

DeepARG+ - A Computational Pipeline for the Prediction of Antibiotic Resistance Genes

Rutwik Shashank Kulkarni

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

Master of Science
in
Computer Science and Applications

Liqing Zhang, Chair
Amy Pruden-Bagchi
Anuj Karpatne

May 06, 2021
Blacksburg, Virginia

Keywords: Antibiotic Resistance, Deep Learning, Machine Learning, Protein Structure

Copyright 2021, Rutwik Shashank Kulkarni

DeepARG+ - A Computational Pipeline for the Prediction of Antibiotic Resistance Genes

Rutwik Shashank Kulkarni

(ABSTRACT)

The global spread of antibiotic resistance warrants concerted surveillance in the clinic and in the environment. The widespread use of metagenomics for various studies has led to the generation of a large amount of sequencing data. Next-generation sequencing of microbial communities provides an opportunity for proactive detection of emerging antibiotic resistance genes (ARGs) from such data, but there are a limited number of pipelines that enable the identification of novel ARGs belonging to diverse antibiotic classes at present. Therefore, there is a need for the development of computational pipelines that can identify these putative novel ARGs. Such pipelines should be scalable, accessible and have good performance. To address this problem we develop a new method for predicting novel ARGs from genomic or metagenomic sequences, leveraging known ARGs of different resistance categories. Our method takes into account the physio-chemical properties that are intrinsic to different ARG families. Traditionally, new ARGs are predicted by making sequence alignment and calculating sequence similarity to existing ARG reference databases, which can be very time consuming. Here we introduce an alignment free and deep learning prediction method that incorporates both the primary protein sequences of ARGs and their physio-chemical properties. We compare our method with existing pipelines including hidden Markov model based Resfams and fARGene, sequence alignment and machine learning-based DeepARG-LS, and homology modelling based Pairwise Comparative Modelling. We also use our model to detect novel ARGs from various environments including human-gut, soil, activated sludge

and the influent samples collected from a waste water treatment plant. Results show that our method achieves greater accuracy compared to existing models for the prediction of ARGs and enables the detection of putative novel ARGs, providing promising targets for experimental characterization to the scientific community.

DeepARG+ - A Computational Pipeline for the Prediction of Antibiotic Resistance Genes

Rutwik Shashank Kulkarni

(GENERAL AUDIENCE ABSTRACT)

Various bacteria contain genes that allow them to survive and grow even after the application of antibiotics. Such genes are called antibiotic resistance genes (ARGs). Each ARG has properties that make it resistant to a particular class of antibiotics. This class is called the resistance class/category of the gene. Antimicrobial resistance (AMR) is one of the biggest challenges to public health in recent times. It has been projected that a large number of deaths might occur due to AMR in the future. Therefore, there is a need for monitoring AMR in various environments. Currently, developed methods use the sequence's similarity with the existing database as a feature for ARG prediction. Some tools also use the 3D structure of proteins as a feature for ARG prediction. In this thesis, we develop a tool that incorporates both the sequence similarity and the structural information of proteins for ARG prediction. The structural information is encoded with physio-chemical properties (such as hydrophobicity, molecular weight etc.) of the amino acids. Our results show the efficacy of the pipeline in various environments. Results also show that our method achieves accuracy greater than existing models for the prediction of ARGs from metagenomic data. It also enables the detection of putative novel ARGs, providing promising targets for experimental characterization to the scientific community.

Dedication

To my parents, Shashank Kulkarni and Manjiri Kulkarni, for inculcating the virtues of patience and hardwork and to my grandfather, Shridhar Kulkarni, for teaching me the importance of consistency and punctuality.

Also to the antibiotic resistant bacteria who have taught me to never give up in life

Acknowledgments

It would not have been possible for me to complete Master's program without the support of countless helping hands. First of all, I would like to thank my advisor, Dr. Liqing Zhang, for believing in me and allowing me to explore the research field even when I was not confident in my abilities. I am grateful for her constant guidance and support throughout my Master's program.

I would also like to thank Dr. Amy Pruden for her guidance and help in understanding the domain problems, which is very important for any interdisciplinary research.

I would like to thank Dr. Anuj Karpatne for guidance in research and valuable feedback.

My special thanks to Suraj Gupta and Connor Brown for helping me whenever I had any issue in understanding the concepts of bioinformatics or any implementation tool. Connor also helped me add additional validation using a phylogenetic tree for our model.

I am grateful To Dr. Gustavo Arango-Argoty for his invaluable guidance and help in conceptualization.

I am obliged to Dr. Dongjuan Dai for providing the nanopore sequences essential for analyzing the platform independence.

I would like to specially thank all the members of Zhang Lab and Pruden Lab for providing me with valuable suggestions for the improvement of this work.

I want to express my deepest gratitude to my family and all my friends, old and new. Thank you for being supportive and for motivating me in times of need. Completing this work without you all would have been very difficult.

Contents

List of Figures	x
List of Tables	xii
1 Introduction	1
1.1 Rise of Antibiotic Resistance	1
1.2 Transfer of Antibiotic Resistance	2
1.3 Antibiotic Resistance in the Environment	3
1.4 Metagenomic Sequencing Data	3
1.5 Computational Pipelines for ARG Prediction	4
1.5.1 Alignment based tools	4
1.5.2 Alignment Free Tools	5
1.6 Protein Structure	6
1.7 NLP in Bioinformatics	7
1.8 Validation of predicted ARGs	8
2 Methods	10
2.1 Database Curation	10
2.2 Workflow for training the deep neural networks model	11

2.2.1	fastText	12
2.2.2	Sequence Embedding Generation	13
2.2.3	Generation of Feature vector based on Physio-chemical Properties	14
2.3	Novel Database Creation	15
2.4	Prediction Pipeline of DeepARG+	16
2.5	Phylogenetic Analysis	16
3	Results	19
3.1	Evaluation on Test Data	19
3.2	Evaluation on Novel beta-lactamases	20
3.3	Evaluation on Soil Dataset	21
3.4	Evaluation on Human Gut Data	23
3.4.1	PCM based predictions	24
3.5	Analysis of Nanopore Sequences in Influent and Activated Sludge data	26
3.6	Performance Analysis of DeepARG+	28
3.7	Phylogenetic Analysis	29
4	Discussion	31
5	Conclusion	33
6	Future Work	34

Bibliography	36
Appendices	46
Appendix A DeepARG+ implementation	47
A.1 Setting up the requirements	47
A.2 Training the model	47
A.2.1 Training fastText model	47
A.3 Prediction	48
Appendix B Supplementary Results	49
B.1 Distribution of ARG classes	49
B.2 Distribution of lengths of sequences	50
B.3 Experiments	50
B.3.1 Independent feature vector based models	50
B.3.2 Ensemble based models	51
B.3.3 Multichannel CNNs	51
B.3.4 Model optimization	52
B.3.5 Analysis of fastText Feature Vectors	52
B.4 Results on soil data	53
B.5 Evaluation on 71 validated ARGs by Ruppe <i>et al.</i> [67]	56

List of Figures

1.1	Brief History of Antibiotic Resistance	2
2.1	Database Curation	11
2.2	Workflow for training DeepARG+	12
2.3	Steps for creating model based on fastText word embeddings	14
2.4	Word embeddings on test data	15
2.5	Prediction Pipeline of DeepARG+	17
2.6	Maximum likelihood phylogeny of TetO-, TetP-, TetS-, TetM-, and TetQ-type tetracycline resistance proteins. Highlighted sequences represent known ARG protein sequences (green: TetQ/TetP/TetS; grey: TetM)	18
3.1	Confusion matrix on test data	20
3.2	Precision on different pipelines (NA part indicates absence of the class in data of pipeline)	22
3.3	Recall on different pipelines (NA part indicates absence of the class in data of pipeline)	22
3.4	Number of sequences predicted by different pipelines on the gutdata)	23
3.5	Intersection of DeepARG+ and BLAST with different pipelines	24
3.6	Metrics on functionally tested data by PCM	25

3.7	Percentage of ARGs predicted by DeepARG+ in total genes present in sludge samples (grouped by region)	27
3.8	Percentage of ARGs predicted by DeepARG+ in total genes present in influent samples (grouped by region)	28
3.9	Percentage of ARGs predicted by all the pipelines in total genes present in influent and sludge samples (grouped by region)	29
6.1	Planned system based on expert based validation	35
B.1	Distribution of classes in database	49
B.2	Distribution of lengths of all sequences in database	50
B.3	fastText vectors on validation data for kmer size 20	53
B.4	fastText vectors on validation data for kmer size 8	54
B.5	fastText vectors on validation data for kmer size 6	55
B.6	Confusion Matrix of DeepARG+ on Soil Data	57
B.7	Percentage Identity of correctly classified genes by DeepARG+	58
B.8	Percentage Identity of correctly classified genes by PCM	59
B.9	Percentage Identity of correctly classified genes by DeepARG-LS	60

List of Tables

3.1	Time Analysis of DeepARG+	28
B.1	Model Optimization	52
B.2	Metrics for soil data	56

List of Abbreviations

AMR Antimicrobial Resistance

AR Antibiotic Resistance

ARG Antibiotic Resistance Gene

NLP Natural Language Processing

Chapter 1

Introduction

Bacteria can survive and multiply despite the use of antibiotics. This phenomenon is called antibiotic resistance (AR). AR is one of the biggest challenges to public health in recent times. According to the Centers for Disease Control and Prevention (CDC), at least 2.8 million people get infected with AR bacteria every year in the US [4]. More than 35,000 people die annually due to an AR infection [4]. The appearance of new AR bacteria in the environment typically follows the release of novel antibiotics. For example, penicillin was discovered in 1928 and released in 1941. In 1942, researchers identified penicillin-resistant *Staphylococcus aureus* bacteria for the first time [71]. It is difficult to treat AR infections. According to a recent study [54], more than 4.6 billion dollars will be required annually to treat AR infections caused by multidrug resistant bacteria. The rise of AR is due to the overuse and misuse of antibiotics for the treatment of bacterial infections.

1.1 Rise of Antibiotic Resistance

The rise of antibiotic resistance in parallel with the development of antibiotics can be seen from figure 1.1. After the discovery of penicillin and the consequent resistance, antibiotics were developed in response to the resistance problem. In the golden age of antibiotics (1950-1970), about 20 new antibiotics were discovered. Antibiotics such as methicillin, sulfonamide and tetracyclines were developed in this period. However, bacteria resistant to each of the

newly discovered antibiotic were soon found [66]. In 2008, a super bug called NDM-1 (New Delhi metallo- β -lactamase-1) was discovered [57]. This bacteria is resistant to the strongest and most commonly used beta-lactam antibiotics. In 2017, scientists developed Teixobactin [2] that kills superbugs. In recent times, the development of antibiotics has slowed down and therefore AR has become a big threat [37] [3].

