G H [ILVM] [FYW] P [ST]) [246]. Then, k-mers of a fixed length are extracted from each protein sequence. However, to consider all k-mers within a sequence, a sliding window of one amino acid is used. Thereafter, a 'sentence' of k-mers is extracted by taking from 3 to 5 consecutive k-mers (see **Figure 6.1A**). At the same time, a table with unique k-mers is built and stored to later be used for filtering sequences that diverge greatly from the reference database during the prediction stage.

Formally, proteins are strings of amino acids arranged in a particular order. Thus, the protein P is represented by $P = \{p_1, \dots, p_i, \dots, p_N\},\ p_i \in \{\mathrm{A, K, R, E, D, N, Q, C, G, H, I, L, V, M, F, Y, W, P, S, T}\}$ where each position corresponds to one of the twenty amino acids found in nature, and N corresponds to the length of the protein sequence. The protein P can be reduced to its simplified 10 amino acids version $R = \{r_1, \dots, r_N\}$ where r_i corresponds to one of the reduced alphabet amino acids. A k-mer $k_{i,l}$ of length l is defined as a substring of size l within the protein sequence R at the position i, such that $k_{i,l} = R[i: i + l] = \{r_i, r_{i+1}, \dots, r_{i+l-1}\}$, where, $1 \leq i \leq N - l + 1$. Finally, the sentence of k-mers s_i, starting at the position i is defined as $s_i = \{k_{i,l}, k_{i+l,l}, \dots, k_{i+(m-1)*l,l}\}$ where m corresponds to the number of k-mers used to build the sentence. Therefore, the protein sequence R can be represented as a set of sentences $\mathbb{S} = \{s_1, s_2, \dots, s_p, \dots, s_{n-m*l-1}\},\ 1 \leq p \leq N - m * l + 1.$

6.2.1.2 Training

MetaMLP uses the FastText [43] implementation of the continuous bag of words (CBOW) technique to learn the semantic relations between protein sequences and their labels by using the protein k-mers (see **Figure 6.1A**). In detail, a protein sequence is represented as a series of k-mer sentences $\mathbb{S}$ (analog to sentence of words in text documents), where, each sentence $s = \{k_1, k_2, \dots, k_j, \dots, k_m\}$ is composed of a set of k-mers. The supervised CBOW model learns the embedding space by looking at the k-mers distribution and the class label. Thus, the first weight matrix A in the classifier comprises the k-mer embeddings $X = \{x_1, \dots, x_j, \dots, x_m\}$, where x_j represents a r-dimensional vector. Then, these k-mer vectors are averaged into the vector $y = \frac{1}{m}\sum_{j=1}^{m} x_j$, that is supplied to a single hidden layer neural network. Then, it is multiplied by a sentence embeddings matrix B to output the probability distribution over the established classes by using a softmax layer f. For a total number of Q k-mer sentences in the reference dataset, the classifier minimizes the negative log-likelihood over the reference classes as follows:

$$-\frac{1}{\mathrm{Q}}\sum_{q=1}^{\mathrm{Q}} c_q * \log(f(B * A * X_q))$$

where X_q corresponds to the k-mer vectors for the q-th sentence, $c_q$is the class label, A and B are the matrices and f is the softmax function (see **Figure 6.1C**).

MetaMLP enables the bag of n-grams from FastText to capture partial information from the k-mers. These n-grams are subsequences from the k-mers passed along with the full size k-mer allowing to identify k-mers with partial matching.

6.2.1.3 Prediction of Short Sequencing Read

MetaMLP is designed to efficiently profile metagenomic samples with millions of reads from short sequencing libraries against a target reference database. As reads are sequences of nucleotides, MetaMLP first translates each sequence into six reading frames. Then, for each reading frame a random k-mer is selected from its sequence and checked against the hash table that was built during the indexing stage. If a k-mer is found in the hash table, all k-mers are subtracted from the read and classified using the trained CBOW model. If not, a new random k-mer is selected from the read at a different position. This process is repeated to a maximum number of tries defined by the user. If more than one reading frame gets classified, MetaMLP picks up the reading frame with the highest classification probability (see **Figure 6.1B**).

Once a full metagenomic data set is processed, MetaMLP counts the number of reads per class using a minimum probability cutoff defined by the user and reports the absolute abundance table. Additionally, MetaMLP also reports a fasta file containing the read name along with its classifications, probabilities and sequence. This file is useful for cases where MetaMLP is used as a filter to target a particular functional classes.

6.2.2 Databases

Pathway	Training Proteins	Validation Genes	Reads
Amino-acid_biosynthesis	501	126	7148
Amino-acid_degradation	99	25	1422
Antibiotic_biosynthesis	82	20	1121
Aromatic_compound_metabolism	68	17	890
Bacterial_outer_membrane_biogenesis	62	16	901
Carbohydrate_biosynthesis	79	20	1071
Carbohydrate_degradation	167	42	2199
Carbohydrate_metabolism	138	35	1903
Cell_wall_biogenesis	190	48	2642
Cofactor_biosynthesis	344	86	5082
Isoprenoid_biosynthesis	42	10	651
Lipid_metabolism	177	45	2569
Metabolic_intermediate_biosynthesis	95	24	1379
Nitrogen_metabolism	50	12	709
Nucleotide-sugar_biosynthesis	42	10	505
One-carbon_metabolism	45	11	588
Porphyrin-containing_compound_metabolism	66	16	713
Protein_modification	46	12	663
Purine_metabolism	133	33	1720
Pyrimidine_metabolism	105	26	1397
Xenobiotic_degradation	41	10	478

Table 6.1: UniProt pathway database with number of proteins used for training, number of genes used for validation and the simulated number of reads for each pathway category.

6.2.2.1 Pathway Reference Database

Bacterial protein sequences from the Universal Protein Resource (UniProt) were downloaded. Then, only proteins that have been manually curated, and contained evidence at the protein level were used for downstream analysis. In total 20,161 proteins were obtained and 4,105 of those were annotated to at least one pathway. Lastly, pathways with less than 50 proteins were discarded to get a total of 3,216 proteins and 21 different pathways (see **Table 6.1**).

6.2.2.2 Antibiotic Resistance Database

MetaMLP was trained to identify short reads associated to Antibiotic Resistance Genes (ARGs) from metagenomic short sequencing data. Thus, the DeepARG-DB-v2 database [25] containing a total of 12,260 sequences distributed through 30 antibiotic categories was downloaded. However, only antibiotic resistance categories with at least 50 protein sequences were considered for downstream analysis. Thus, a total of 12,147 proteins and 14 categories were used to train the MetaMLP model (see **Table 6.2**).

Antibiotic Class	Proteins
multidrug	4456
beta-lactam	2885
MLS	1710
tetracycline	557
fosfomycin	434
aminoglycoside	403
glycopeptide	346
unclassified	311
bacitracin	280
polymyxin	245
fluoroquinolone	158
phenicol	157
sulfonamide	125
diaminopyrimidine	80

Table 6.2: Antibiotic resistance categories from ARGminer

6.2.2.3 Gene Ontology Reference Database

Protein sequences associated to the biological process response to stress (GO:0006950) were downloaded from the UniProt website. However, only bacterial curated sequences and biological processes with at least 100 sequences were considered for downstream analysis (see **Table 6.3**).

GO Term	Biological Process	Proteins
GO:0006935	chemotaxis	312
GO:0006515	protein_quality_control_for_misfolded_or_incompletely_synthesized_proteins	123
GO:0006298	mismatch_repair	1306
GO:0042742	defense_response_to_bacterium	136
GO:0046677	response_to_antibiotic	975
GO:0009432	SOS_response	1030
GO:0006814	sodium_ion_transport	125
GO:0051607	defense_response_to_virus	149
GO:0051775	response_to_redox_state	164
GO:0045454	cell_redox_homeostasis	164
GO:0009236	cobalamin_biosynthetic_process	505
GO:0045910	negative_regulation_of_DNA_recombination	150
GO:0006281	DNA_repair	3450
GO:0006261	DNA-dependent_DNA_replication	228
GO:0006289	nucleotide-excision_repair	1485
GO:0010038	response_to_metal_ion	123
GO:0009163	nucleoside_biosynthetic_process	204
GO:0006541	glutamine_metabolic_process	258
GO:0030091	protein_repair	186
GO:0034605	cellular_response_to_heat	128
GO:0019835	cytolysis	140
GO:0006355	regulation_of_transcription,_DNA-templated	360
GO:0019380	3-phenylpropionate_catabolic_process	224
GO:0045892	negative_regulation_of_transcription,_DNA-templated	203
GO:0000724	double-strand_break_repair_via_homologous_recombination	296
GO:0000160	phosphorelay_signal_transduction_system	449
GO:0006979	response_to_oxidative_stress	677
GO:0006310	DNA_recombination	1766
GO:0043571	maintenance_of_CRISPR_repeat_elements	101
GO:0009307	DNA_restriction-modification_system	283
GO:0006284	base-excision_repair	1097
GO:0006974	cellular_response_to_DNA_damage_stimulus	118
GO:0042744	hydrogen_peroxide_catabolic_process	371
GO:0005975	carbohydrate_metabolic_process	131
GO:0006260	DNA_replication	958
GO:0009636	response_to_toxic_substance	149
GO:0009405	pathogenesis	116
GO:0006811	ion_transport	102
GO:0006109	regulation_of_carbohydrate_metabolic_process	214

Table 6.3: Database of response to stress associated categories using Gene Ontology terms.

In addition, the GO database comprises proteins with multiple associated labels. For instance, the protein sequence Q55002 is associated to response to antibiotic (GO:0046677) and translation (GO:0006412). Therefore, reads from this protein would be classified to both categories. However, as MetaMLP uses a softmax layer for prediction, it will distribute the probability between the two

categories. In an ideal scenario, both classes would have a probability of 0.5. This database was used to test the ability of MetaMLP to represent sequences associated to multiple labels.

6.2.2.4 True Positive Data set

The pathway database was used to build a true positive database. Because MetaMLP uses amino acid sequences for training and nucleotide sequences for querying, it was necessary to identify the corresponding nucleotide sequences for each one of the proteins in the pathways database. Therefore, UniProt identifiers were cross referenced against the RefSeq database and a list of gene candidates were found. Then, those candidates were aligned to the protein sequences using DIAMOND with a 90% identity and a 90% overlap. If multiple alignments were obtained using this criterion, the best hit was selected as the representative gene sequence for the target protein sequence. Thus, each entry in the database contained a respective gene sequence. Finally, the pathway database was randomly splitted into training (80%) and validation (20%). The training set was used to train the model whereas the validation set was exclusively used to test the method after the training was done. Therefore, the validation set was never used during the training process. Note that the training set corresponds to amino acid sequences whereas the validation set consists of nucleotide sequences. To simulate a library of short sequence reads, sequences that are 100bp long were randomly selected from each nucleotide sequence from the validation data set. Thus, a total of 35,751 short reads were generated.

DIAMOND is currently one of the widely used tools for metagenomic analysis. Therefore, to test the performance of MetaMLP, the best hit approach was used. DIAMOND was run by using a sequence alignment identity of 80%, whereas MetaMLP was set with a minimum probability of 0.8. Precision, recall and, F1 score were computed to measure the performance of both approaches.

6.2.2.5 False Positive Dataset

To test the ability of MetaMLP to filter out sequences that are not associated to any of the selected pathways (false positives), a synthetic dataset was constructed by using the same number of reads from the true positive dataset. However, each nucleotide position on this dataset was randomly selected. This negative dataset was then ran against MetaMLP and the best hit approach using DIAMOND with default parameters. Precision, recall and F1 score were computed to measure the performance of both methods.