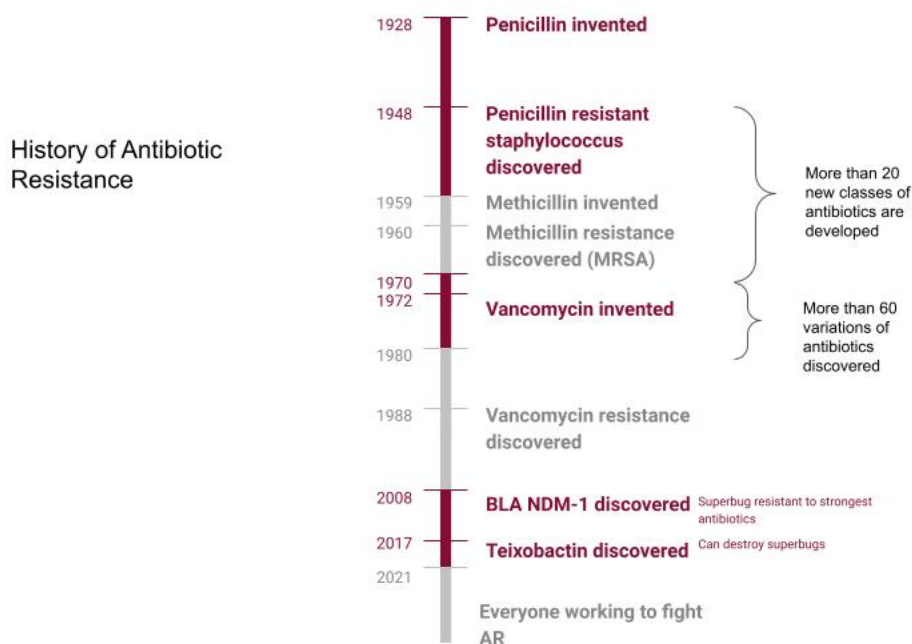


Figure 1.1: Brief History of Antibiotic Resistance

1.2 Transfer of Antibiotic Resistance

Bacteria mutate and acquire genes that confer AR either horizontally (transfer from bacteria to bacteria), or vertically (by evolution). Horizontal Gene Transfer (HGT) is the most common route of AR transfer [53]. HGT occurs by 4 routes [36] [58].

- Transportation: The plasmids containing the DNA of a resistant bacterium are trans-

mitted to another bacterium.

- Transduction: Bacteriophage infects bacteria and in the course of that infection it can insert some of its own DNA which might contain an ARG, making the bacteria antibiotic resistant.
- Conjugation: Bacterial “sex” which leads to the transfer of antibiotic resistance from one bacteria to another.
- Transformation: It is the ability of bacteria to take naked DNA from the environment and develop antibiotic resistance.

1.3 Antibiotic Resistance in the Environment

Daves [16], in his work, stated that the resistance in bacteria is not restricted to pathogens. Resistance is also present in environmental microbes. Bacteria interact with various chemicals in their respective environments and act as a reservoir of antibiotic resistance. Studies such as [27] and [28] study the antibiotic resistance in the soil environment. Additionally, water has also been screened for antibiotic resistance by studies including [75] and [12]. ARGs have also been found in animals, human guts and even insects [72]. As AR bacteria are prevalent in all environments we need to effectively monitor their spread.

1.4 Metagenomic Sequencing Data

The study of AR in bacteria has been greatly promoted by the development of next-generation sequencing (NGS) technology that allows scientists to rapidly sequence bacterial

genomes. Metagenomic sequencing can generate tens of millions of sequences from many bacterial genomes and has allowed researchers to explore ARG in various environments. In order to identify ARGs, researchers collect samples from the environment, extract DNAs, prepare a library and then use a sequencing platform for metagenomic sequencing [22]. DNAs are fragmented and sequenced to produce short reads/sequences ranging from 75 bps to 400 bps. These short reads can be assembled into contigs by various short read assembly algorithms [22]. However, with the rise of the third generation sequencing, sequences up to 30 kilobase pairs can be generated [6]. Both Oxford Nanopore sequencing technology [21] and Pacbio [64] can produce longer reads but have high sequencing error rates [7].

After generating metagenomic sequencing data or genomic sequences, researchers will apply computational pipelines for annotating or predicting ARGs in the data. The detailed algorithms and tools are reviewed next.

1.5 Computational Pipelines for ARG Prediction

1.5.1 Alignment based tools

Computational tools such as BLAST [43] and DIAMOND [23] use the ‘best-hit’ approach for annotation of new sequences. This approach calculates similarity of the new sequence to existing sequences and uses this to predict the resistance category of the new sequence. One of the main reasons for the lack of good prediction is that this technique considers only the sequence similarity for annotation. Best-hit approaches require sequence alignment, a very time-consuming process. Alignment based algorithms use a “seed-and-extend” strategy, which involves an exact match/search of the seed DNA sequence in the reference database (a database with known ARGs) as the first step and then extension of the match to the

entire input string using a sequence alignment algorithm such as Smith-Waterman [68] as the second step. Widely used computer programs such as BLAST [43] and FASTA [61] [60] [63] fall into this category. BLAST has grown to become the most trusted tool for sequence search and alignment. However, BLAST, like all other alignment-based algorithms, does not scale well when comparing hundreds of thousands of sequences. This is because the sequence alignment step uses dynamic programming, which takes $O(n^2)$ time, where n is the length of the sequences aligned. As a result, variants of BLAST-type tools have been developed such as DIAMOND [23], BLAT [41], USEARCH [30], and RAPSearch [74] [76]. Particularly, the dramatic speed up of DIAMOND (20,000 \times) is achieved using a double indexing/hashing strategy (hashing both input sequences and reference database sequences), spaced seeds (longer seeds where not all positions are used), and a reduced alphabet. Comparatively, these newer tools run much faster than the BLAST family of tools, but with some loss of sensitivity.

1.5.2 Alignment Free Tools

Alignment free tools do not use sequence alignment for ARG prediction. These pipelines are based on Hidden Markov Models (HMMs) and deep learning. Some of them concentrate on the annotation of resistant genes belonging to specific classes of interest. For instance, Berglund *et al.* [18], developed a Hidden Markov Model (HMM)-based system (fARGene) to predict genes resistant to various subclasses of beta-lactam antibiotics. HMMs were optimized for the prediction of the resistant class from the short read metagenomic data. Other pipelines such as Resfinder [42], SEAR [65] and Mykrobyobe predictor [20] are used to annotate genes from plasmids, and various resistance classes. Such specific-annotation models provide better sensitivity compared to generic models. Recently developed pipelines such as DeepARG [10] and Resfams [33] annotate genes conferring resistance to up to 30

antibiotic classes. Resfams uses a database of HMMs each optimized for annotation of genes conferring resistance to different antibiotic classes. DeepARG generates a feature vector for a new sequence based on its similarity to known sequences belonging to 30 classes of antibiotics. A deep neural network predicts the class of resistant genes based on this feature vector. DeepARG also consists of different models optimized for annotation of short read metagenomic data (DeepARG-SS) and full length genomic sequences (DeepARG-LS) [10]. All the pipelines discussed above annotate the genes based on sequence similarity with known genes in existing databases. A different approach, exemplified in PCM (Pairwise Comparative Modelling) [67], a pipeline developed by Ruppe *et al.*, used the 3D structure of proteins to annotate ARGs in the intestinal gut microbiota. Their results show an improved accuracy over the similarity based models. PCM also found new gene sequences that have less than 40 percent identity to the known sequences. Thus, the structural properties help the identification of the unknown resistant gene sequences. However, the process of modelling 3D structure of proteins is computationally expensive.

1.6 Protein Structure

Proteins are evolutionarily conserved more at a structural level than the primary amino acid sequence level [39]. Therefore, structural information as expressed in the physio-chemical properties of the protein should retain useful information relevant to annotation. Other studies such as [59] have been conducted that utilize the physio-chemical properties of the amino acids to predict the antimicrobial activity of a drug. Additionally, the classification of drugs into known classes based on these properties has also been demonstrated. Therefore, we hypothesized that any model employing physio-chemical properties retains the value of the technique introduced in PCM. Most studies [10] assume percentage identity as the basis

of similarity in homology. However, according to the study [14] homology is not merely related to percentage identity. Studies such as [62] indicate that homologous sequences can have less than 30 percent sequence identity with known sequences.

1.7 NLP in Bioinformatics

The representation of a word in a vector format is called word embedding. Word embeddings are commonly used as features in many NLP tasks such as text classification. Neural Probabilistic Language model (NPLM) [13] developed by Bengio, *et al.* uses a single layer feed forward neural network for the generation of word embeddings. To avoid the computational cost, Mikolov, *et al.* [51] removed the hidden layer in NPLM and added a hierarchical softmax layer in order to develop the word2vec [51] model. Words that appear in the same context and can be used interchangeably is the basic principle of these models. A DNA or protein sequence can be viewed as a sentence with each kmer as a word [8] [73]. Therefore, we can represent each kmer as a vector. DNA2vec [55], a model based on word2vec, generates distributed representations of variable length kmers. These representations correlate with the results of time consuming sequence alignment process. ProtVec,[11] developed by Asgari *et al.* represents a sequence by a vector created from overlapping kmers of length 3. These features of protein can be used for protein family classification. All these methods follow an unsupervised way of embedding generation. However, some techniques like fastText developed by Facebook allows us to train word embeddings in a supervised way. Supervised learning gives higher accuracy for tasks like text classification.

1.8 Validation of predicted ARGs

In tandem with robust detection of novel ARGs in genomic data, it is essential to provide biological context to the detected genes in order to better inform action strategies. Available knowledge bases such as the CARD [50] provide centralized, well-curated sources for ARG research. However, while the CARD ontologies are important resources, they are not generally rooted in the molecular sequence evolution of ARGs, and, therefore, cannot be used to rigorously classify novel ARGs. In contrast, phylogenetics, the process of inferring the evolution of bacterial taxa or gene/protein sequences, constitutes the most rigorous and fundamental method for constructing biological classification systems. Phylogenetic reconstruction within a gene family typically proceeds in the following steps. First, homologs of the gene family are identified in public databases. Homologs are genes that are present in at least two different organisms and have a shared evolutionary history. Once acquired, representative sequences are selected from the list of homologs (typically hundreds of thousands of sequences) to extract the major subfamilies of the gene family. These representative sequences are then aligned to one another using one of many available multiple sequence alignments (MSA) algorithms, and a phylogenetic tree is reconstructed using an appropriate algorithm. Depending on the research objectives, different stochastic models of sequence evolution can be used to explain the observed patterns of gene family evolution. Multiple models and model parameters are typically evaluated to identify the combination that best explains the data. Finally, overlaying the phylogenetic tree with knowledge of gene function allows for the transfer of functional knowledge from characterized genes to putative novel genes, or even the detection of novel clades (evolutionary distinct clusters of genes) with uncertain function.

Based on the two approaches used in prior work, we hypothesized that a computationally efficient method (fast text embedding) that also leveraged signatures of protein structure would enable rapid and accurate detection of novel ARGs. In this work we propose DeepARG+, a tool that uses embedding-based sequence similarity and also incorporates the physio-chemical properties of amino acids for feature vector generation, and consequently, for ARG annotation. In our model, we use the hydrophobicity of amino acids (a physio-chemical property) to encode structural information. Hydrophobicity helps understand protein folding, secondary structures and membrane associated regions [34]. We found that the use of hydrophobicity helped us better understand the structure of the protein and also its chemical properties. Section B.3.4 discusses the experiments and results with encoding by other physio-chemical properties such as molecular weight, isoelectric point etc. Thus, addition of such features provided us improved models for resistant gene annotation. We use our method to analyze genome sequences in multiple environments and identify novel (previously unknown) ARGs. Thus, DeepARG+ facilitates the effective monitoring of ARGs and also contributes to the identification of previously undiscovered ARG sequences.