6.2.2.6 Time and Memory Profiling

To evaluate the time performance and memory footprint of MetaMLP, a data set of 100k, 1M, 10M and 100M reads were built by randomly selecting reads from a real metagenomic soil sample of 407,645,066 reads. This sample is under the SRA accession number SRR2901746 and corresponds to a 250bp long read sample from the Illumina HiSeq 2000 sequencer. Along with

MetaMLP, DIAMOND was also run with the same data sets. Both methods ran with only one enabled CPU in the Ubuntu 16.4 Linux distribution.

6.2.2.7 Functional Annotation of Metagenomic Data sets

MetaMLP was used to profile 68 samples from four different environments by using the pathways, response to stress, and antibiotic resistance databases. The 68 public available metagenomes were downloaded from the Sequence Read Archive (SRA) from the National Center for Biotechnology Information (NCBI) spanning four different environments as follows: 10 soil (S), 15 human gut (HG), 15 freshwater (FW) and 28 wastewater (WW) samples. Results from MetaMLP were compared against the best hit approach using DIAMOND with an identity of 80%.
For the GO reference database, MetaMLP was run with a permissive 0.5 minimum probability to retrieve multiple classifications. Relative abundance results were compared against those obtained using sequence alignment with DIAMOND at an 80% identity cutoff.

6.3 Results and Discussion

6.3.1 Effect of k-mer size

The k-mer size is one of the key parameters for MetaMLP to perform accurate predictions. Therefore, MetaMLP was evaluated using the true positive dataset with a k-mer size ranging from 3 to 20 amino acids (see **Figure 6.2**). It was observed that for a large k-mer size, MetaMLP generates accurate predictions but penalizes the number of predicted reads. For instance, **Figure 6.2** shows that for a k-mer size of $k = 3$ MetaMLP is able to predict 99% of the reads, but, with a very low performance with a 0.7 F1 score. On the other hand, when using a k-mer size of $k = 20$ MetaMLP achieves a 0.99 F1 score. However, it was only able to detect 17% of the total number of reads. Interestingly, a k-mer size of $k = 11$ it is enough to achieve a 0.99 F1 score with a 29% of predicted reads. As shown in **Figure 6.2**, performance of MetaMLP does not improve when using a k-mer sizes larger than 11. Therefore, a k-mer size of $k = 11$ was set as default for training the MetaMLP models.

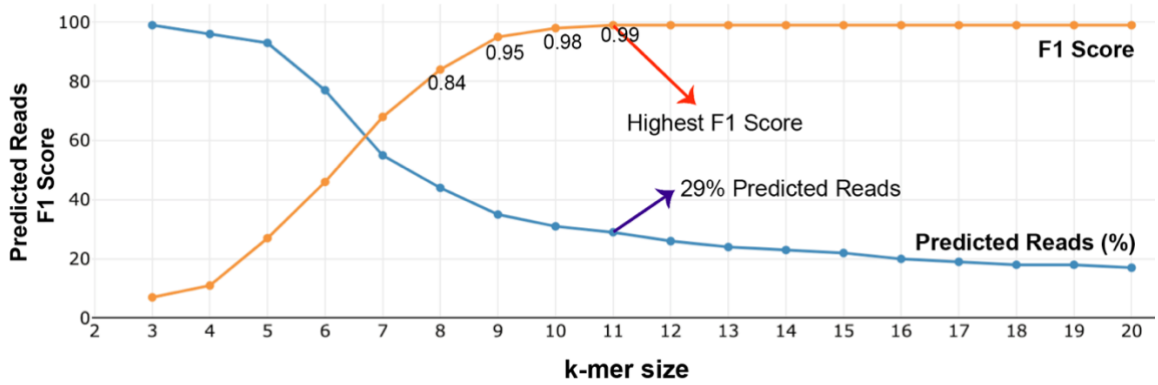

Figure 6.2: Performance of MetaMLP with different k-mer sizes.

The sequence embedding strategy allows MetaMLP to represent amino acid sequences into numerical vectors (embedding dimension) by taking into account the distribution of the k-mers in the protein sequence as well as their labels. Thus, MetaMLP uses the supervised embedding implementation from FastText to learn these numerical to group proteins based on their labels and k-mer context. For instance, proteins that belong to Beta-lactamase class are expected to cluster together and remain distant from members of other classes. **Figure 6.3** shows the distribution of the MetaMLP embeddings in a two-dimensional space generated by using the t-SNE algorithm [247].

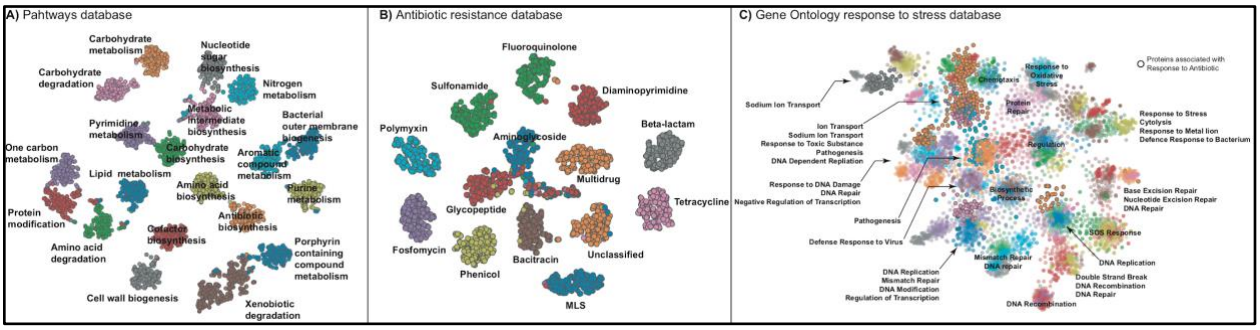

Figure 6.3: MetaMLP embeddings representation in two dimensional space for the pathways, ARGs and GO response to stress databases.

For targeted databases such as the ARG categories or pathways database, MetaMLP clustered categories according to their labels with a representative cohesion and separation (silhouette score: 0.56 and 0.62 for pathways and ARGs, respectively) (See **Figure 6.3A-B**). Interestingly, in a complex classification problem represented by the GO database where proteins contain multiple labels, MetaMLP show a consistent distribution over the clusters and its corresponding categories. Clusters shown in **Figure 6.3C** describes the relationship among different biological processes involved in response to stress. For example, proteins responding to antibiotics are also associated to other biological process such as response to toxic substances, pathogenesis, defense to virus, chemotaxis, response to DNA damage, among others. Such associations can be clearly seen from the embeddings visualization. Therefore, the embedding strategy adopted in MetaMLP is also suitable for representing reference databases where proteins contains multiple labels.

6.3.2 Detection of True Positive Hits

The pathways database was used to assess the ability of MetaMLP to 1) discriminate between pathway-like reads and 2) to evaluate the performance of MetaMLP on classifying short sequences from a particular pathway. To compare the performance of MetaMLP, the best hit approach using DIAMOND was used. In total, MetaMLP was able to identify 10,433 (29%) pathway-like reads out of the 35,751 reads with a probability greater than 0.8, whereas, the baseline approach was

able to identify 8,695 (24%) reads out of the 35,751 reads. This means MetaMLP was able to identify 5% more reads than the best hit approach at 80% identity.

Pathway	Precision	Recall	F1 Score
Amino-acid_biosynthesis	1	1	1
Amino-acid_degradation	0.94	1	0.97
Antibiotic_biosynthesis	1	0.97	0.98
Aromatic_compound_metabolism	0.78	1	0.87
Bacterial_outer_membrane_biogenesis	0	0	0
Carbohydrate_biosynthesis	1	0.98	0.99
Carbohydrate_degradation	1	1	1
Carbohydrate_metabolism	0.97	1	0.99
Cell_wall_biogenesis	1	1	1
Cofactor_biosynthesis	1	1	1
Isoprenoid_biosynthesis	1	1	1
Lipid_metabolism	1	1	1
Metabolic_intermediate_biosynthesis	1	1	1
Nitrogen_metabolism	1	1	1
Nucleotide-sugar_biosynthesis	1	1	1
One-carbon_metabolism	1	1	1
Porphyrin-containing_compound_metabolism	1	1	1
Protein_modification	1	1	1
Purine_metabolism	1	1	1
Pyrimidine_metabolism	1	1	1
Xenobiotic_degradation	1	0.47	0.64
Average	0.99	1	1

Table 6.4: Prediction performance of the best hit approach using DIAMOND

Further, both methods were compared on their positive predictions to evaluate their performance for discriminating reads from a particular pathway. As expected, the sequence alignment approach at 80% identity performed with a high average precision (0.99) and recall (1.00) (see **Table 6.4**), whereas MetaMLP was also close to perfect prediction with 0.99 average precision and 0.99 average recall (see **Table 6.5**), indicating the potential of k-mer embeddings to represent protein sequences to profile metagenomes.

Pathway	Precision	Recall	F1 Score
Amino-acid_biosynthesis	0.99	0.99	0.99
Amino-acid_degradation	0.97	0.97	0.97
Antibiotic_biosynthesis	0.93	0.82	0.87
Aromatic_compound_metabolism	0.42	0.44	0.43
Bacterial_outer_membrane_biogenesis	1	0.36	0.53
Carbohydrate_biosynthesis	0.97	1	0.98
Carbohydrate_degradation	0.99	0.99	0.99
Carbohydrate_metabolism	0.97	0.97	0.97
Cell_wall_biogenesis	0.99	1	0.99
Cofactor_biosynthesis	0.99	0.99	0.99
Isoprenoid_biosynthesis	1	0.99	0.99
Lipid_metabolism	0.98	1	0.99
Metabolic_intermediate_biosynthesis	0.97	0.97	0.97
Nitrogen_metabolism	1	1	1
Nucleotide-sugar_biosynthesis	1	0.99	1
One-carbon_metabolism	0.99	0.95	0.97
Porphyrin-containing_compound_metabolism	0.99	0.97	0.98
Protein_modification	1	1	1
Purine_metabolism	1	0.99	0.99
Pyrimidine_metabolism	0.99	1	1
Xenobiotic_degradation	0.8	0.46	0.59
Average	**0.99**	**0.99**	**0.99**

Table 6.5: Prediction performance of MetaMLP for the pathway database.

It is also worth mentioning that MetaMLP and the best hit approach did not perform well for three categories (aromatic compound metabolism, bacterial outer membrane biogenesis, and xenobiotic degradation). Interestingly, the best hit approach was not able to identify any read from bacterial outer membrane biogenesis when MetaMLP obtained a 1.00 precision but a low 0.13 recall, indicating a high sensitivity of MetaMLP in discriminating true positives from this category but failing for false negatives. In terms of relative abundance, the comparison of the read counts between the best hit approach and MetaMLP was very close with a correlation of 0.988, indicating that MetaMLP can correctly characterize the composition of the pathways in the simulated data set (see **Figure 6.4**).

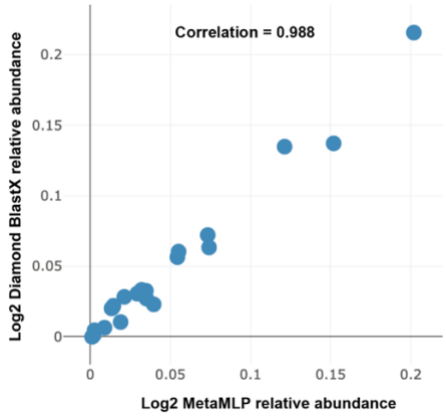

Figure 6.4: Correlation between pathway relative abundances computed from results from MetaMLP (x axis) and DIAMOND (y axis) using the true positive dataset.