Chapter 2

Methods

2.1 Database Curation

The ARG database was created by merging publicly available resistance gene databases like CARD [50], ARDB [49] and ARGMiner[9]. The model must classify a sequence as an ARG or negative sequence before eventually classifying it into resistance classes. The negative sequences are functional proteins which do not confer resistance to any antibiotic. Negative reference gene sequences were included in our dataset by performing sequence alignment on human based proteins in UNIPROT [25] [26]. The negative subset we use has 19,773 human based proteins.

The merged dataset has some duplicate sequences. Therefore CD-HIT [48], a fast clustering algorithm is used to remove redundancy in our genomics dataset. CD-HIT [48] with parameter $c=1$ was used to remove duplicate sequences in the dataset. $c=1$ indicates the grouping of identical sequences in single clusters.

Apart from duplicates, another major problem in database curation is the lack of consistent ARG nomenclature [29]. For example, a gene resistant to diaminopyrimidine antibiotics may have been documented as a gene resistant to trimethoprim (an alias for diaminopyrimidine). Additionally, some databases might annotate sequences to a sub class level. For example, the aminoglycoside gene [70] *aadA1* can be documented as ANT(3)-I, *aadA1*-pm, ANT3-DPRIME, or *ant3* in different ARG databases. Similar patterns are observed for

beta-lactamase and tetracycline gene nomenclature [45] [35]. Therefore, manual curation of the sequences was done in order to create a consistent database annotated at a resistance class level. We curated the dataset to obtain 36,792 gene sequences belonging to 16 antibi-

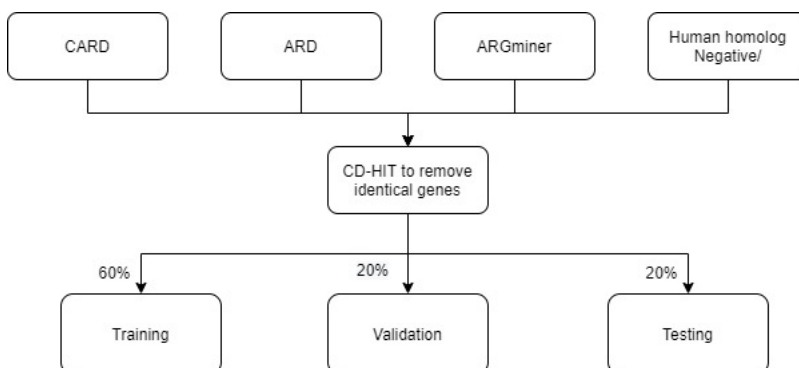


Figure 2.1: Database Curation

otic classes, including MLS, Negative, aminoglycoside, bacitracin, beta-lactam, diaminopyrimidine, fosfomycin, fosmidomycin, glycopeptide, multidrug, peptide, phenicol, polymyxin, quinolone, rifamycin, sulfonamide, and tetracycline. Sequence alignment was done against the list of efflux pumps and all ARGs having greater than 90% identity with these inserts were removed from the database. The distribution of ARGs in the database is shown in figure B.1. This curated dataset is split into training, validation and testing datasets in a 60:20:20 proportion, which were used for building and testing our model. Figure 2.1 shows the process of database curation.

2.2 Workflow for training the deep neural networks model

In order to perform antibiotic resistance gene annotation, we generate feature vectors that incorporate both the similarity with known sequences and physio-chemical properties of

amino acids. Figure 2.2 shows the workflow for training the model. The input to the system is a fasta file consisting of sequences. These sequences are passed through 2 pipelines for feature vector generation. The first feature vector is formed by generating sequence embeddings using the fastText library. A feed-forward neural network is used to extract patterns from the word vectors of the sequences. The second feature vector is formed by converting the original amino acid sequence to a sequence of the values of the physio-chemical properties of the corresponding amino acids. A 1D convolutional neural network is used to extract the patterns from the physio-chemical property vector. Concatenating these models helped us perform a combined analysis for the better annotation of new sequences. The generation of feature vectors is described in detail in consequent sections.

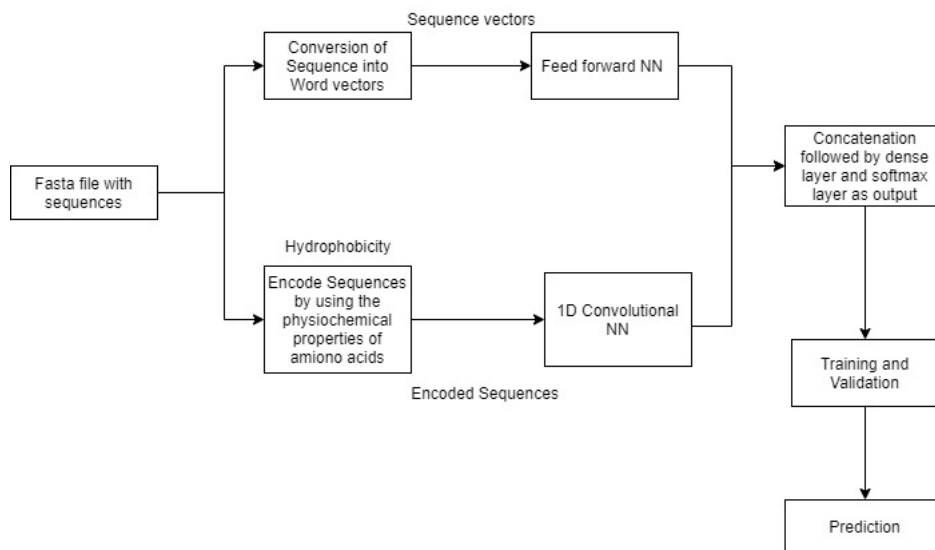


Figure 2.2: Workflow for training DeepARG+

2.2.1 fastText

fastText [19] is a popular model to learn the distributed representation of the words. It was developed by a research group at Facebook. fastText not only uses the entire word and con-

text but also uses the subwords or characters for word embedding generation. For example, a fastText model would use characters such as "Bio", "iol" etc , for generating embedding for the word Biology. This technique is useful in genomics as the word that we consider is a kmer and it does not have any literal meaning. As fastText considers the subwords, we can also consider variable length kmers, that can identify locally conserved patterns in the sequence. Additionally, it is a light-weight model and improves the performance of the pipeline.

2.2.2 Sequence Embedding Generation

The fastText [19] library was used for sequence embedding generation. In order to generate the sequence embeddings, the following steps were followed:

1. Initially, we split gene sequences into non-overlapping kmers of size 20. The kmer size was decided as an hyperparameter after parameter tuning. Results for different kmer sizes are shown in section [B.3.5](#).
2. We generate kmer number of sentences for a sequence using the sliding window technique as shown in figure.
3. Now each sequence can be considered as a document with kmer number of sentences and a label of the resistant antibiotic associated with it
4. Supervised fastText model is trained on these generated documents.
5. This trained model is used to further create 300 dimensional word embeddings for each sequence in the training data.

A similar process is followed to generate embeddings of the test data.

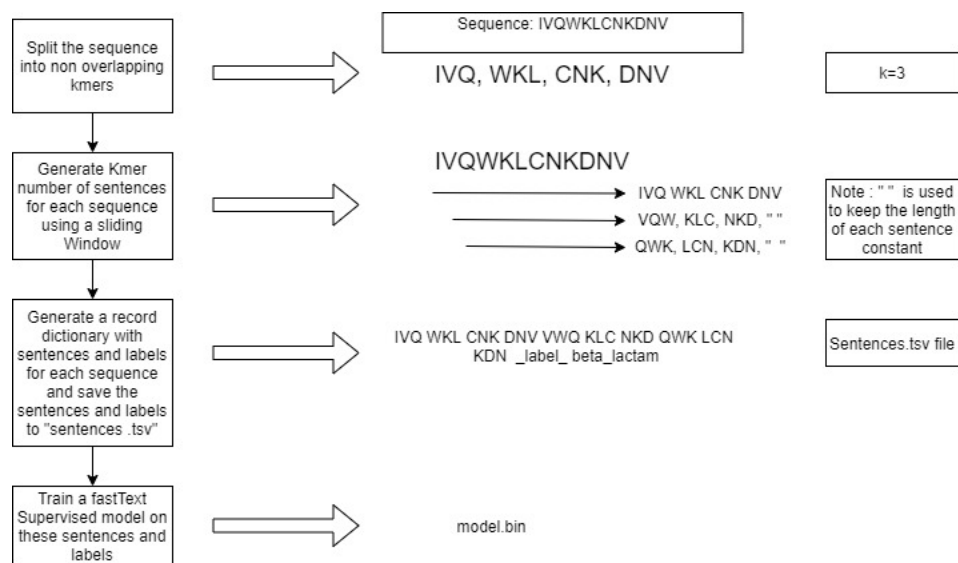


Figure 2.3: Steps for creating model based on fastText word embeddings

Figure 2.4 shows word embeddings generated from the test data. TSNE [69] algorithm is used to reduce dimensionality to facilitate the 2D visualization of embeddings. Figure 2.4 shows the separation among the embeddings of sequences belonging to different classes. Therefore, we can say that our embeddings are powerful enough to get separation of the classes.

2.2.3 Generation of Feature vector based on Physio-chemical Properties

In order to incorporate physio-chemical properties into our model we encode each letter (protein) in the sequence with corresponding physio-chemical property value (e.g. hydrophobicity value). The Kyle and Doolittle [44] scale was used for the generation of the feature vectors. Any machine learning algorithm requires a feature vector of the same length as an input; therefore, the vector length for each sequence was extended to 1000 positions by inputting a 0 where there was not a corresponding amino acid. To decide the optimal length of the input vector we performed analysis of the lengths of the sequences in the data. Most of the

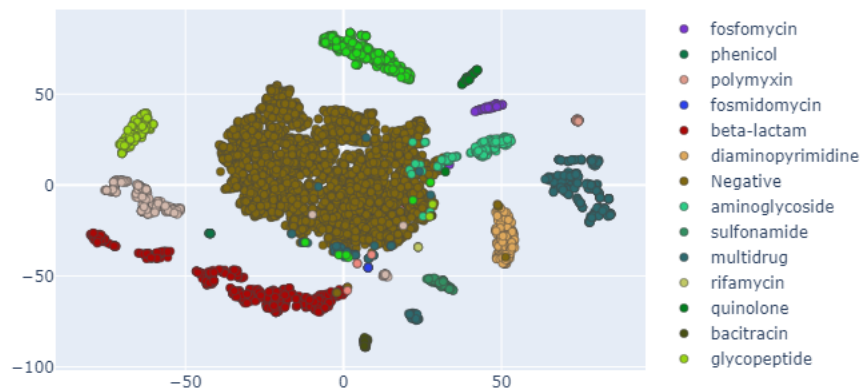


Figure 2.4: Word embeddings on test data

sequences have lengths less than 1000, the input length was chosen to be 1000. Section B.2 shows the distribution of lengths of all sequences in our data. However, some kind of data loss takes place in this step as certain sequences have length greater than 1000. We assume that this data loss is handled by the feed forward network based on feature embeddings. This method encodes physio-chemical properties that also capture the structural variation among the protein sequences. The feature vector is passed to a 1D convolutional neural network. The 1D CNN is used to identify the locally conserved patterns at different sites in the sequence.