6.3.3 Detection of False Positives Hits

The false positive rate is defined as the counts of negative samples predicted as positives. Thus, MetaMLP was ran against the true negative reads. Then, the false positive rate was computed as the number of reads predicted as any pathways from the total 35,751 negative reads. As result, MetaMLP classified only 2 reads out of the 35,751 negative reads indicating a very low false positive rate (0.005%). As expected, the best hit approach did not produce any relevant alignment, hence, having a null false positive rate (0%).

6.3.4 Time and Memory Usage of MetaMLP

The main advantage for building up a classifier instead of performing a sequence alignment is the improvement over the speed for making the annotations. Results have shown that MetaMLP keeps an almost identical level of sensitivity compared to DIAMOND. However, the strength of MetaMLP relies on its speed. **Table 6.6** shows the speed benchmarking over data sets with different numbers of reads. Note that MetaMLP is >50 times faster than DIAMOND for all the sample sizes. MetaMLP produces very similar results in terms of relative abundance using the ARGs database and pathway database with a correlation of 0.951 and 0.953, respectively (See **Figure 6.5**). Note that, in this test, MetaMLP identified 35% more ARG-like reads (253,370) compared to the number of reads (186,736) detected from DIAMOND. In addition, MetaMLP is also memory efficient, depending mostly on the size of the reference database. For instance, it requires a minimum ram memory of 1.0Gb to run the pathways database, 1.2Gb when using the ARGs database and 2.8Gb when using the GO database. When processing 100M reads, it required 1.7Gb in total with the pathways database whereas DIAMOND required 6.68Gb. The low memory usage in MetaMLP is a consequence of its classification strategy, where reads are loaded in chunks

of 10,000 reads for efficient I/O. Therefore, MetaMLP can be run in any personal computer without the need of using a large cluster with high amount of RAM memory.

Number of Reads	MetaMLP	Diamond
100,000	9 s	38 s
1,000,000	27 s	6 m
10,000,000	1 m	67 m
100,000,000	14 m	714 m

Table 6.6: Time profiling of MetaMLP compared to Diamond over different sample sizes.

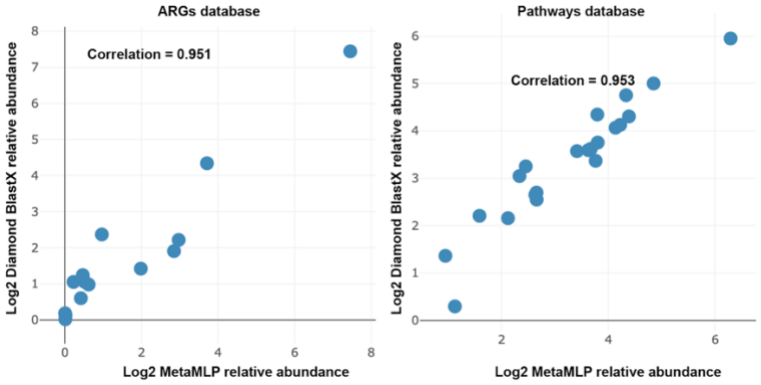

Figure 6.5: Correlation between MetaMLP and DIAMOND relative abundance results from the 100 Million dataset obtained from a real soil sample.

6.3.5 Functional Annotation of Different Environments

MetaMLP was run over the 67 real metagenomic samples processing a total of 2,186,933,071 reads. Of those reads, MetaMLP was able to predict 2,343,026 as ARG-like reads in 710 minutes using only one CPU, whereas DIAMOND identified 782,639 reads taking a total of 5,256 minutes using 20 CPUs. Thereafter, the average correlation of the abundances between Diamond and MetaMLP was of 0.84 (0.83 log transformed relative abundance). When tested against the pathways database, MetaMLP classified 6,920,009 reads whereas DIAMOND identified 6,179,910 reads. In terms of relative abundances, the average correlation of was of 0.94 (0.83 log transformed relative abundance).

6.3.5.1 Observation of MetaMLP Annotations against an extensive Metagenomics Study

An extensive study carried out by Pal et. al. [91] uses over 800 metagenomic samples spanning several environments with a sequence alignment strategy at a 90% identity cutoff for annotation. This study (named Pal800 for simplicity) shows that the human gut microbiota is one of the environments with the highest relative abundance compared to other microbiomes (soil, wastewater, and freshwater). Concordantly, when MetaMLP was run over the 68 real metagenomic samples using the GO database, it also profiled the human gut microbiome as the highest relative abundance for the response to antibiotic process (see **Figure 6.6**). Note that Pal800 used a curated ARG database, and, therefore, it did not consider the induction of false positives. However, the GO database only provides a general overview of the functional composition of those environments. Therefore, a more detailed analysis was obtained by looking at the results from MetaMLP using the specialized ARG database. As result, the same trend was observed when comparing both analyses (MetaMLP, Pal800). For example, the tetracycline category had the highest relative abundance in the human microbiome, sulfonamide shows the highest relative abundance in the wastewater environment, the relative abundance of the beta-lactamase class was higher in the freshwater compared to the wastewater, and both are higher than human gut and soil environments. Pal800 also performed a composition profile of the mobile genetic elements present in the microbiomes. It shows that wastewater, freshwater, and soil environments had a higher relative abundance compared to the human gut. Interestingly, for MetaMLP, the GO response to stress database conveyed a similar trend in relative abundance for the biological process "establishment of competence for transformation" (see Transformation **Figure 6.6**). This term is associated to genetic transfer between organisms and is described by the GO consortium as the process where exogenous DNA is acquired by a bacterium. Overall, despite only using 67 real metagenomes, the functional annotation carried out by MetaMLP described a very similar trending for relative abundances when compared to the Pal800 study, indicating a real scenario usage of MetaMLP.

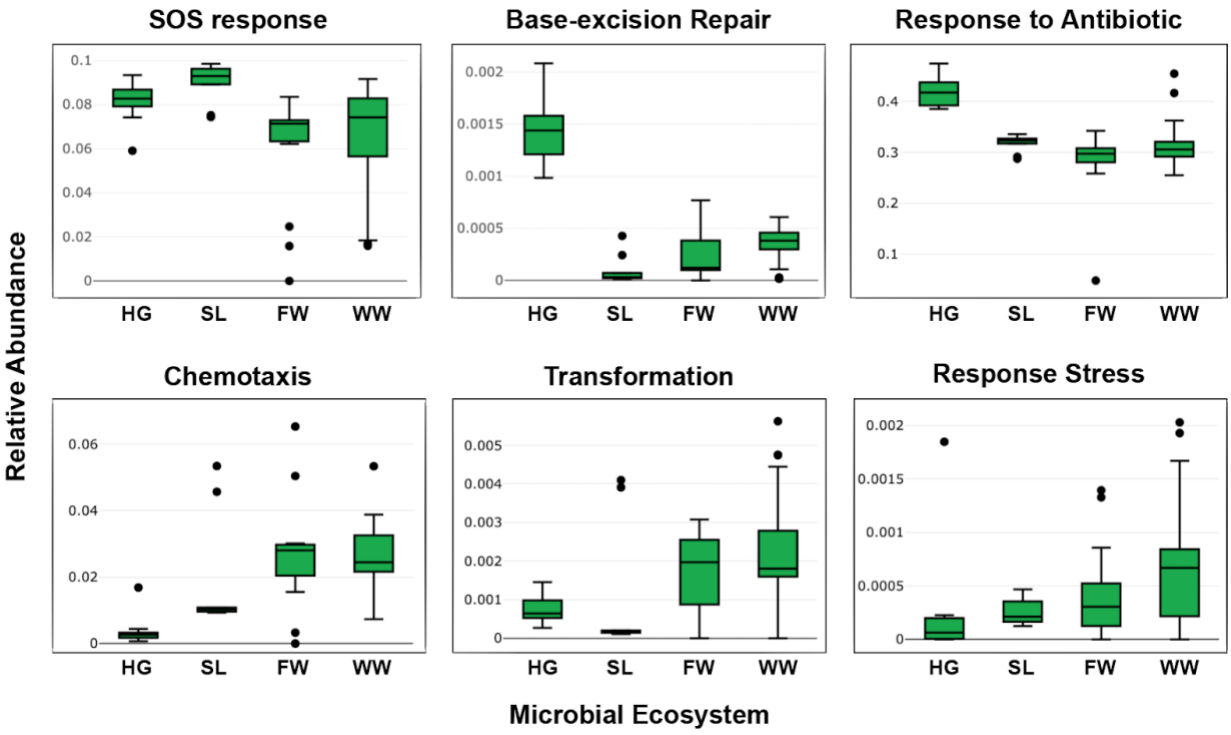

Figure 6.6: Relative abundance of biological process from the GO response to stress database.

6.4 Conclusions

MetaMLP is an alignment-free method for profiling metagenomic samples to specific target group of proteins (e.g., ARGs, pathways, GO terms) using a machine learning classifier. It uses sequence embeddings to represent protein and DNA sequences as numerical vectors and a machine learning classifier to discriminate between protein functions. Results show that MetaMLP identifies more reads than the widely used best hit approach (sequence alignment with identity >80%) and has as good performance as a sequence alignment method. Remarkably, MetaMLP is around 50 times faster than the DIAMOND aligner, the most widely used sequence alignment tool for metagenomic data sets. MetaMLP can be trained using any collection of protein sequences (reference database) and keeps a very low memory footprint for the specialized databases used. Finally, MetaMLP is open sourced and freely available at https://bitbucket.org/gaarangoa/metamlp/src/master/.

CHAPTER 7 CONCLUSIONS

In this thesis, several computing tools were developed to address different challenges in the processing and analysis of metagenomic data. These tools were aimed to help with the understanding of the role of microbes in their environment, particularly to profile antibiotic resistance by using a combination of web platforms and standalone tools. The main conclusions of this thesis are listed below:

1. MetaStorm, a web platform dedicated to the analysis of metagenomic samples has been proposed and developed. In its current state, MetaStorm has been widely used by the scientific community around the world. It has processed more than 10TB of raw sequencing data and has 247 users who have submitted 3,116 samples and customized reference databases. MetaStorm has been used primary for the analysis of antibiotic resistance genes as well as for mobile genetic elements, metal resistance genes, cluster of orthologous proteins, and 16S rRNA databases, making it a valuable resource for the antibiotic resistance and environmental engineering community.
2. DeepARG, a machine learning approach for profiling ARGs from metagenomic samples, was proposed as an alternative to the current practices. It was shown that DeepARG improved significantly the performance of ARG annotation compared to the best hit approach. By using a machine learning strategy along with sequence alignment scores, DeepARG is not sensitive to strict identity cutoffs. In addition, an ARG database (DeepARG-DB) was also released along with the deep learning models. However, as this database was developed from other ARG resources, it is sensitive to propagation errors, highlighting the need for developing an strategy for the curation and validation of ARGs.
3. The annotation of ARGs is a time consuming task, which is normally performed by expert curators. Thus, ARGminer, a web platform, was developed with the aim to incorporate information from different ARG resources (8 ARGs databases, MGEs, pathogen databases as well as general databases such as UniProt and PubMed) to decrease the complexity of manual curation. Additionally, ARGminer was developed as an open source resource with collaboration as its major strength. Therefore, the annotation of ARGs is open to expert curators as well as the general community who wants to work on this task. To validate ARGminer, crowdsourcing was used to recruit curators without specific expertise in ARGs. With results comparable to that obtained by expert curators, crowdsourcing has shown to be a powerful alternative for finding the consensus of antibiotic classes as well for validating ARG names that may vary from databases.
4. NanoARG, a web platform for the annotation of nanopore sequences was designed to profile long sequences against several databases including ARGs, MGEs, MRGs, and taxonomy. It has the potential to also process large contigs and full genomes. It integrates our previously developed deep learning models from DeepARG, which facilitates the detection of ARGs with low sequence similarity to known ARGs.

5. Word vectors have proven to be a powerful representation of protein and DNA sequences. We developed MetaMLP, a machine learning based approach for profiling metagenomic data sets to a variety of gene functions. Unlike DIAMOND, MetaMLP is an alignment free technique that represents protein sequences with numerical vectors by using word embeddings, a widely used approach in natural language processing. MetaMLP proved to be as sensitive as a sequence alignment approach but much faster (>50 times over 100 million reads), also keeping a high similarity in terms of relative abundance (correlation > 0.9). MetaMLP is the first effort of using the supervised CBOW models for representing protein sequences. This opens the opportunity for future exploration in this area, particularly, in the detection of novel genes from metagenomic samples and improvements on the ways to represent protein sequences.

REFERENCES

1. Riesenfeld CS, Schloss PD, Handelsman J: **Metagenomics: genomic analysis of microbial communities**. *Annu Rev Genet* 2004, **38**:525-552.
2. Streit WR, Schmitz RA: **Metagenomics–the key to the uncultured microbes**. *Current opinion in microbiology* 2004, **7**(5):492-498.
3. Daniel R: **The metagenomics of soil**. *Nature Reviews Microbiology* 2005, **3**(6):470.
4. Wilke A, Bischof J, Gerlach W, Glass E, Harrison T, Keegan KP, Paczian T, Trimble WL, Bagchi S, Grama A: **The MG-RAST metagenomics database and portal in 2015**. *Nucleic acids research* 2015, **44**(D1):D590-D594.
5. Hunter S, Corbett M, Denise H, Fraser M, Gonzalez-Beltran A, Hunter C, Jones P, Leinonen R, McAnulla C, Maguire E: **EBI metagenomics—a new resource for the analysis and archiving of metagenomic data**. *Nucleic acids research* 2013, **42**(D1):D600-D606.
6. Oulas A, Pavloudi C, Polymenakou P, Pavlopoulos GA, Papanikolaou N, Kotoulas G, Arvanitidis C, Iliopoulos l: **Metagenomics: tools and insights for analyzing next-generation sequencing data derived from biodiversity studies**. *Bioinformatics and biology insights* 2015, **9**:BBI. S12462.
7. Luo C, Tsementzi D, Kyrpides N, Read T, Konstantinidis KT: **Direct comparisons of Illumina vs. Roche 454 sequencing technologies on the same microbial community DNA sample**. *PloS one* 2012, **7**(2):e30087.
8. Ip CL, Loose M, Tyson JR, de Cesare M, Brown BL, Jain M, Leggett RM, Eccles DA, Zalunin V, Urban JM: **MinION Analysis and Reference Consortium: Phase 1 data release and analysis**. *F1000Research* 2015, **4**.
9. Souza RC, Cantão ME, Vasconcelos ATR, Nogueira MA, Hungria M: **Soil metagenomics reveals differences under conventional and no-tillage with crop rotation or succession**. *Applied Soil Ecology* 2013, **72**:49-61.
10. Amos G, Zhang L, Hawkey P, Gaze W, Wellington E: **Functional metagenomic analysis reveals rivers are a reservoir for diverse antibiotic resistance genes**. *Veterinary microbiology* 2014, **171**(3-4):441-447.
11. Szczepanowski R, Linke B, Krahn I, Gartemann K-H, Guetzkow T, Eichler W, Pühler A, Schlueter A: **Detection of 140 clinically relevant antibiotic-resistance genes in the plasmid metagenome of wastewater treatment plant bacteria showing reduced susceptibility to selected antibiotics**. *Microbiology* 2009, **155**(7):2306-2319.
12. Yang Y, Yu K, Xia Y, Lau FT, Tang DT, Fung WC, Fang HH, Zhang T: **Metagenomic analysis of sludge from full-scale anaerobic digesters operated in municipal wastewater treatment plants**. *Applied microbiology and biotechnology* 2014, **98**(12):5709-5718.

13. Wang Z, Zhang X-X, Huang K, Miao Y, Shi P, Liu B, Long C, Li A: **Metagenomic profiling of antibiotic resistance genes and mobile genetic elements in a tannery wastewater treatment plant**. *PloS one* 2013, **8**(10):e76079.
14. Preidis GA, Versalovic J: **Targeting the human microbiome with antibiotics, probiotics, and prebiotics: gastroenterology enters the metagenomics era**. *Gastroenterology* 2009, **136**(6):2015-2031.
15. Tang P, Chiu C: **Metagenomics for the discovery of novel human viruses**. *Future microbiology* 2010, **5**(2):177-189.
16. Walker AW, Duncan SH, Louis P, Flint HJ: **Phylogeny, culturing, and metagenomics of the human gut microbiota**. *Trends in microbiology* 2014, **22**(5):267-274.
17. Lewin A, Wentzel A, Valla S: **Metagenomics of microbial life in extreme temperature environments**. *Current opinion in biotechnology* 2013, **24**(3):516-525.
18. Cowan DA, Ramond J-B, Makhalanyane TP, De Maayer P: **Metagenomics of extreme environments**. *Current opinion in microbiology* 2015, **25**:97-102.
19. Arango-Argoty G, Singh G, Heath LS, Pruden A, Xiao W, Zhang L: **MetaStorm: A Public Resource for Customizable Metagenomics Annotation**. *PloS one* 2016, **11**(9):e0162442.
20. Yang Y, Jiang X, Chai B, Ma L, Li B, Zhang A, Cole JR, Tiedje JM, Zhang T: **ARGs-OAP: online analysis pipeline for antibiotic resistance genes detection from metagenomic data using an integrated structured ARG-database**. *Bioinformatics* 2016, **32**(15):2346-2351.
21. Bengtsson-Palme J, Larsson DJ, Kristiansson E: **Using metagenomics to investigate human and environmental resistomes**. *Journal of Antimicrobial Chemotherapy* 2017, **72**(10):2690-2703.
22. McArthur AG, Waglechner N, Nizam F, Yan A, Azad MA, Baylay AJ, Bhullar K, Canova MJ, De Pascale G, Ejim L: **The comprehensive antibiotic resistance database**. *Antimicrobial agents and chemotherapy* 2013, **57**(7):3348-3357.
23. Lakin SM, Dean C, Noyes NR, Dettenwanger A, Ross AS, Doster E, Rovira P, Abdo Z, Jones KL, Ruiz J: **MEGARes: an antimicrobial resistance database for high throughput sequencing**. *Nucleic acids research* 2017, **45**(D1):D574-D580.
24. Liu B, Pop M: **ARDB—antibiotic resistance genes database**. *Nucleic acids research* 2008, **37**(suppl_1):D443-D447.
25. Arango-Argoty G, Garner E, Pruden A, Heath LS, Vikesland P, Zhang L: **DeepARG: A deep learning approach for predicting antibiotic resistance genes from metagenomic data**. *Microbiome* 2018, **6**(1):23.
26. Gupta SK, Padmanabhan BR, Diene SM, Lopez-Rojas R, Kempf M, Landraud L, Rolain JM: **ARG-ANNOT, a new bioinformatic tool to discover antibiotic resistance genes in bacterial genomes**. *Antimicrob Agents Chemother* 2014, **58**(1):212-220.
27. Organization WH: **Antimicrobial resistance: global report on surveillance**: World Health Organization; 2014.

28. O'Neill J: **Antimicrobial resistance: tackling a crisis for the health and wealth of nations**. *The Review on Antimicrobial Resistance* 2014, **20**.
29. Pehrsson EC, Tsukayama P, Patel S, Mejía-Bautista M, Sosa-Soto G, Navarrete KM, Calderon M, Cabrera L, Hoyos-Arango W, Bertoli MT: **Interconnected microbiomes and resistomes in low-income human habitats**. *Nature* 2016, **533**(7602):212-216.
30. Wright GD: **Antibiotic resistance in the environment: a link to the clinic?** *Current opinion in microbiology* 2010, **13**(5):589-594.
31. Li A-D, Li L-G, Zhang T: **Exploring antibiotic resistance genes and metal resistance genes in plasmid metagenomes from wastewater treatment plants**. *Frontiers in microbiology* 2015, **6**:1025.
32. Kleinheinz KA, Joensen KG, Larsen MV: **Applying the ResFinder and VirulenceFinder web-services for easy identification of acquired antibiotic resistance and E. coli virulence genes in bacteriophage and prophage nucleotide sequences**. *Bacteriophage* 2014, **4**(1):e27943.
33. Baker-Austin C, Wright MS, Stepanauskas R, McArthur J: **Co-selection of antibiotic and metal resistance**. *Trends in microbiology* 2006, **14**(4):176-182.
34. Chapman JS: **Disinfectant resistance mechanisms, cross-resistance, and co-resistance**. *International Biodeterioration & Biodegradation* 2003, **51**(4):271-276.
35. Haenni M, Poirel L, Kieffer N, Châtre P, Saras E, Métayer V, Dumoulin R, Nordmann P, Madec J-Y: **Co-occurrence of extended spectrum β lactamase and MCR-1 encoding genes on plasmids**. *The Lancet infectious diseases* 2016, **16**(3):281-282.
36. Gillings MR, Gaze WH, Pruden A, Smalla K, Tiedje JM, Zhu Y-G: **Using the class 1 integron-integrase gene as a proxy for anthropogenic pollution**. *The ISME journal* 2015, **9**(6):1269.
37. Branton D, Deamer DW, Marziali A, Bayley H, Benner SA, Butler T, Di Ventra M, Garaj S, Hibbs A, Huang X: **The potential and challenges of nanopore sequencing**. *Nature biotechnology* 2008, **26**(10):1146.
38. Mikheyev AS, Tin MM: **A first look at the Oxford Nanopore MinION sequencer**. *Molecular ecology resources* 2014, **14**(6):1097-1102.
39. Vinga S, Almeida J: **Alignment-free sequence comparison—a review**. *Bioinformatics* 2003, **19**(4):513-523.
40. Buchfink B, Xie C, Huson DH: **Fast and sensitive protein alignment using DIAMOND**. *Nature methods* 2015, **12**(1):59.
41. Smith T, BEYER W: **M. S. WATERMAN**. 1976.
42. Bojanowski P, Grave E, Joulin A, Mikolov T: **Enriching word vectors with subword information**. *arXiv preprint arXiv:160704606* 2016.
43. Joulin A, Grave E, Bojanowski P, Mikolov T: **Bag of tricks for efficient text classification**. *arXiv preprint arXiv:160701759* 2016.
44. Walter J, Ley R: **The human gut microbiome: ecology and recent evolutionary changes**. *Annual review of microbiology* 2011, **65**:411-429.