2.3 Novel Database Creation

DeepARG+ aims to identify the novel antibiotic resistance genes from the samples provided by the user. Novel ARGs have very less sequence similarity with the genes in existing database. After the prediction pipeline of DeepARG+ is executed, we use DIAMOND to

find the percentage identity of the predicted sequences to the existing sequences. Finally, we apply the cut-off of 60% to the percentage identity of the obtained hits. The 60% identity cutoff is obtained empirically. The genes that have less identity than this probability cut-off are stored in a novel-database file. These sequences can be used for further analysis and validation in the laboratory.

2.4 Prediction Pipeline of DeepARG+

A trained model is deployed for the prediction of ARGs. Given a fasta file as an input the prediction pipeline performs following steps.

1. Test the validity of the input file
2. If the sequences in the file is DNA, check whether it's already gene sequence or not. If it is gene sequence, or a coding sequence, then translate it into protein sequence.
3. If it is simply a DNA string, use prodigal to predict ORFs on it
4. Generate features and predict the sequences.
5. Pass the predicted sequence through the pipeline discussed in [2.3](#) to get the required files for prediction.

2.5 Phylogenetic Analysis

An elongation factor Tu outgroup from Bacteroides was combined with 84 verified sequences from our validation dataset, 20 sequences identified by both DeepARG+ and the best-hit

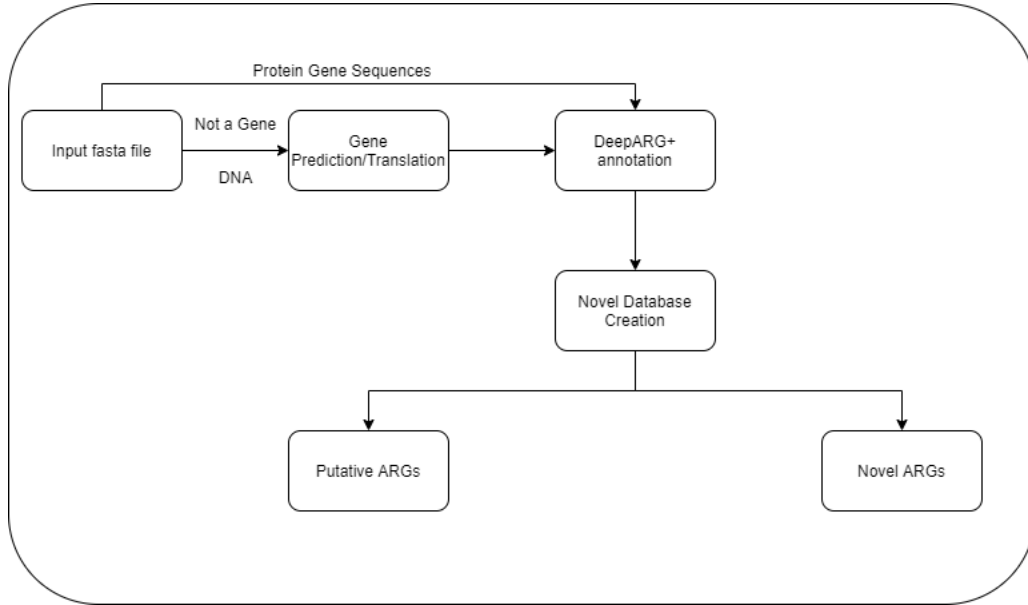


Figure 2.5: Prediction Pipeline of DeepARG+

approaches, were aligned using mafft [40] with parameters `mafft --localpair --maxiterate 1000`. Sequences were then trimmed using the heuristic setting in trimal (automated1) (Trimal) [24]. Finally, we used iqtree2 [52] [56] to select the best model (LG+F+R4) by the Bayesian Information Criterion using the following settings: `-alrt 1000 -B 1000`. software and default parameters. The tree was visualized with FigTree [1] software 2.6.

The models are built using tensorflow 1.3 [5] library and keras [32] API. These models were optimized by the accuracy, precision and recall metrics.

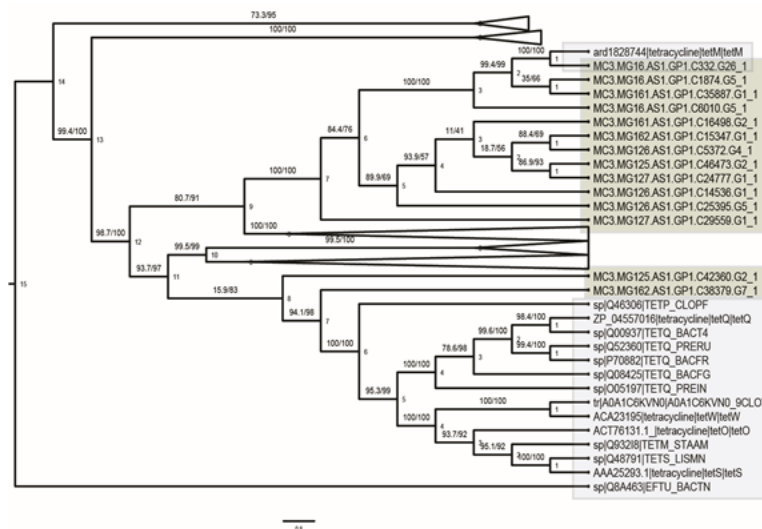


Figure 2.6: Maximum likelihood phylogeny of TetO-, TetP-, TetS-, TetM-, and TetQ-type tetracycline resistance proteins. Highlighted sequences represent known ARG protein sequences (green: TetQ/TetP/TetS; grey: TetM)

Chapter 3

Results

3.1 Evaluation on Test Data

We use the holdout method for the evaluation on test data, where data is split into independent training and testing datasets. The training and testing data consisted of 23,758 and 7,359 independent sequences belonging to 16 antibiotic classes and a negative class. We also use a standard cut-off for each class, to filter out non-confident predictions. our model gave an accuracy of 97.41%, precision of 97%, recall of 97% and F1-score of 97%. These metrics can be biased towards the majority class in an imbalance-multiclass classification problem as shown in figure B.1. Therefore, we also evaluated the model on metrics like weighted precision (94%), weighted recall (90%) and weighted F1-score (92%). The values of metrics indicate a good performance of the model across classes. From the confusion matrix (Figure 3.1) we can see that performance of the model is not good on the polymyxin class. Most of the polymyxin resistance genes are classified as genes resistant to the sulfonamide class initially with a low confidence. However, after applying the probability cut-off they are classified into negative class. This might be due to the alpha-helix rich structure of both polymyxin class and sulfonamide class resistance proteins (e.g., Sul1 PDB structure and polymyxin PDB structure).

Any machine learning model assumes that the training and testing data come from the same distribution. The comparison of other approaches on this test data would impute some bias

in our analysis. Therefore, the comparison of pipelines was done on the independent datasets discussed in the consequent sub sections.

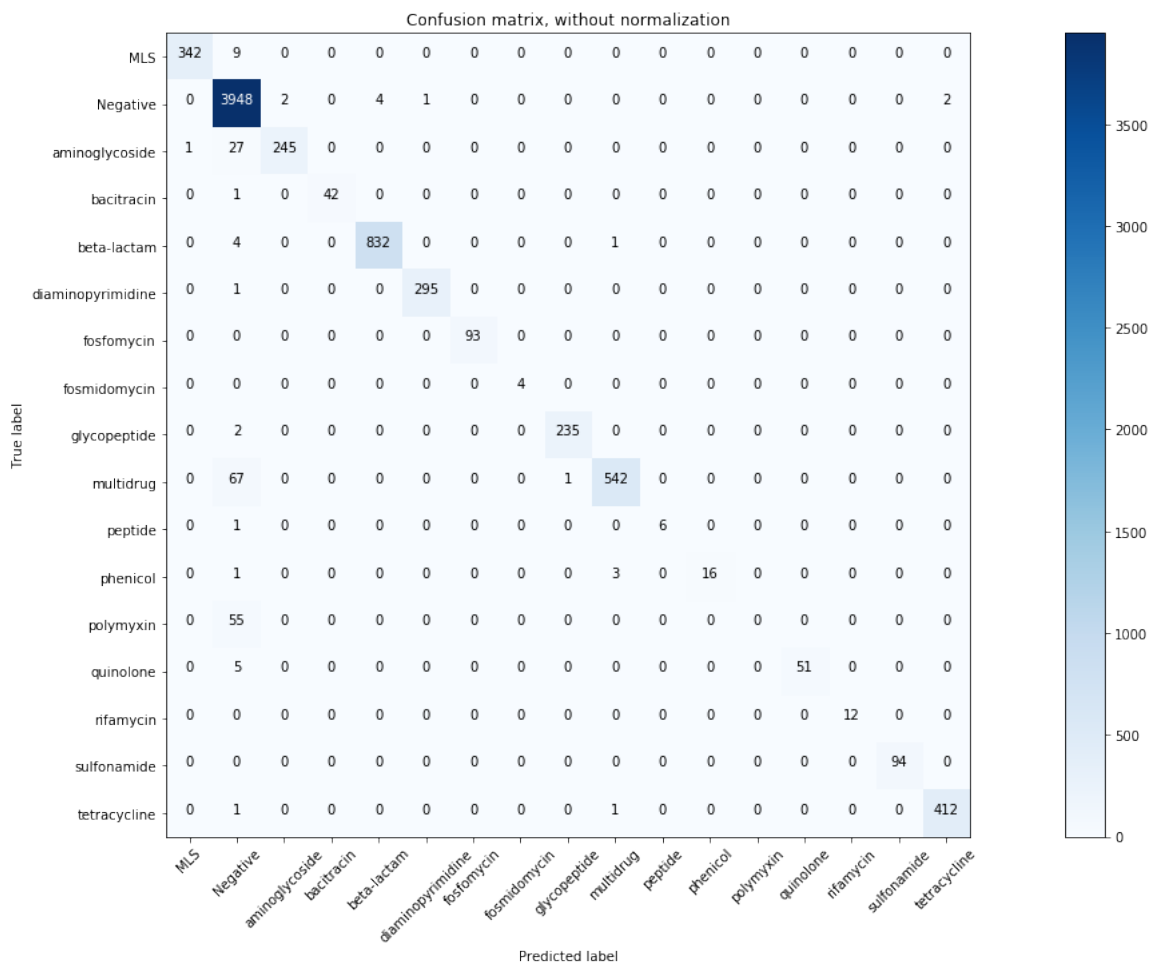


Figure 3.1: Confusion matrix on test data

3.2 Evaluation on Novel beta-lactamases

We evaluated our model against 77 metallo-beta lactamase genes obtained from the study by Berglund et al. [17]. Berglund *et al.* have experimentally validated these genes to confer resistance to carbapenem antibiotics in *Escherichia coli*. Out of the 77 Novel beta-lactamases,

DeepARG+ was able to predict 71 (92.2%) of the genes correctly while DeepARG-LS was able to predict 65 of these beta-lactamases correctly. It is important to compare with DeepARG-LS as these pipelines are not the class-specific pipelines like fARGene (used to annotate only beta-lactams). We use the best hit approach to find similarity of these beta-lactam resistant sequences with existing sequences in our training dataset. All sequences had less than 50% identity with the existing data. These results indicate that along with the samples in the same distribution (test data) we were also able to identify samples that are very different from our data, demonstrating the ability of DeepARG+ to identify novel ARGs.