45. Gill SR, Pop M, DeBoy RT, Eckburg PB, Turnbaugh PJ, Samuel BS, Gordon JI, Relman DA, Fraser-Liggett CM, Nelson KE: **Metagenomic analysis of the human distal gut microbiome**. *science* 2006, **312**(5778):1355-1359.
46. Qin J, Li Y, Cai Z, Li S, Zhu J, Zhang F, Liang S, Zhang W, Guan Y, Shen D: **A metagenome-wide association study of gut microbiota in type 2 diabetes**. *Nature* 2012, **490**(7418):55.
47. Quaiser A, Zivanovic Y, Moreira D, López-García P: **Comparative metagenomics of bathypelagic plankton and bottom sediment from the Sea of Marmara**. *The ISME journal* 2011, **5**(2):285.
48. Parthasarathy H, Hill E, MacCallum C: **Global ocean sampling collection**. In.: Public Library of Science; 2007.
49. Sogin ML, Morrison HG, Huber JA, Welch DM, Huse SM, Neal PR, Arrieta JM, Herndl GJ: **Microbial diversity in the deep sea and the underexplored "rare biosphere"**. *Proceedings of the National Academy of Sciences* 2006, **103**(32):12115-12120.
50. Ghai R, Rodŕíguez-Valera F, McMahon KD, Toyama D, Rinke R, de Oliveira TCS, Garcia JW, de Miranda FP, Henrique-Silva F: **Metagenomics of the water column in the pristine upper course of the Amazon river**. *PloS one* 2011, **6**(8):e23785.
51. Oh S, Caro-Quintero A, Tsementzi D, DeLeon-Rodriguez N, Luo C, Poretsky R, Konstantinidis KT: **Metagenomic insights into the evolution, function, and complexity of the planktonic microbial community of Lake Lanier, a temperate freshwater ecosystem**. *Appl Environ Microbiol* 2011, **77**(17):6000-6011.
52. Ghai R, Hernandez CM, Picazo A, Mizuno CM, Ininbergs K, Díez B, Valas R, DuPont CL, McMahon KD, Camacho A: **Metagenomes of Mediterranean coastal lagoons**. *Scientific reports* 2012, **2**:490.
53. Schlüter A, Krause L, Szczepanowski R, Goesmann A, Pühler A: **Genetic diversity and composition of a plasmid metagenome from a wastewater treatment plant**. *Journal of biotechnology* 2008, **136**(1-2):65-76.
54. Berry D, Xi C, Raskin L: **Microbial ecology of drinking water distribution systems**. *Current opinion in biotechnology* 2006, **17**(3):297-302.
55. Fierer N, Breitbart M, Nulton J, Salamon P, Lozupone C, Jones R, Robeson M, Edwards RA, Felts B, Rayhawk S: **Metagenomic and small-subunit rRNA analyses reveal the genetic diversity of bacteria, archaea, fungi, and viruses in soil**. *Appl Environ Microbiol* 2007, **73**(21):7059-7066.
56. Adey P: **Mobility**: Routledge; 2009.
57. Dupré J, O'Malley MA: **Metagenomics and biological ontology**. *Studies in History and Philosophy of Science Part C: Studies in History and Philosophy of Biological and Biomedical Sciences* 2007, **38**(4):834-846.
58. Sharpton TJ: **An introduction to the analysis of shotgun metagenomic data**. *Frontiers in plant science* 2014, **5**:209.

59. Wooley JC, Godzik A, Friedberg I: **A primer on metagenomics**. *PLoS computational biology* 2010, **6**(2):e1000667.
60. Meyer F, Paarmann D, D'Souza M, Olson R, Glass EM, Kubal M, Paczian T, Rodriguez A, Stevens R, Wilke A: **The metagenomics RAST server–a public resource for the automatic phylogenetic and functional analysis of metagenomes**. *BMC bioinformatics* 2008, **9**(1):386.
61. Huson DH, Auch AF, Qi J, Schuster SC: **MEGAN analysis of metagenomic data**. *Genome research* 2007, **17**(3):377-386.
62. Kultima JR, Sunagawa S, Li J, Chen W, Chen H, Mende DR, Arumugam M, Pan Q, Liu B, Qin J: **MOCAT: a metagenomics assembly and gene prediction toolkit**. *PloS one* 2012, **7**(10):e47656.
63. Caporaso JG, Kuczynski J, Stombaugh J, Bittinger K, Bushman FD, Costello EK, Fierer N, Pena AG, Goodrich JK, Gordon JI: **QIIME allows analysis of high-throughput community sequencing data**. *Nature methods* 2010, **7**(5):335.
64. Haft DH, Tovchigrechko A: **High-speed microbial community profiling**. *Nature methods* 2012, **9**(8):793.
65. Ehrlich SD, Consortium M: **MetaHIT: The European Union Project on metagenomics of the human intestinal tract**. In: *Metagenomics of the human body.* Springer; 2011: 307-316.
66. Luo C, Rodriguez-r LM, Konstantinidis KT: **MyTaxa: an advanced taxonomic classifier for genomic and metagenomic sequences**. *Nucleic acids research* 2014, **42**(8):e73-e73.
67. Wu CH, Apweiler R, Bairoch A, Natale DA, Barker WC, Boeckmann B, Ferro S, Gasteiger E, Huang H, Lopez R: **The Universal Protein Resource (UniProt): an expanding universe of protein information**. *Nucleic acids research* 2006, **34**(suppl_1):D187-D191.
68. Tatusov RL, Galperin MY, Natale DA, Koonin EV: **The COG database: a tool for genome-scale analysis of protein functions and evolution**. *Nucleic acids research* 2000, **28**(1):33-36.
69. Li B, Yang Y, Ma L, Ju F, Guo F, Tiedje JM, Zhang T: **Metagenomic and network analysis reveal wide distribution and co-occurrence of environmental antibiotic resistance genes**. *The ISME journal* 2015, **9**(11):2490.
70. Miller JR, Koren S, Sutton G: **Assembly algorithms for next-generation sequencing data**. *Genomics* 2010, **95**(6):315-327.
71. Cock PJ, Fields CJ, Goto N, Heuer ML, Rice PM: **The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants**. *Nucleic acids research* 2009, **38**(6):1767-1771.
72. Consortium U: **UniProt: a hub for protein information**. *Nucleic acids research* 2014:gku989.

73. DeSantis TZ, Hugenholtz P, Larsen N, Rojas M, Brodie EL, Keller K, Huber T, Dalevi D, Hu P, Andersen GL: **Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with ARB**. *Appl Environ Microbiol* 2006, **72**(7):5069-5072.
74. Bolger AM, Lohse M, Usadel B: **Trimmomatic: a flexible trimmer for Illumina sequence data**. *Bioinformatics* 2014, **30**(15):2114-2120.
75. Peng Y, Leung HC, Yiu S-M, Chin FY: **IDBA-UD: a de novo assembler for single-cell and metagenomic sequencing data with highly uneven depth**. *Bioinformatics* 2012, **28**(11):1420-1428.
76. Abbas MM, Malluhi QM, Balakrishnan P: **Assessment of de novo assemblers for draft genomes: a case study with fungal genomes**. *BMC genomics* 2014, **15**(9):S10.
77. Lax S, Smith DP, Hampton-Marcell J, Owens SM, Handley KM, Scott NM, Gibbons SM, Larsen P, Shogan BD, Weiss S: **Longitudinal analysis of microbial interaction between humans and the indoor environment**. *Science* 2014, **345**(6200):1048-1052.
78. Di Rienzi SC, Sharon I, Wrighton KC, Koren O, Hug LA, Thomas BC, Goodrich JK, Bell JT, Spector TD, Banfield JF: **The human gut and groundwater harbor non-photosynthetic bacteria belonging to a new candidate phylum sibling to Cyanobacteria**. *Elife* 2013, **2**:e01102.
79. Hyatt D, Chen G-L, LoCascio PF, Land ML, Larimer FW, Hauser LJ: **Prodigal: prokaryotic gene recognition and translation initiation site identification**. *BMC bioinformatics* 2010, **11**(1):119.
80. McGinnis S, Madden TL: **BLAST: at the core of a powerful and diverse set of sequence analysis tools**. *Nucleic acids research* 2004, **32**(suppl_2):W20-W25.
81. Quast C, Pruesse E, Yilmaz P, Gerken J, Schweer T, Yarza P, Peplies J, Glöckner FO: **The SILVA ribosomal RNA gene database project: improved data processing and web-based tools**. *Nucleic acids research* 2012, **41**(D1):D590-D596.
82. Wilke A, Glass E, Bischof J, Braithwaite D, Souza M, Gerlach W: **MG-RAST technical report and manual for version 3.3. 6–Rev 1**. *Lemont, IL: Argonne National Laboratory* 2013.
83. Truong DT, Franzosa EA, Tickle TL, Scholz M, Weingart G, Pasolli E, Tett A, Huttenhower C, Segata N: **MetaPhlAn2 for enhanced metagenomic taxonomic profiling**. *Nature methods* 2015, **12**(10):902.
84. Pearson WR: **An introduction to sequence similarity ("homology") searching**. *Curr Protoc Bioinformatics* 2013, **Chapter 3**:Unit3 1.
85. Cantarel BL, Coutinho PM, Rancurel C, Bernard T, Lombard V, Henrissat B: **The Carbohydrate-Active EnZymes database (CAZy): an expert resource for glycogenomics**. *Nucleic acids research* 2008, **37**(suppl_1):D233-D238.
86. Langmead B, Salzberg SL: **Fast gapped-read alignment with Bowtie 2**. *Nature methods* 2012, **9**(4):357.

87. Vuong C, Yeh AJ, Cheung GY, Otto M: **Investigational drugs to treat methicillin-resistant Staphylococcus aureus**. *Expert opinion on investigational drugs* 2016, **25**(1):73-93.
88. Gandhi NR, Nunn P, Dheda K, Schaaf HS, Zignol M, Van Soolingen D, Jensen P, Bayona J: **Multidrug-resistant and extensively drug-resistant tuberculosis: a threat to global control of tuberculosis**. *The Lancet* 2010, **375**(9728):1830-1843.
89. Mediavilla JR, Patrawalla A, Chen L, Chavda KD, Mathema B, Vinnard C, Dever LL, Kreiswirth BN: **Colistin-and carbapenem-resistant Escherichia coli harboring mcr-1 and blaNDM-5, causing a complicated urinary tract infection in a patient from the United States**. *MBio* 2016, **7**(4):e01191-01116.
90. Hu Y, Liu F, Lin IY, Gao GF, Zhu B: **Dissemination of the mcr-1 colistin resistance gene**. *The Lancet infectious diseases* 2016, **16**(2):146-147.
91. Pal C, Bengtsson-Palme J, Kristiansson E, Larsson DG: **The structure and diversity of human, animal and environmental resistomes**. *Microbiome* 2016, **4**(1):54.
92. Forsberg KJ, Patel S, Gibson MK, Lauber CL, Knight R, Fierer N, Dantas G: **Bacterial phylogeny structures soil resistomes across habitats**. *Nature* 2014, **509**(7502):612-616.
93. Berendonk TU, Manaia CM, Merlin C, Fatta-Kassinos D, Cytryn E, Walsh F, Bürgmann H, Sørum H, Norström M, Pons M-N: **Tackling antibiotic resistance: the environmental framework**. *Nature Reviews Microbiology* 2015, **13**(5):310-317.
94. Pruden A, Larsson DJ, Amézquita A, Collignon P, Brandt KK, Graham DW, Lazorchak JM, Suzuki S, Silley P, Snape JR: **Management options for reducing the release of antibiotics and antibiotic resistance genes to the environment**. *Environmental Health Perspectives (Online)* 2013, **121**(8):878.
95. Fahrenfeld N, Knowlton K, Krometis LA, Hession WC, Xia K, Lipscomb E, Libuit K, Green BL, Pruden A: **Effect of manure application on abundance of antibiotic resistance genes and their attenuation rates in soil: field-scale mass balance approach**. *Environmental science & technology* 2014, **48**(5):2643-2650.
96. Mao D, Yu S, Rysz M, Luo Y, Yang F, Li F, Hou J, Mu Q, Alvarez P: **Prevalence and proliferation of antibiotic resistance genes in two municipal wastewater treatment plants**. *Water research* 2015, **85**:458-466.
97. Pal C, Bengtsson-Palme J, Kristiansson E, Larsson DJ: **Co-occurrence of resistance genes to antibiotics, biocides and metals reveals novel insights into their co-selection potential**. *BMC genomics* 2015, **16**(1):964.
98. Wallace PJ, Shah ND, Dennen T, Bleicher PA, Crown WH: **Optum Labs: building a novel node in the learning health care system**. *Health Aff (Millwood)* 2014, **33**(7):1187-1194.
99. Zankari E, Hasman H, Cosentino S, Vestergaard M, Rasmussen S, Lund O, Aarestrup FM, Larsen MV: **Identification of acquired antimicrobial resistance genes**. *J Antimicrob Chemother* 2012, **67**(11):2640-2644.

100. Davis JJ, Boisvert S, Brettin T, Kenyon RW, Mao C, Olson R, Overbeek R, Santerre J, Shukla M, Wattam AR *et al*: **Antimicrobial Resistance Prediction in PATRIC and RAST**. *Sci Rep* 2016, **6**:27930.
101. McArthur AG, Tsang KK: **Antimicrobial resistance surveillance in the genomic age**. *Ann N Y Acad Sci* 2016.
102. Rowe W, Baker KS, Verner-Jeffreys D, Baker-Austin C, Ryan JJ, Maskell D, Pearce G: **Search Engine for Antimicrobial Resistance: A Cloud Compatible Pipeline and Web Interface for Rapidly Detecting Antimicrobial Resistance Genes Directly from Sequence Data**. *PLoS One* 2015, **10**(7):e0133492.
103. Bradley P, Gordon NC, Walker TM, Dunn L, Heys S, Huang B, Earle S, Pankhurst LJ, Anson L, de Cesare M *et al*: **Rapid antibiotic-resistance predictions from genome sequence data for Staphylococcus aureus and Mycobacterium tuberculosis**. *Nat Commun* 2015, **6**:10063.
104. Yang Y, Jiang X, Chai B, Ma L, Li B, Zhang A, Cole JR, Tiedje JM, Zhang T: **ARGs-OAP: online analysis pipeline for antibiotic resistance genes detection from metagenomic data using an integrated structured ARG-database**. *Bioinformatics* 2016, **32**(15):2346-2351.
105. Apweiler R, Bairoch A, Wu CH, Barker WC, Boeckmann B, Ferro S, Gasteiger E, Huang H, Lopez R, Magrane M *et al*: **UniProt: the Universal Protein knowledgebase**. *Nucleic Acids Res* 2004, **32**(Database issue):D115-119.
106. Liu X, Song JL, Wang SH, Zhao JW, Chen YQ: **Learning to Diagnose Cirrhosis with Liver Capsule Guided Ultrasound Image Classification**. *Sensors (Basel)* 2017, **17**(1).
107. Liu Q, Liao X, Carin HL, Stack JR, Carin L: **Semisupervised multitask learning**. *IEEE Trans Pattern Anal Mach Intell* 2009, **31**(6):1074-1086.
108. Xavier BB, Das AJ, Cochrane G, De Ganck S, Kumar-Singh S, Aarestrup FM, Goossens H, Malhotra-Kumar S: **Consolidating and exploring antibiotic resistance gene data resources**. *Journal of clinical microbiology* 2016, **54**(4):851-859.
109. LeCun Y, Bengio Y, Hinton G: **Deep learning**. *Nature* 2015, **521**(7553):436-444.
110. Tabar YR, Halici U: **A novel deep learning approach for classification of EEG motor imagery signals**. *J Neural Eng* 2017, **14**(1):016003.
111. Salakhutdinov R, Hinton G: **An efficient learning procedure for deep Boltzmann machines**. *Neural Comput* 2012, **24**(8):1967-2006.
112. Alipanahi B, Delong A, Weirauch MT, Frey BJ: **Predicting the sequence specificities of DNA- and RNA-binding proteins by deep learning**. *Nat Biotechnol* 2015, **33**(8):831-838.
113. Jiang M, Pan Z, Tang Z: **Visual Object Tracking Based on Cross-Modality Gaussian-Bernoulli Deep Boltzmann Machines with RGB-D Sensors**. *Sensors (Basel)* 2017, **17**(1).

114. Huang X, Li KF, Du J, Li R: **Effects of gas supersaturation on lethality and avoidance responses in juvenile rock carp (Procypris rabaudi Tchang)**. *J Zhejiang Univ Sci B* 2010, **11**(10):806-811.
115. Eyre TA, Ducluzeau F, Sneddon TP, Povey S, Bruford EA, Lush MJ: **The HUGO Gene Nomenclature Database, 2006 updates**. *Nucleic Acids Res* 2006, **34**(Database issue):D319-321.
116. Yujian L, Bo L: **A normalized Levenshtein distance metric**. *IEEE Trans Pattern Anal Mach Intell* 2007, **29**(6):1091-1095.
117. Healy MD: **Using BLAST for performing sequence alignment**. *Curr Protoc Hum Genet* 2007, **Chapter 6**:Unit 6 8.
118. Li L-G, Xia Y, Zhang T: **Co-occurrence of antibiotic and metal resistance genes revealed in complete genome collection**. *The ISME journal* 2017, **11**(3):651-662.
119. Sorensen L, Loog M, Lo P, Ashraf H, Dirksen A, Duin RP, de Bruijne M: **Image dissimilarity-based quantification of lung disease from CT**. *Med Image Comput Comput Assist Interv* 2010, **13**(Pt 1):37-44.
120. Min S, Lee B, Yoon S: **Deep learning in bioinformatics**. *Brief Bioinform* 2016.
121. Coates A, Ng A, Lee H: **An analysis of single-layer networks in unsupervised feature learning**. In: *Proceedings of the fourteenth international conference on artificial intelligence and statistics: 2011*. 215-223.
122. Sun Y, Wang X, Tang X: **Deep learning face representation from predicting 10,000 classes**. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition: 2014*. 1891-1898.
123. Buggenthin F, Buettner F, Hoppe PS, Endele M, Kroiss M, Strasser M, Schwarzfischer M, Loeffler D, Kokkaliaris KD, Hilsenbeck O *et al*: **Prospective identification of hematopoietic lineage choice by deep learning**. *Nat Methods* 2017, **14**(4):403-406.
124. Qin Q, Feng J: **Imputation for transcription factor binding predictions based on deep learning**. *PLoS Comput Biol* 2017, **13**(2):e1005403.
125. Dong X, Qian L, Guan Y, Huang L, Yu Q, Yang J: **A multiclass classification method based on deep learning for named entity recognition in electronic medical records**. In: *Scientific Data Summit (NYSDS), 2016 New York: 2016*. IEEE: 1-10.
126. Bhatkoti P, Paul M: **Early diagnosis of Alzheimer's disease: A multi-class deep learning framework with modified k-sparse autoencoder classification**. In: *Image and Vision Computing New Zealand (IVCNZ), 2016 International Conference on: 2016*. IEEE: 1-5.
127. Baldi P, Sadowski P, Whiteson D: **Searching for exotic particles in high-energy physics with deep learning**. *Nat Commun* 2014, **5**:4308.
128. Dunne RA, Campbell NA: **On the pairing of the softmax activation and cross-entropy penalty functions and the derivation of the softmax activation function**. In: *Proc 8th Aust Conf on the Neural Networks, Melbourne, 181: 1997*.

129. Sundermeyer M, Schlüter R, Ney H: **LSTM Neural Networks for Language Modeling**. In: *Interspeech: 2012*. 194-197.
130. Van Merriënboer B, Bahdanau D, Dumoulin V, Serdyuk D, Warde-Farley D, Chorowski J, Bengio Y: **Blocks and fuel: Frameworks for deep learning**. *arXiv preprint arXiv:150600619* 2015.
131. Bergstra J, Bastien F, Breuleux O, Lamblin P, Pascanu R, Delalleau O, Desjardins G, Warde-Farley D, Goodfellow I, Bergeron A: **Theano: Deep learning on gpus with python**. In: *NIPS 2011, BigLearning Workshop, Granada, Spain: 2011*. Citeseer.
132. Qin J, Li R, Raes J, Arumugam M, Burgdorf KS, Manichanh C, Nielsen T, Pons N, Levenez F, Yamada T: **A human gut microbial gene catalogue established by metagenomic sequencing**. *nature* 2010, **464**(7285):59-65.
133. Chen C-M, Ke S-C, Li C-R, Wu Y-C, Chen T-H, Lai C-H, Wu X-X, Wu L-T: **High Diversity of Antimicrobial Resistance Genes, Class 1 Integrons, and Genotypes of Multidrug-Resistant Escherichia coli in Beef Carcasses**. *Microbial Drug Resistance* 2017.
134. Linhares I, Raposo T, Rodrigues A, Almeida A: **Incidence and diversity of antimicrobial multidrug resistance profiles of uropathogenic bacteria**. *BioMed research international* 2015, **2015**.
135. Berglund F, Marathe NP, Österlund T, Bengtsson-Palme J, Kotsakis S, Flach C-F, Larsson DJ, Kristiansson E: **Identification of 76 novel B1 metallo-β-lactamases through large-scale screening of genomic and metagenomic data**. *Microbiome* 2017, **5**(1):134.
136. Bengtsson-Palme J, Kristiansson E, Larsson DJ: **Environmental factors influencing the development and spread of antibiotic resistance**. *FEMS microbiology reviews* 2017, **42**(1):fux053.
137. Wattam AR, Abraham D, Dalay O, Disz TL, Driscoll T, Gabbard JL, Gillespie JJ, Gough R, Hix D, Kenyon R *et al*: **PATRIC, the bacterial bioinformatics database and analysis resource**. *Nucleic Acids Res* 2014, **42**(Database issue):D581-591.
138. O'Neill J: **Tackling drug-resistant infections globally: final report and recommendations.** *The review on antimicrobial resistance* 2016.
139. Pires D, de Kraker MEA, Tartari E, Abbas M, Pittet D: **'Fight Antibiotic Resistance—It's in Your Hands': Call From the World Health Organization for 5th May 2017**. *Clinical Infectious Diseases* 2017, **64**(12):1780-1783.
140. Baquero F, Martínez J-L, Cantón R: **Antibiotics and antibiotic resistance in water environments**. *Current opinion in biotechnology* 2008, **19**(3):260-265.
141. Gibson MK, Forsberg KJ, Dantas G: **Improved annotation of antibiotic resistance determinants reveals microbial resistomes cluster by ecology**. *ISME J* 2015, **9**(1):207-216.
142. Sello JK: **Mining the antibiotic resistome**. *Chemistry & biology* 2012, **19**(10):1220-1221.