3.3 Evaluation on Soil Dataset

DeepARG+ was compared against existing pipelines for validation on the functional metagenomic dataset. Forsberg *et al.* [31], provide the data of assembled metagenomic gene sequences obtained from soil. The samples are present in the Genbank [15] database with accession codes KJ691878–KJ696532. There are a total 4655 genome sequences in the dataset. Prodigal [38] with default parameters was used to obtain 12,711 open reading frames (ORFs) from the sequences. Forsberg *et al.* [31] in their work have functionally validated 2895 ARGs. The functionally resistant DNA fragments were sequenced and assembled using PARFuMs [31]. They predicted the ORFs using MetaGeneMark [77]. The final class annotation was done using either profile HMMs or the ORF was subcloned and confirmed to confer resistance when expressed in *E. coli*. This data consists of ARGs belonging to the classes on which DeepARG+ was not trained. Therefore, it is necessary to curate the data to find relevant ARGs in the data. We remove genes that have been annotated to efflux pumps and the classes not present in DeepARG from the dataset. After curation we found 1661

insert sequences belonging to 9 ARG families relevant for our analysis with the pipelines. Detailed results are discussed in section B.4. We get an accuracy of 94.94% on these 1661 sequences.

Different pipelines curate the data in their own way to assess the performance of the model. Therefore, the comparison of pipelines like PCM, DeepARG+, DeepARG, fARGene and Resfams was done on the classes common between them.

	DeepARG+	PCM	DeepARG-LS	fARGene
beta-lactam	1.00	0.90	1.00	1.00
aminoglycoside	0.84	0.87	1.00	NA
glycopeptide	1.00	NA	1.00	NA
tetracycline	1.00	1.00	0.74	NA
diaminopyrimidine	0.99	0.99	1.00	NA

Figure 3.2: Precision on different pipelines (NA part indicates absence of the class in data of pipeline)

	DeepARG+	PCM	DeepARG-LS	fARGene
beta-lactam	0.92	0.97	0.73	0.83
aminoglycoside	1.00	0.91	0.14	NA
glycopeptide	0.94	NA	0.47	NA
tetracycline	1.00	1.00	0.61	NA
diaminopyrimidin	0.99	0.99	0.78	NA

Figure 3.3: Recall on different pipelines (NA part indicates absence of the class in data of pipeline)

From the results in figures 3.2, 3.3, we can see that DeepARG+ has a good recall on the pipelines stringent for false positives. It is encouraging to see that even though fARGene [18] is a pipeline for classification of beta-lactams, we identify more beta-lactams than fARGene, without compromising on false positives. Precision is also comparable with all other pipelines, which proves the superiority of our model.

3.4 Evaluation on Human Gut Data

DeepARG+ was also used to find novel ARG sequences in the 3.9M MetaHIT gene catalogue [47]. These genes are built from the metagenomic sequencing of the faeces of 396 subjects from Denmark and Spain. The nucleotide sequences were converted into protein sequences. Prodigal[38] was used to obtain around 3.9M ORFs from the data. We also compared our methods against the homology, HMM and best-hit based pipelines. The number of predictions by each pipeline according to the class is given in the figure 3.4. In order to

ARGs predicted

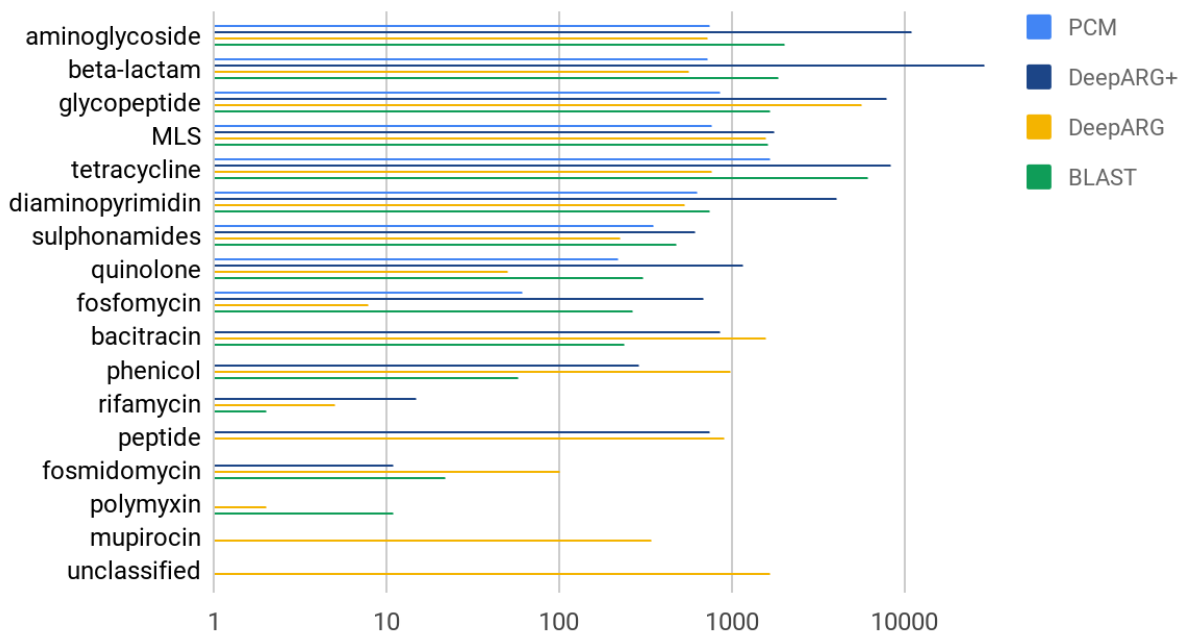


Figure 3.4: Number of sequences predicted by different pipelines on the gutdata)

minimize the number of false positives predicted by the best-hit approach, we curated the results of the BLAST pipeline to filter out the genes that have percentage identity less than 60% and e-value greater than $10e-5$. DeepARG+ predictions were also analyzed after applying a class specific threshold as discussed above. We can observe that PCM (a homology

based method) is able to predict only 6095 genes as ARGs from the dataset, while the HMM based approach captures a lot more (157,729) gene sequences from the dataset. It is highly possible that one of the pipelines is very conservative and has a lot of false negatives while the other approach produces a lot of false positives. It is known that the best hit approach identifies a lot of false negatives but less false positives [10]. Therefore, we can assume that the genes predicted by the best hit approach are correct positives, if appropriate cut-off is applied. As the results on this data is not functionally validated it is important to find the sequences that are common between the pipelines. We can see from figure 3.5

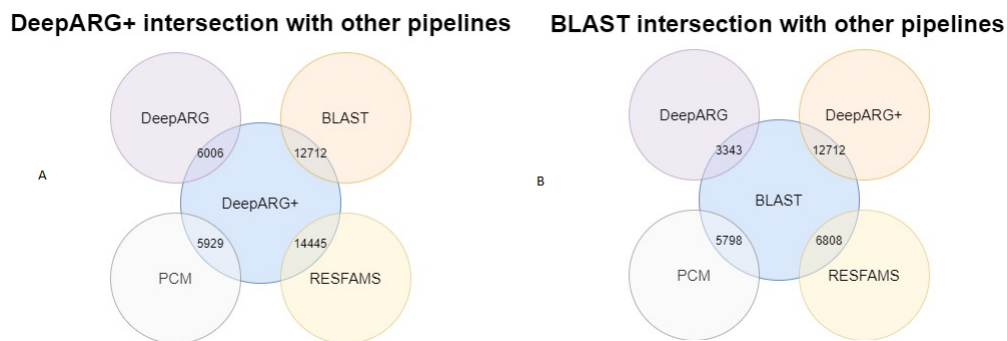


Figure 3.5: Intersection of DeepARG+ and BLAST with different pipelines

that DeepARG+ has the maximum intersection with the best-hit approach, indicating its correctness in identifying the positives. Additionally, from the figure 3.5 we can see that DeepARG+ is able to capture most of the sequences predicted by all pipelines. This proves that we are able to capture and combine the advantages of different approaches for ARG prediction.

3.4.1 PCM based predictions

PCM [67] functionally validated 71 genes having different percentage identity with the existing genes in the dataset. The genes were categorized according to the percentage identity with the known sequences. Categories were sequences having

- >80 %
- 30-80 %
- <30%

identity with known genes. We perform the analysis of different pipelines on this functionally validated dataset. The accuracy, precision, recall and F1 score of each of pipeline on this data is shown in figure 3.6.

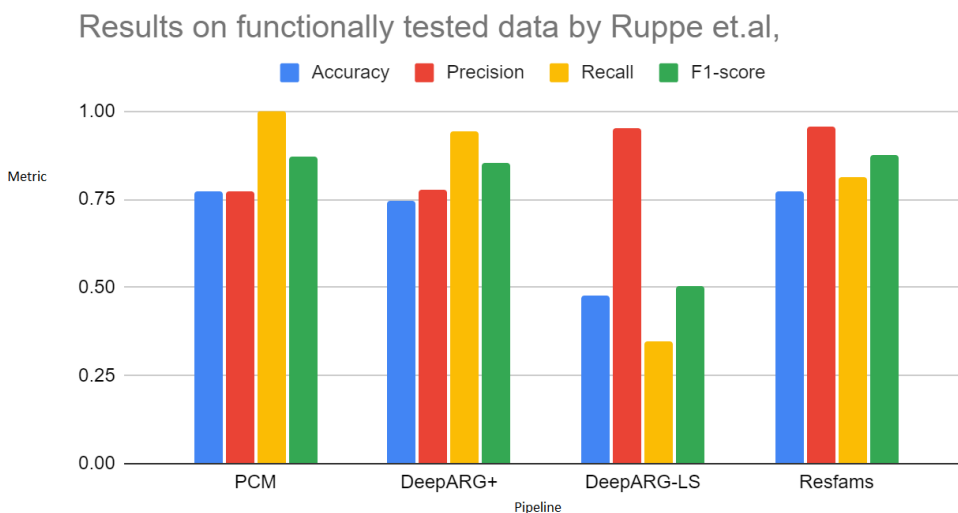


Figure 3.6: Metrics on functionally tested data by PCM

From figure 3.6, we can see that DeepARG-LS (a sequence similarity based deep learning approach) with default parameters is not able to capture a lot of the true positive genes in the dataset. It is not able to classify most genes having identity less than 30 percent with the sequences in the database (Figure B.9). On the contrary, DeepARG+ is able to identify these ‘novel’ gene sequences obtained and shows performance similar to PCM. We cannot completely disregard the performance of DeepARG-LS, since most of the misclassified ARGs belong to the sulfonamide class [10]. DeepARG mentions its lack of good performance on

sulfonamide due to lack of enough sequences in their training dataset.

Additionally, Ruppe, *et al.* [67] also provide a list of 16 ARG sequences that have greater than 40% identity with the reference gene sequence. PCM failed to capture these sequences as ARGs. DeepARG+ identified 12 out of these 16 sequences correctly as ARG classes, while DeepARG-LS with default parameters identified 11 of these ARGs. Thus, considering both the situations discussed above DeepARG+ was able to predict well where homology modelling fails and has an advantage over PCM, retaining the advantages of DeepARG (i.e. sequence similarity). This can be attributed to the use of physio-chemical properties, which identify the local structures of the protein sequences. This can be helpful in identifying the intended novel sequences.

3.5 Analysis of Nanopore Sequences in Influent and Activated Sludge data

Influent and activated sludge samples were collected from five waste water treatment plants (WWTPs) located in India (IND), Hong Kong (HKG), Switzerland (CHE), United States of America (USA), and Sweden (SWE). DNA was extracted with a FastDNA SPIN soil kit (MP Biomedicals, Solon OH), purified with Zymo DNA clean kit (Zymo Research, Irvine CA), prepped with a 1D native barcoding genomic DNA kit (SQK-LSK108, EXP-NBD103, Oxford Nanopore Technologies), and sequenced with a flow cell (R9.0 or 9.4) in a MinION. Sequences were base-called using Albacore (v2.3.1). More details of data acquisition can be found in previous publications [46] [22].

Figure shows the results on the influential data in different countries. We can see that

DeepARG+ captures less number genes than the best-hit approach with no identity cut-off and an e-value of 0.001. Surprisingly, the HMM based approach, gave very few hits on the nanopore assembled sequences contrary to its large number of predictions on the human gut data. Similar pattern is observed on Activated Sludge data as shown in figures 3.7 and 3.8.

Percentage ARG classified from Sludge

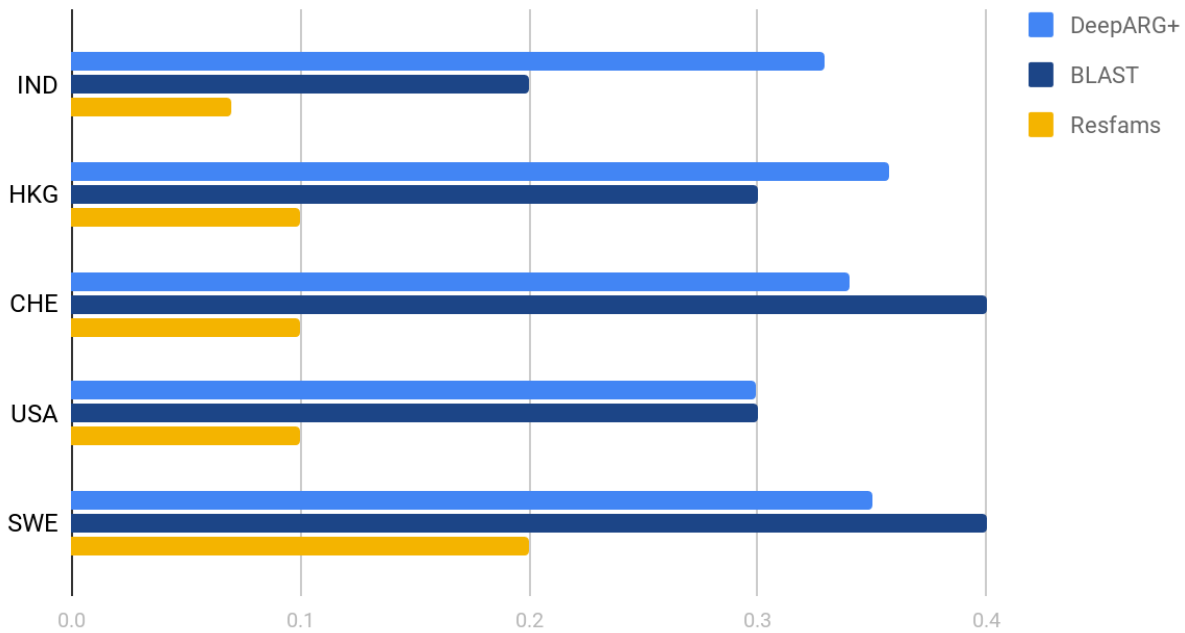


Figure 3.7: Percentage of ARGs predicted by DeepARG+ in total genes present in sludge samples (grouped by region)

We also perform inter-region analysis of ARGs in activated sludge and influents. Across all the pipelines, we can see the similar trend that the percentage of ARGs in activated sludge is less than the percentage of ARGs in the influent. This analysis can be seen in figure 3.9.

Percentage ARG classified in influent

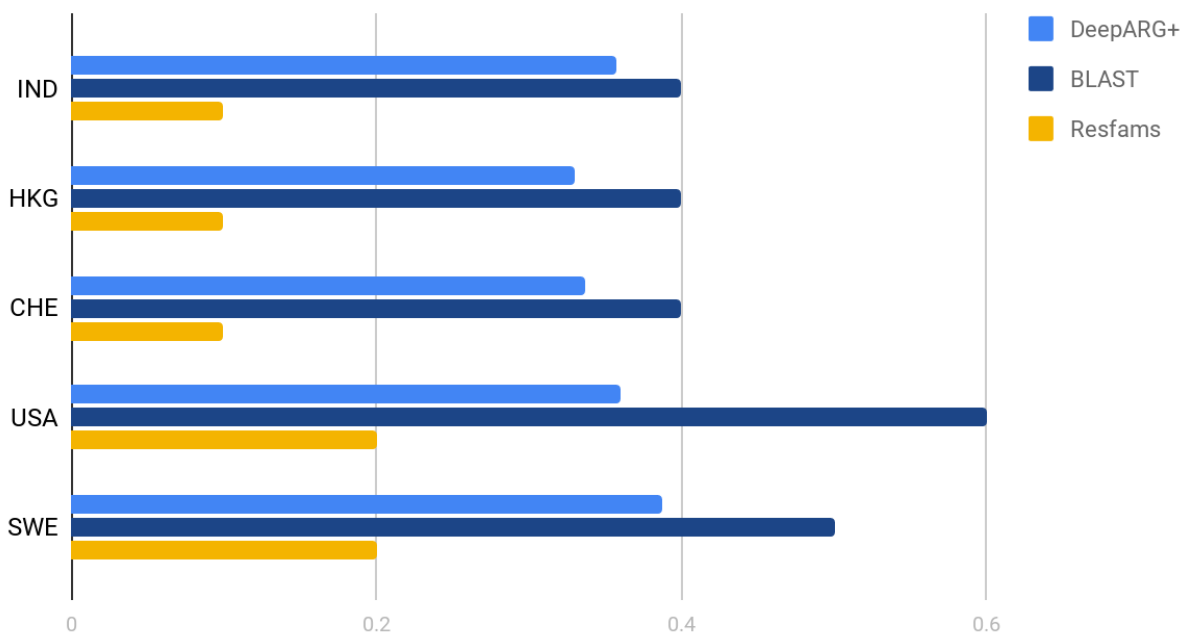


Figure 3.8: Percentage of ARGs predicted by DeepARG+ in total genes present in influent samples (grouped by region)

3.6 Performance Analysis of DeepARG+

Analysis of large scale data requires good accuracy as well as the speed with which the predictions can be obtained, we compare the performance of DeepARG-LS and DeepARG+ on the time in which it can classify the data of different sizes. We can see the improved performance of DeepARG+ over DeepARG-LS on large scale data. The increased performance can be seen in table 3.1

Table 3.1: Time Analysis of DeepARG+

Number of Sequences	DeepARG	DeepARG+
12711	4m 20 sec	1m 30 sec
400000	11m 34 sec	11m 25 sec
3.9M	90m 19 sec	56m 20 sec

Percentage of ARGs classified in Sludge and influent according to countries

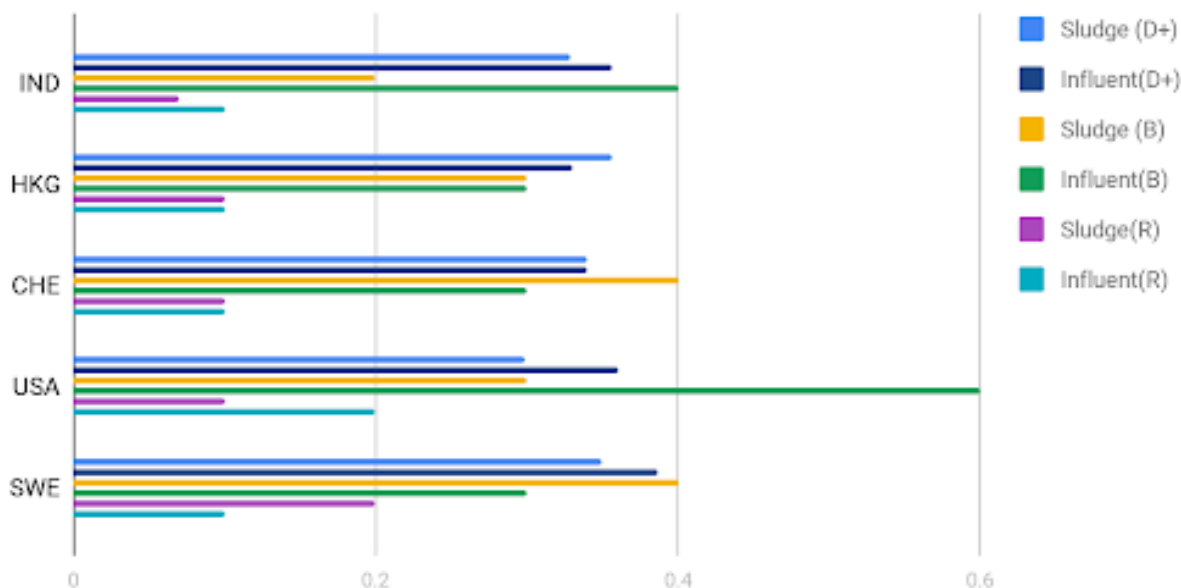


Figure 3.9: Percentage of ARGs predicted by all the pipelines in total genes present in influent and sludge samples (grouped by region)

3.7 Phylogenetic Analysis

To provide an example verification, we inferred a phylogeny of TetO-, TetS-, TetP-, TetQ-, and TetM-type tetracycline resistance genes in order to detect potential evolutionary relatedness between the novel sequences with well characterized ARGs. This revealed several putative novel tetracycline resistance proteins that were monophyletic with TetP and TetO family proteins.

Thus, in order to validate the pipeline we compared the predictions of our model on 4 different datasets. These datasets contain samples from soil, human gut, activated sludge and influents. The evaluation was done against the pipelines like fARGene, Resfams, DeepARG, PCM on these datasets. The results were evaluated on metrics like accuracy, precision, recall

and F1-score.

Chapter 4

Discussion

A large number of unknown ARGs are present in various environments such as waste-water, soil and human gut. DeepARG+ provides a comprehensive pipeline for finding such novel full-length gene sequences present in these environments. We validated our results on the known test data as well as the beta-lactams that have very low percentage identity with the ARGs seen by our model. The model shows high accuracy on these labelled datasets. On the test data our model does not perform well on the polymyxin class. Most of the polymyxin class ARGs in the test data are classified into sulfonamide class. We also compared our results with the existing pipelines such as fARGene, Resfams, DeepARG and a homology based method PCM. On the data obtained from soil samples we can see an improved performance of DeepARG+ over class-specific pipelines such as fARGene. DeepARG+ gives a high precision and recall as compared to most of the pipelines. From the functionally validated results on the human gut samples, we can see a comparable performance of our model to homology based methods. Additionally, the DeepARG+ model captures advantages of sequence similarity as well as homology based methods. We can also see that DeepARG+ is able to identify a higher number of ARGs as compared to other pipelines from the nanopore data while showing similar trends. It shows the ability of DeepARG+ to capture relevant novel sequences from any dataset. DeepARG+ also models negative sequences in its training dataset, most of the existing pipelines ignore negative sequences. Modelling negative sequences gives a clear understanding of the decision boundary between positive and negative classes, irrespective

of the sequence alignment. This helps capture the correct positives and negatives from any dataset. The performance of DeepARG+ is dependent on the training dataset used to develop the model. It is also dependent on the physio-chemical property used for encoding the sequences. The performance of different models can be seen in Supplementary Information in the appendix B. Our phylogenetic analysis supports the preliminary identification of many potential novel tetracycline resistance proteins 2.6. This revealed a monophyletic relationship between a confirmed resistance protein TetM.