143. Li B, Chen H, Su J-Q, Gillings MR, Chen Q-L, Zhang T, An X-L, Zhu Y-G: **Metagenomics of urban sewage identifies an extensively shared antibiotic resistome in China**. *Microbiome* 2017, **5**(1):84.
144. Garner E, Wallace JS, Argoty GA, Wilkinson C, Fahrenfeld N, Heath LS, Zhang L, Arabi M, Aga DS, Pruden A: **Metagenomic profiling of historic Colorado Front Range flood impact on distribution of riverine antibiotic resistance genes**. *Scientific reports* 2016, **6**:38432.
145. Bengtsson-Palme J, Hammarén R, Pal C, Östman M, Björlenius B, Flach C-F, Fick J, Kristiansson E, Tysklind M, Larsson DJ: **Elucidating selection processes for antibiotic resistance in sewage treatment plants using metagenomics**. *Science of the Total Environment* 2016, **572**:697-712.
146. Islamaj Dogan R, Murray GC, Névéol A, Lu Z: **Understanding PubMed® user search behavior through log analysis**. *Database* 2009, **2009**.
147. Khare R, Good BM, Leaman R, Su AI, Lu Z: **Crowdsourcing in biomedicine: challenges and opportunities**. *Briefings in bioinformatics* 2015, **17**(1):23-32.
148. Lu Z, Kim W, Wilbur WJ: **Evaluation of query expansion using MeSH in PubMed**. *Information retrieval* 2009, **12**(1):69-80.
149. MacLean DL, Heer J: **Identifying medical terms in patient-authored text: a crowdsourcing-based approach**. *Journal of the American Medical Informatics Association* 2013, **20**(6):1120-1127.
150. Good BM, Nanis M, Wu C, Su AI: **Microtask crowdsourcing for disease mention annotation in PubMed abstracts**. In: *Pacific Symposium on Biocomputing Co-Chairs: 2014*. World Scientific: 282-293.
151. Wei C-H, Kao H-Y, Lu Z: **PubTator: a web-based text mining tool for assisting biocuration**. *Nucleic acids research* 2013, **41**(W1):W518-W522.
152. Wei C-H, Harris BR, Li D, Berardini TZ, Huala E, Kao H-Y, Lu Z: **Accelerating literature curation with text-mining tools: a case study of using PubTator to curate genes in PubMed abstracts**. *Database* 2012, **2012**.
153. Arighi CN, Carterette B, Cohen KB, Krallinger M, Wilbur WJ, Fey P, Dodson R, Cooper L, Van Slyke CE, Dahdul W: **An overview of the BioCreative 2012 Workshop Track III: interactive text mining task**. *Database* 2013, **2013**.
154. Good BM, Su AI: **Crowdsourcing for bioinformatics**. *Bioinformatics* 2013, **29**(16):1925-1933.
155. Demerec M, Adelberg E, Clark A, Hartman PE: **A proposal for a uniform nomenclature in bacterial genetics**. *Genetics* 1966, **54**(1):61.
156. Levy SB, McMurry LM, Barbosa TM, Burdett V, Courvalin P, Hillen W, Roberts MC, Rood JI, Taylor DE: **Nomenclature for new tetracycline resistance determinants**. *Antimicrobial agents and chemotherapy* 1999, **43**(6):1523-1524.
157. Hall RM, Schwarz S: **Resistance gene naming and numbering: is it a new gene or not?** *Journal of Antimicrobial Chemotherapy* 2015, **71**(3):569-571.

158. Vanhoof R, Hannecart-Pokorni E, Content J: **Nomenclature of genes encoding aminoglycoside-modifying enzymes**. *Antimicrobial agents and chemotherapy* 1998, **42**(2):483-483.
159. Leplae R, Hebrant A, Wodak SJ, Toussaint A: **ACLAME: a CLAssification of Mobile genetic Elements**. *Nucleic acids research* 2004, **32**(suppl_1):D45-D49.
160. Goldberg Y, Levy O: **word2vec explained: Deriving mikolov et al.'s negative-sampling word-embedding method**. *arXiv preprint arXiv:14023722* 2014.
161. Turian J, Ratinov L, Bengio Y: **Word representations: a simple and general method for semi-supervised learning**. In: *Proceedings of the 48th annual meeting of the association for computational linguistics: 2010*. Association for Computational Linguistics: 384-394.
162. Gillings MR: **Integrons: past, present, and future**. *Microbiology and Molecular Biology Reviews* 2014, **78**(2):257-277.
163. Bush K, Jacoby GA: **Updated functional classification of β-lactamases**. *Antimicrobial agents and chemotherapy* 2010, **54**(3):969-976.
164. Gertz EM, Yu Y-K, Agarwala R, Schäffer AA, Altschul SF: **Composition-based statistics and translated nucleotide searches: improving the TBLASTN module of BLAST**. *BMC biology* 2006, **4**(1):41.
165. Prill RJ, Saez-Rodriguez J, Alexopoulos LG, Sorger PK, Stolovitzky G: **Crowdsourcing network inference: the DREAM predictive signaling network challenge**. *Science signaling* 2011, **4**(189):mr7.
166. Huja S, Oren Y, Trost E, Brzuszkiewicz E, Biran D, Blom J, Goesmann A, Gottschalk G, Hacker J, Ron EZ: **Genomic avenue to avian colisepticemia**. *MBio* 2015, **6**(1):e01681-01614.
167. Paterson DL, Hujer KM, Hujer AM, Yeiser B, Bonomo MD, Rice LB, Bonomo RA: **Extended-spectrum β-lactamases in Klebsiella pneumoniae bloodstream isolates from seven countries: dominance and widespread prevalence of SHV-and CTX-M-type β-lactamases**. *Antimicrobial agents and chemotherapy* 2003, **47**(11):3554-3560.
168. Friedrich M: **WHO Survey Reveals Misconceptions About Antibiotic Resistance**. *Jama* 2016, **315**(3):242-242.
169. Allen HK, Donato J, Wang HH, Cloud-Hansen KA, Davies J, Handelsman J: **Call of the wild: antibiotic resistance genes in natural environments**. *Nature Reviews Microbiology* 2010, **8**(4):251-259.
170. Stalder T, Barraud O, Jové T, Casellas M, Gaschet M, Dagot C, Ploy M-C: **Quantitative and qualitative impact of hospital effluent on dissemination of the integron pool**. *The ISME journal* 2014, **8**(4):768.
171. Soucy SM, Huang J, Gogarten JP: **Horizontal gene transfer: building the web of life**. *Nature Reviews Genetics* 2015, **16**(8):472.

172. von Wintersdorff CJ, Penders J, van Niekerk JM, Mills ND, Majumder S, van Alphen LB, Savelkoul PH, Wolffs PF: **Dissemination of antimicrobial resistance in microbial ecosystems through horizontal gene transfer**. *Frontiers in microbiology* 2016, **7**:173.
173. Stokes HW, Gillings MR: **Gene flow, mobile genetic elements and the recruitment of antibiotic resistance genes into Gram-negative pathogens**. *FEMS microbiology reviews* 2011, **35**(5):790-819.
174. Enright MC, Robinson DA, Randle G, Feil EJ, Grundmann H, Spratt BG: **The evolutionary history of methicillin-resistant Staphylococcus aureus (MRSA)**. *Proceedings of the National Academy of Sciences* 2002, **99**(11):7687-7692.
175. Johnson AP, Woodford N: **Global spread of antibiotic resistance: the example of New Delhi metallo-β-lactamase (NDM)-mediated carbapenem resistance**. *Journal of medical microbiology* 2013, **62**(4):499-513.
176. Marquez-Ortiz RA, Haggerty L, Olarte N, Duarte C, Garza-Ramos U, Silva-Sanchez J, Castro BE, Sim EM, Beltran M, Moncada MV: **Genomic epidemiology of NDM-1-encoding plasmids in Latin American clinical isolates reveals insights into the evolution of multidrug resistance**. *Genome biology and evolution* 2017, **9**(6):1725-1741.
177. Mataseje L, Boyd D, Lefebvre B, Bryce E, Embree J, Gravel D, Katz K, Kibsey P, Kuhn M, Langley J: **Complete sequences of a novel bla NDM-1-harbouring plasmid from Providencia rettgeri and an FII-type plasmid from Klebsiella pneumoniae identified in Canada**. *Journal of Antimicrobial Chemotherapy* 2013, **69**(3):637-642.
178. Schmieder R, Edwards R: **Insights into antibiotic resistance through metagenomic approaches**. *Future microbiology* 2012, **7**(1):73-89.
179. Martínez JL: **Antibiotics and antibiotic resistance genes in natural environments**. *Science* 2008, **321**(5887):365-367.
180. Petrosino JF, Highlander S, Luna RA, Gibbs RA, Versalovic J: **Metagenomic pyrosequencing and microbial identification**. *Clinical chemistry* 2009, **55**(5):856-866.
181. Ye L, Shao M-F, Zhang T, Tong AHY, Lok S: **Analysis of the bacterial community in a laboratory-scale nitrification reactor and a wastewater treatment plant by 454-pyrosequencing**. *Water Research* 2011, **45**(15):4390-4398.
182. Keegan KP, Glass EM, Meyer F: **MG-RAST, a metagenomics service for analysis of microbial community structure and function**. In: *Microbial Environmental Genomics (MEG)*. Springer; 2016: 207-233.
183. McArthur AG, Tsang KK: **Antimicrobial resistance surveillance in the genomic age**. *Annals of the New York Academy of Sciences* 2017, **1388**(1):78-91.
184. Prakash T, Taylor TD: **Functional assignment of metagenomic data: challenges and applications**. *Briefings in bioinformatics* 2012, **13**(6):711-727.
185. Jeck WR, Reinhardt JA, Baltrus DA, Hickenbotham MT, Magrini V, Mardis ER, Dangl JL, Jones CD: **Extending assembly of short DNA sequences to handle error**. *Bioinformatics* 2007, **23**(21):2942-2944.

186. Namiki T, Hachiya T, Tanaka H, Sakakibara Y: **MetaVelvet: an extension of Velvet assembler to de novo metagenome assembly from short sequence reads**. *Nucleic acids research* 2012, **40**(20):e155-e155.
187. Bankevich A, Nurk S, Antipov D, Gurevich AA, Dvorkin M, Kulikov AS, Lesin VM, Nikolenko SI, Pham S, Prjibelski AD: **SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing**. *Journal of computational biology* 2012, **19**(5):455-477.
188. Urban JM, Bliss J, Lawrence CE, Gerbi SA: **Sequencing ultra-long DNA molecules with the Oxford Nanopore MinION**. *bioRxiv* 2015:019281.
189. Jain M, Olsen HE, Paten B, Akeson M: **The Oxford Nanopore MinION: delivery of nanopore sequencing to the genomics community**. *Genome biology* 2016, **17**(1):239.
190. Laver T, Harrison J, O'neill P, Moore K, Farbos A, Paszkiewicz K, Studholme DJ: **Assessing the performance of the oxford nanopore technologies minion**. *Biomolecular detection and quantification* 2015, **3**:1-8.
191. Sović I, Šikić M, Wilm A, Fenlon SN, Chen S, Nagarajan N: **Fast and sensitive mapping of nanopore sequencing reads with GraphMap**. *Nature communications* 2016, **7**.
192. Loman NJ, Quick J, Simpson JT: **A complete bacterial genome assembled de novo using only nanopore sequencing data**. *Nature methods* 2015, **12**(8):733-735.
193. Koren S, Walenz BP, Berlin K, Miller JR, Bergman NH, Phillippy AM: **Canu: scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation**. *bioRxiv* 2017:071282.
194. Greninger AL, Naccache SN, Federman S, Yu G, Mbala P, Bres V, Stryke D, Bouquet J, Somasekar S, Linnen JM: **Rapid metagenomic identification of viral pathogens in clinical samples by real-time nanopore sequencing analysis**. *Genome medicine* 2015, **7**(1):99.
195. Edwards A, Debbonaire AR, Sattler B, Mur LA, Hodson AJ: **Extreme metagenomics using nanopore DNA sequencing: a field report from Svalbard, 78 N**. *bioRxiv* 2016:073965.
196. van der Helm E, Imamovic L, Ellabaan MMH, van Schaik W, Koza A, Sommer MO: **Rapid resistome mapping using nanopore sequencing**. *Nucleic Acids Research* 2017:gkw1328.
197. Schmidt K, Mwaigwisya S, Crossman L, Doumith M, Munroe D, Pires C, Khan A, Woodford N, Saunders N, Wain J: **Identification of bacterial pathogens and antimicrobial resistance directly from clinical urines by nanopore-based metagenomic sequencing**. *Journal of Antimicrobial Chemotherapy* 2017, **72**(1):104-114.
198. Judge K, Harris SR, Reuter S, Parkhill J, Peacock SJ: **Early insights into the potential of the Oxford Nanopore MinION for the detection of antimicrobial resistance genes**. *Journal of Antimicrobial Chemotherapy* 2015, **70**(10):2775-2778.