Chapter 5

Conclusion

Annotation and validation of novel ARGs is a challenging task. Most developed pipelines use the sequence similarity based approach for ARG annotation. Protein structural homology modeling has been developed as an alternative. However, this approach is limited to study only those ARGs that have structural references. In this thesis, the physio-chemical properties are used as a proxy to the structural information of the proteins. Results have shown that the use of similarity based word-embeddings and physio-chemical properties based vectors together yields an improved accuracy over existing models.

Chapter 6

Future Work

Validating predicted ARGs is a challenging task. For future work, we plan to incorporate additional validation procedures into the ARG prediction pipeline. Specifically, we envision a system that incorporates both literature mining and user or expert feedback into the prediction workflow. Figure 6.1 shows such a prototype of the system, where users will play an important role in accepting or rejecting the prediction of the models based on a variety of information provided by the system (e.g., probability scores that the gene belonging to a certain ARG class and relevant literature on the genes). The feedback provided by the users will in turn be fed back into deep neural networks to further improve the prediction models.

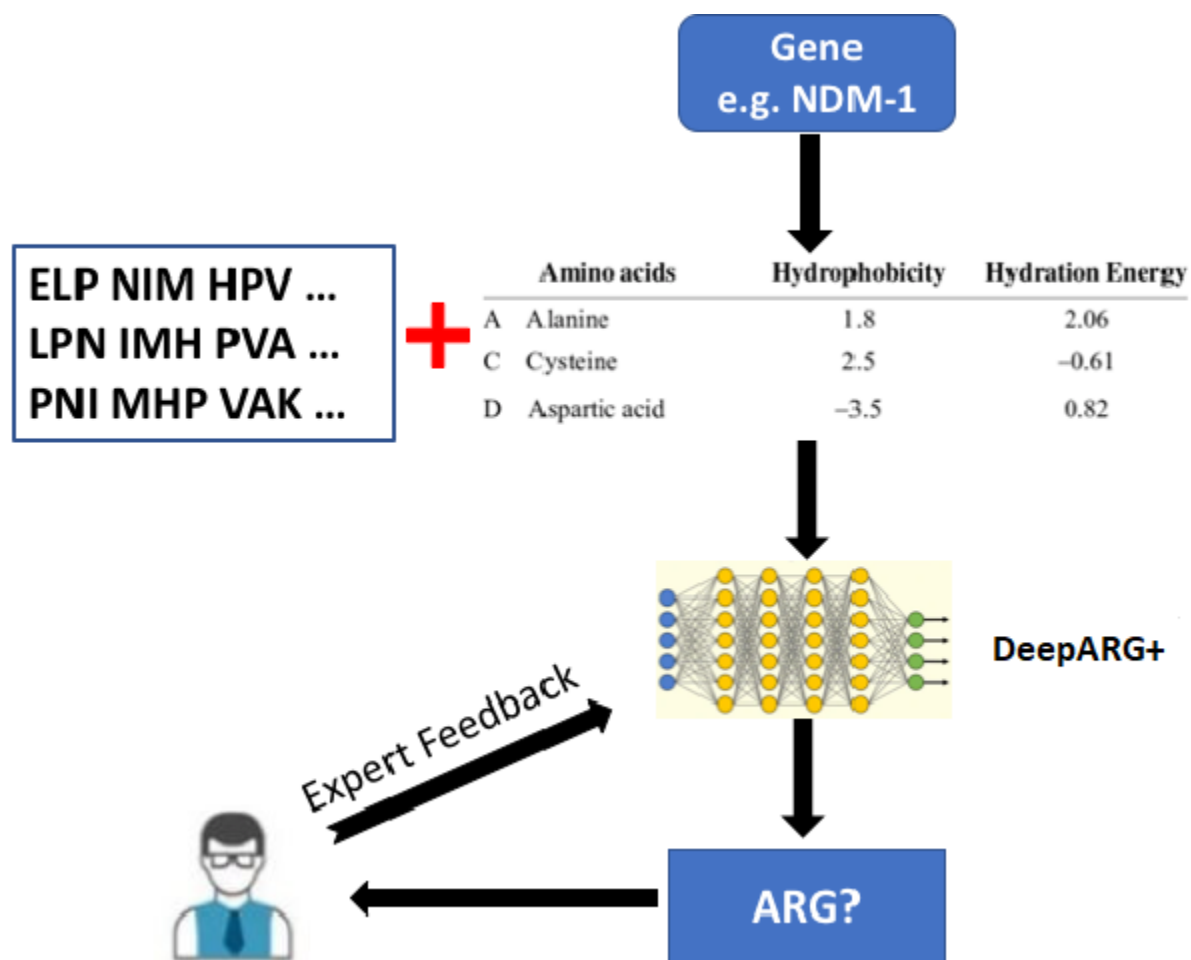


Figure 6.1: Planned system based on expert based validation

Bibliography

- [1] Figtree. URL <http://tree.bio.ed.ac.uk/software/figtree/>. [Accessed: 5th March 2021].

- [2] Scientists make significant breakthrough on superbug-killing antibiotic teixobactin, Nov 2017. URL <https://www.sciencedaily.com/releases/2017/11/171106112241.htm>. [Accessed: 10th April 2021].

- [3] The development of antimicrobial resistance antibiotics, Oct 2018. URL <https://www.grepmed.com/images/3731/resistance-policy-history-antimicrobial-timeline>. [Accessed: 10th April 2021].

- [4] Antibiotic / antimicrobial resistance, Jul 2020. URL <https://www.cdc.gov/drugresistance/index.html>. [Accessed: 10th April 2021].

- [5] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16)*, pages 265–283, 2016.

- [6] Boluwatife A Adewale. Will long-read sequencing technologies replace short-read sequencing technologies in the next 10 years? *African Journal of Laboratory Medicine*, 9 (1), 2020.

- [7] Shanika L Amarasinghe, Shian Su, Xueyi Dong, Luke Zappia, Matthew E Ritchie, and

- Quentin Gouil. Opportunities and challenges in long-read sequencing data analysis. *Genome biology*, 21(1):1–16, 2020.
- [8] GA Arango-Argoty, LS Heath, A Pruden, PJ Vikesland, and L Zhang. Metamp: A fast word embedding based classifier to profile target gene databases in metagenomic samples. *bioRxiv*, page 569970, 2019.
- [9] GA Arango-Argoty, GKP Guron, Emily Garner, Maria V Riquelme, LS Heath, Amy Pruden, PJ Vikesland, and Liqing Zhang. Argminer: a web platform for the crowdsourcing-based curation of antibiotic resistance genes. *Bioinformatics*, 36(9):2966–2973, 2020.
- [10] Gustavo Arango-Argoty, Emily Garner, Amy Pruden, Lenwood S Heath, Peter Vikesland, and Liqing Zhang. Deeparg: a deep learning approach for predicting antibiotic resistance genes from metagenomic data. *Microbiome*, 6(1):1–15, 2018.
- [11] Ehsaneddin Asgari and Mohammad RK Mofrad. Continuous distributed representation of biological sequences for deep proteomics and genomics. *PloS one*, 10(11):e0141287, 2015.
- [12] Fernando Baquero, José-Luis Martínez, and Rafael Cantón. Antibiotics and antibiotic resistance in water environments. *Current opinion in biotechnology*, 19(3):260–265, 2008.
- [13] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. A neural probabilistic language model. *The journal of machine learning research*, 3:1137–1155, 2003.
- [14] Johan Bengtsson-Palme, Erik Kristiansson, and DG Joakim Larsson. Environmental

- factors influencing the development and spread of antibiotic resistance. *FEMS microbiology reviews*, 42(1):fux053, 2018.
- [15] Dennis A Benson, Mark Cavanaugh, Karen Clark, Ilene Karsch-Mizrachi, David J Lipman, James Ostell, and Eric W Sayers. Genbank. *Nucleic acids research*, 41(D1):D36–D42, 2012.
- [16] Raoul Benveniste and Julian Davies. Aminoglycoside antibiotic-inactivating enzymes in actinomycetes similar to those present in clinical isolates of antibiotic-resistant bacteria. *Proceedings of the National Academy of Sciences*, 70(8):2276–2280, 1973.
- [17] Fanny Berglund, Nachiket P Marathe, Tobias Österlund, Johan Bengtsson-Palme, Stathis Kotsakis, Carl-Fredrik Flach, DG Joakim Larsson, and Erik Kristiansson. Identification of 76 novel b1 metallo- β -lactamases through large-scale screening of genomic and metagenomic data. *Microbiome*, 5(1):1–13, 2017.
- [18] Fanny Berglund, Tobias Österlund, Fredrik Boulund, Nachiket P Marathe, DG Joakim Larsson, and Erik Kristiansson. Identification and reconstruction of novel antibiotic resistance genes from metagenomes. *Microbiome*, 7(1):52, 2019.
- [19] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.
- [20] Phelim Bradley, N Claire Gordon, Timothy M Walker, Laura Dunn, Simon Heys, Bill Huang, Sarah Earle, Louise J Pankhurst, Luke Anson, Mariateresa De Cesare, et al. Rapid antibiotic-resistance predictions from genome sequence data for staphylococcus aureus and mycobacterium tuberculosis. *Nature communications*, 6(1):1–15, 2015.
- [21] Daniel Branton, David W Deamer, Andre Marziali, Hagan Bayley, Steven A Benner,

- Thomas Butler, Massimiliano Di Ventra, Slaven Garaj, Andrew Hibbs, Xiaohua Huang, et al. The potential and challenges of nanopore sequencing. *Nanoscience and technology: A collection of reviews from Nature Journals*, pages 261–268, 2010.
- [22] Connor L Brown, Ishi M Keenum, Dongjuan Dai, Liqing Zhang, Peter J Vikesland, and Amy Pruden. Critical evaluation of short, long, and hybrid assembly for contextual analysis of antibiotic resistance genes in complex environmental metagenomes. *Scientific reports*, 11(1):1–12, 2021.
- [23] Benjamin Buchfink, Chao Xie, and Daniel H Huson. Fast and sensitive protein alignment using diamond. *Nature methods*, 12(1):59–60, 2015.
- [24] Salvador Capella-Gutiérrez, José M Silla-Martínez, and Toni Gabaldón. trimal: a tool for automated alignment trimming in large-scale phylogenetic analyses. *Bioinformatics*, 25(15):1972–1973, 2009.
- [25] UniProt Consortium. Uniprot: a hub for protein information. *Nucleic acids research*, 43(D1):D204–D212, 2015.
- [26] UniProt Consortium. Uniprot: a worldwide hub of protein knowledge. *Nucleic acids research*, 47(D1):D506–D515, 2019.
- [27] Gautam Dantas, Morten OA Sommer, Rantimi D Oluwasegun, and George M Church. Bacteria subsisting on antibiotics. *Science*, 320(5872):100–103, 2008.
- [28] Vanessa M D’Costa, Katherine M McGrann, Donald W Hughes, and Gerard D Wright. Sampling the antibiotic resistome. *Science*, 311(5759):374–377, 2006.
- [29] M Demerec, EA Adelberg, AJ Clark, and Philip E Hartman. A proposal for a uniform nomenclature in bacterial genetics. *Genetics*, 54(1):61, 1966.