199. Szabó M, Nagy T, Wilk T, Farkas T, Hegyi A, Olasz F, Kiss J: **Characterization of Two Multidrug-Resistant IncA/C Plasmids from the 1960s by Using the MinION Sequencer Device**. *Antimicrobial Agents and Chemotherapy* 2016, **60**(11):6780-6786.
200. Ashton PM, Nair S, Dallman T, Rubino S, Rabsch W, Mwaigwisya S, Wain J, O'grady J: **MinION nanopore sequencing identifies the position and structure of a bacterial antibiotic resistance island**. *Nature biotechnology* 2015, **33**(3):296-300.
201. Pignatelli M, Moya A: **Evaluating the fidelity of de novo short read metagenomic assembly using simulated data**. *PloS one* 2011, **6**(5):e19984.
202. Fallenbeck N, Picht H-J, Smith M, Freisleben B: **Xen and the art of cluster scheduling**. In: *Proceedings of the 2nd International Workshop on Virtualization Technology in Distributed Computing: 2006*. IEEE Computer Society: 4.
203. Pearson WR: **[5] Rapid and sensitive sequence comparison with FASTP and FASTA**. 1990.
204. Sahoo N: **Sequence Base-calling through Albacore software: A part of the Oxford Nanopore Technology**. 2017.
205. David M, Dursi LJ, Yao D, Boutros PC, Simpson JT: **Nanocall: an open source basecaller for Oxford Nanopore sequencing data**. *Bioinformatics* 2016, **33**(1):49-55.
206. Loman NJ, Quinlan AR: **Poretools: a toolkit for analyzing nanopore sequence data**. *Bioinformatics* 2014, **30**(23):3399-3401.
207. Ma L, Xia Y, Li B, Yang Y, Li L-G, Tiedje JM, Zhang T: **Metagenomic assembly reveals hosts of antibiotic resistance genes and the shared resistome in pig, chicken, and human feces**. *Environmental science & technology* 2015, **50**(1):420-427.
208. Guo J, Li J, Chen H, Bond PL, Yuan Z: **Metagenomic analysis reveals wastewater treatment plants as hotspots of antibiotic resistance genes and mobile genetic elements**. *Water research* 2017, **123**:468-478.
209. Quinlan AR, Hall IM: **BEDTools: a flexible suite of utilities for comparing genomic features**. *Bioinformatics* 2010, **26**(6):841-842.
210. Pal C, Bengtsson-Palme J, Rensing C, Kristiansson E, Larsson DJ: **BacMet: antibacterial biocide and metal resistance genes database**. *Nucleic acids research* 2013, **42**(D1):D737-D743.
211. Zhang AN, Li L-G, Ma L, Gillings MR, Tiedje JM, Zhang T: **Conserved phylogenetic distribution and limited antibiotic resistance of class 1 integrons revealed by assessing the bacterial genome and plasmid collection**. *Microbiome* 2018, **6**(1):130.
212. Li W, Godzik A: **Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences**. *Bioinformatics* 2006, **22**(13):1658-1659.
213. Kim D, Song L, Breitwieser FP, Salzberg SL: **Centrifuge: rapid and sensitive classification of metagenomic sequences**. *Genome research* 2016, **26**(12):1721-1729.
214. Trapnell C, Roberts A, Goff L, Pertea G, Kim D, Kelley DR, Pimentel H, Salzberg SL, Rinn JL, Pachter L: **Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks**. *Nature protocols* 2012, **7**(3):562.

215. Patro R, Mount SM, Kingsford C: **Sailfish enables alignment-free isoform quantification from RNA-seq reads using lightweight algorithms**. *Nature biotechnology* 2014, **32**(5):462.
216. Franz M, Lopes CT, Huck G, Dong Y, Sumer O, Bader GD: **Cytoscape. js: a graph theory library for visualisation and analysis**. *Bioinformatics* 2015, **32**(2):309-311.
217. Tacconelli E, Carrara E, Savoldi A, Harbarth S, Mendelson M, Monnet DL, Pulcini C, Kahlmeter G, Kluytmans J, Carmeli Y: **Discovery, research, and development of new antibiotics: the WHO priority list of antibiotic-resistant bacteria and tuberculosis**. *The Lancet Infectious Diseases* 2017.
218. Santajit S, Indrawattana N: **Mechanisms of antimicrobial resistance in ESKAPE pathogens**. *BioMed research international* 2016, **2016**.
219. Li A-D, Metch JW, Wang Y, Garner E, Zhang AN, Riquelme MV, Vikesland PJ, Pruden A, Zhang T: **Effects of sample preservation and DNA extraction on enumeration of antibiotic resistance genes in wastewater**. *FEMS microbiology ecology* 2017, **94**(2):fix189.
220. Du J, Ren H, Geng J, Zhang Y, Xu K, Ding L: **Occurrence and abundance of tetracycline, sulfonamide resistance genes, and class 1 integron in five wastewater treatment plants**. *Environmental Science and Pollution Research* 2014, **21**(12):7276-7284.
221. Vikesland PJ, Pruden A, Alvarez PJ, Aga D, Burgmann H, Li X-d, Manaia CM, Nambi I, Wigginton K, Zhang T: **Toward a comprehensive strategy to mitigate dissemination of environmental sources of antibiotic resistance**. In.: ACS Publications; 2017.
222. Hyeon J-Y, Li S, Mann DA, Zhang S, Li Z, Chen Y, Deng X: **Quasimetagenomics-Based and Real-Time-Sequencing-Aided Detection and Subtyping of Salmonella enterica from Food Samples**. *Applied and environmental microbiology* 2018, **84**(4):e02340-02317.
223. Gillings M, Boucher Y, Labbate M, Holmes A, Krishnan S, Holley M, Stokes HW: **The evolution of class 1 integrons and the rise of antibiotic resistance**. *Journal of bacteriology* 2008, **190**(14):5095-5100.
224. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ: **Basic local alignment search tool**. *Journal of molecular biology* 1990, **215**(3):403-410.
225. Finn RD, Clements J, Eddy SR: **HMMER web server: interactive sequence similarity searching**. *Nucleic acids research* 2011, **39**(suppl_2):W29-W37.
226. BLAST G: **PSI-BLAST: a new generation of protein database search programs Altschul**. *Stephen F* 1997:3389-3402.
227. Li H, Homer N: **A survey of sequence alignment algorithms for next-generation sequencing**. *Briefings in bioinformatics* 2010, **11**(5):473-483.
228. Zielezinski A, Vinga S, Almeida J, Karlowski WM: **Alignment-free sequence comparison: benefits, applications, and tools**. *Genome biology* 2017, **18**(1):186.

229. Kent WJ: **BLAT—the BLAST-like alignment tool**. *Genome research* 2002, **12**(4):656-664.
230. Edgar R: **USEARCH: ultra-fast sequence analysis**. In.; 2015.
231. Ye Y, Choi J-H, Tang H: **RAPSearch: a fast protein similarity search tool for short reads**. *BMC bioinformatics* 2011, **12**(1):159.
232. Pearson WR: **Searching protein sequence libraries: comparison of the sensitivity and selectivity of the Smith-Waterman and FASTA algorithms**. *Genomics* 1991, **11**(3):635-650.
233. Weijers S, De Jonge J, Van Zanten O, Benedetti L, Langeveld J, Menkveld H, Van Nieuwenhuijzen A: **KALLISTO: cost effective and integrated optimization of the urban wastewater system Eindhoven**. *Water Practice and Technology* 2012, **7**(2):wpt2012036.
234. Patro R, Duggal G, Kingsford C: **Accurate, fast, and model-aware transcript expression quantification with Salmon**. *bioRxiv* 2015, **21592**.
235. Zhang Z, Wang W: **RNA-Skim: a rapid method for RNA-Seq quantification at transcript level**. *Bioinformatics* 2014, **30**(12):i283-i292.
236. Li Y, Heavican TB, Vellichirammal NN, Iqbal J, Guda C: **ChimeRScope: a novel alignment-free algorithm for fusion transcript prediction using paired-end RNA-Seq data**. *Nucleic acids research* 2017:gkx315.
237. Pajuste F-D, Kaplinski L, Möls M, Puurand T, Lepamets M, Remm M: **FastGT: an alignment-free method for calling common SNVs directly from raw sequencing reads**. *Scientific reports* 2017, **7**(1):2537.
238. Li H: **Minimap and miniasm: fast mapping and de novo assembly for noisy long sequences**. *Bioinformatics* 2016, **32**(14):2103-2110.
239. Berlin K, Koren S, Chin C-S, Drake JP, Landolin JM, Phillippy AM: **Assembling large genomes with single-molecule sequencing and locality-sensitive hashing**. *Nature biotechnology* 2015, **33**(6):623.
240. Wood DE, Salzberg SL: **Kraken: ultrafast metagenomic sequence classification using exact alignments**. *Genome biology* 2014, **15**(3):R46.
241. Ondov BD, Treangen TJ, Melsted P, Mallonee AB, Bergman NH, Koren S, Phillippy AM: **Mash: fast genome and metagenome distance estimation using MinHash**. *Genome biology* 2016, **17**(1):132.
242. Ounit R, Wanamaker S, Close TJ, Lonardi S: **CLARK: fast and accurate classification of metagenomic and genomic sequences using discriminative k-mers**. *BMC genomics* 2015, **16**(1):236.
243. Gupta A, Jordan IK, Rishishwar L: **stringMLST: a fast k-mer based tool for multilocus sequence typing**. *Bioinformatics* 2016, **33**(1):119-121.
244. Ng P: **dna2vec: Consistent vector representations of variable-length k-mers**. *arXiv preprint arXiv:170106279* 2017.

245. Yang KK, Wu Z, Bedbrook CN, Arnold FH: **Learned protein embeddings for machine learning**. *Bioinformatics* 2018, **34**(15):2642-2648.
246. Zhao Y, Tang H, Ye Y: **RAPSearch2: a fast and memory-efficient protein similarity search tool for next-generation sequencing data**. *Bioinformatics* 2011, **28**(1):125-126.
247. Maaten Lvd, Hinton G: **Visualizing data using t-SNE**. *Journal of machine learning research* 2008, **9**(Nov):2579-2605.