- [30] Robert C Edgar. Search and clustering orders of magnitude faster than blast. *Bioinformatics*, 26(19):2460–2461, 2010.
- [31] Kevin J Forsberg, Alejandro Reyes, Bin Wang, Elizabeth M Selleck, Morten OA Sommer, and Gautam Dantas. The shared antibiotic resistome of soil bacteria and human pathogens. *science*, 337(6098):1107–1111, 2012.
- [32] Aurélien Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O’Reilly Media, 2019.
- [33] Molly K Gibson, Kevin J Forsberg, and Gautam Dantas. Improved annotation of antibiotic resistance determinants reveals microbial resistomes cluster by ecology. *The ISME journal*, 9(1):207–216, 2015.
- [34] M Michael Gromiha. *Protein bioinformatics: from sequence to function*. academic press, 2010.
- [35] Ruth M Hall and Stefan Schwarz. Resistance gene naming and numbering: is it a new gene or not? *Journal of Antimicrobial Chemotherapy*, 71(3):569–571, 2016.
- [36] Julie C Dunning Hotopp. Horizontal gene transfer between bacteria and animals. *Trends in genetics*, 27(4):157–163, 2011.
- [37] Matthew I Hutchings, Andrew W Truman, and Barrie Wilkinson. Antibiotics: past, present and future. *Current opinion in microbiology*, 51:72–80, 2019.
- [38] Doug Hyatt, Gwo-Liang Chen, Philip F LoCascio, Miriam L Land, Frank W Larimer, and Loren J Hauser. Prodigal: prokaryotic gene recognition and translation initiation site identification. *BMC bioinformatics*, 11(1):1–11, 2010.

- [39] Kristoffer Illergård, David H Ardell, and Arne Elofsson. Structure is three to ten times more conserved than sequence—a study of structural response in protein cores. *Proteins: Structure, Function, and Bioinformatics*, 77(3):499–508, 2009.
- [40] Kazutaka Katoh, John Rozewicki, and Kazunori D Yamada. Mafft online service: multiple sequence alignment, interactive sequence choice and visualization. *Briefings in bioinformatics*, 20(4):1160–1166, 2019.
- [41] W James Kent. Blat—the blast-like alignment tool. *Genome research*, 12(4):656–664, 2002.
- [42] Kortine Annina Kleinheinz, Katrine Grimstrup Joensen, and Mette Voldby Larsen. 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*, 4(2):e27943, 2014.
- [43] Ian Korf, Mark Yandell, and Joseph Bedell. *Blast*. ” O’Reilly Media, Inc.”, 2003.
- [44] Jack Kyte and Russell F Doolittle. A simple method for displaying the hydropathic character of a protein. *Journal of molecular biology*, 157(1):105–132, 1982.
- [45] Stuart B Levy, Laura M McMurry, Teresa M Barbosa, Vickers Burdett, Patrice Courvalin, Wolfgang Hillen, Marilyn C Roberts, Julian I Rood, and Diane E Taylor. Nomenclature for new tetracycline resistance determinants. *Antimicrobial agents and chemotherapy*, 43(6):1523–1524, 1999.
- [46] An-Dong Li, Jacob W Metch, Yulin Wang, Emily Garner, An Ni Zhang, Maria V Riquelme, Peter J Vikesland, Amy Pruden, and Tong Zhang. Effects of sample preservation and dna extraction on enumeration of antibiotic resistance genes in wastewater. *FEMS microbiology ecology*, 94(2):fix189, 2018.

- [47] Junhua Li, Huijue Jia, Xianghang Cai, Huanzi Zhong, Qiang Feng, Shinichi Sunagawa, Manimozhiyan Arumugam, Jens Roat Kultima, Edi Prifti, Trine Nielsen, et al. An integrated catalog of reference genes in the human gut microbiome. *Nature biotechnology*, 32(8):834–841, 2014.
- [48] Weizhong Li and Adam Godzik. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics*, 22(13):1658–1659, 2006.
- [49] Bo Liu and Mihai Pop. Ardb—antibiotic resistance genes database. *Nucleic acids research*, 37(suppl_1):D443–D447, 2009.
- [50] Andrew G McArthur, Nicholas Waglechner, Fazmin Nizam, Austin Yan, Marisa A Azad, Alison J Baylay, Kirandeep Bhullar, Marc J Canova, Gianfranco De Pascale, Linda Ejim, et al. The comprehensive antibiotic resistance database. *Antimicrobial agents and chemotherapy*, 57(7):3348–3357, 2013.
- [51] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
- [52] Bui Quang Minh, Heiko A Schmidt, Olga Chernomor, Dominik Schrempf, Michael D Woodhams, Arndt Von Haeseler, and Robert Lanfear. Iq-tree 2: New models and efficient methods for phylogenetic inference in the genomic era. *Molecular biology and evolution*, 37(5):1530–1534, 2020.
- [53] Jose M Munita and Cesar A Arias. Mechanisms of antibiotic resistance. *Virulence mechanisms of bacterial pathogens*, pages 481–511, 2016.
- [54] Richard E Nelson, Kelly M Hatfield, Hannah Wolford, Matthew H Samore, R Douglas Scott, Sujan C Reddy, Babatunde Olubajo, Prbasaj Paul, John A Jernigan, and James

- Baggs. National estimates of healthcare costs associated with multidrug-resistant bacterial infections among hospitalized patients in the united states. *Clinical Infectious Diseases*, 72(Supplement_1):S17–S26, 2021.
- [55] Patrick Ng. dna2vec: Consistent vector representations of variable-length k-mers. *arXiv preprint arXiv:1701.06279*, 2017.
- [56] Lam-Tung Nguyen, Heiko A Schmidt, Arndt Von Haeseler, and Bui Quang Minh. Iq-tree: a fast and effective stochastic algorithm for estimating maximum-likelihood phylogenies. *Molecular biology and evolution*, 32(1):268–274, 2015.
- [57] Patrice Nordmann, Laurent Poirel, Timothy R Walsh, and David M Livermore. The emerging ndm carbapenemases. *Trends in microbiology*, 19(12):588–595, 2011.
- [58] Howard Ochman, Jeffrey G Lawrence, and Eduardo A Groisman. Lateral gene transfer and the nature of bacterial innovation. *nature*, 405(6784):299–304, 2000.
- [59] Rosemarie O’Shea and Heinz E Moser. Physicochemical properties of antibacterial compounds: implications for drug discovery. *Journal of medicinal chemistry*, 51(10):2871–2878, 2008.
- [60] William R Pearson. Searching protein sequence libraries: comparison of the sensitivity and selectivity of the smith-waterman and fasta algorithms. *Genomics*, 11(3):635–650, 1991.
- [61] William R Pearson. Using the fasta program to search protein and dna sequence databases. In *Computer Analysis of Sequence Data*, pages 307–331. Springer, 1994.
- [62] William R Pearson. An introduction to sequence similarity (“homology”) searching. *Current protocols in bioinformatics*, 42(1):3–1, 2013.

- [63] William R Pearson. Finding protein and nucleotide similarities with fasta. *Current protocols in bioinformatics*, 53(1):3–9, 2016.
- [64] Anthony Rhoads and Kin Fai Au. Pacbio sequencing and its applications. *Genomics, proteomics & bioinformatics*, 13(5):278–289, 2015.
- [65] Will Rowe, Kate S Baker, David Verner-Jeffreys, Craig Baker-Austin, Jim J Ryan, Duncan Maskell, and Gareth Pearce. Search engine for antimicrobial resistance: a cloud compatible pipeline and web interface for rapidly detecting antimicrobial resistance genes directly from sequence data. *PloS one*, 10(7):e0133492, 2015.
- [66] Lakshmi Kalyani Ruddaraju, Sri Venkata Narayana Pammi, Girija sankar Guntuku, Veerabhadra Swamy Padavala, and Venkata Ramana Murthy Kolapalli. A review on anti-bacterials to combat resistance: From ancient era of plants and metals to present and future perspectives of green nano technological combinations. *Asian journal of pharmaceutical sciences*, 15(1):42–59, 2020.
- [67] Etienne Ruppé, Amine Ghoulane, Julien Tap, Nicolas Pons, Anne-Sophie Alvarez, Nicolas Maziers, Trinidad Cuesta, Sara Hernando-Amado, Irene Clares, Jose Luís Martínez, et al. Prediction of the intestinal resistome by a three-dimensional structure-based method. *Nature microbiology*, 4(1):112–123, 2019.
- [68] Temple F Smith, Michael S Waterman, et al. Identification of common molecular subsequences. *Journal of molecular biology*, 147(1):195–197, 1981.
- [69] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.
- [70] Raymond Vanhoof, Eleonora Hannecart-Pokorni, and Jean Content. Nomencla-

- ture of genes encoding aminoglycoside-modifying enzymes. *Antimicrobial agents and chemotherapy*, 42(2):483–483, 1998.
- [71] C Lee Ventola. The antibiotic resistance crisis: part 1: causes and threats. *Pharmacy and therapeutics*, 40(4):277, 2015.
- [72] Gerard D Wright. Antibiotic resistance in the environment: a link to the clinic? *Current opinion in microbiology*, 13(5):589–594, 2010.
- [73] Kevin K Yang, Zachary Wu, Claire N Bedbrook, and Frances H Arnold. Learned protein embeddings for machine learning. *Bioinformatics*, 34(15):2642–2648, 2018.
- [74] Yuzhen Ye, Jeong-Hyeon Choi, and Haixu Tang. Rapsearch: a fast protein similarity search tool for short reads. *BMC bioinformatics*, 12(1):1–10, 2011.
- [75] Xu-Xiang Zhang, Tong Zhang, and Herbert HP Fang. Antibiotic resistance genes in water environment. *Applied microbiology and biotechnology*, 82(3):397–414, 2009.
- [76] Yongan Zhao, Haixu Tang, and Yuzhen Ye. Rapsearch2: a fast and memory-efficient protein similarity search tool for next-generation sequencing data. *Bioinformatics*, 28(1):125–126, 2012.
- [77] Wenhan Zhu, Alexandre Lomsadze, and Mark Borodovsky. Ab initio gene identification in metagenomic sequences. *Nucleic acids research*, 38(12):e132–e132, 2010.

Appendices

Appendix A

DeepARG+ implementation

A.1 Setting up the requirements

1. Run `python setup.py build` command to build the files
2. Run `python setup.py install` command to install the requirements
3. Run `deepARG+ - - help` to check whether everything is correctly installed

A.2 Training the model

A.2.1 Training fastText model

1. Run `deepARG+ train_word_vectors` command to get the fastText model for generating word vectors
2. Run `deepARG+ fasta2vec` to create wordvectors for your training, validation and testing. **model.bin** will be the output
3. Use `deepARG+ train` command to train the model
 - (a) Input: Training and validation fasta files
 - (b) Output: Trained `model.hdf5` and `parameters.json` file

A.3 Prediction

1. Use *deepARG+ predict* command to get the predictions
 - (a) Input: fasta file of test sequences
 - (b) Output: 2 text files for novel ARGs and all predictions

Appendix B

Supplementary Results

B.1 Distribution of ARG classes

As discussed in section 2.1, we curated the dataset to have 36,792 sequences. The sequences were curated in a way that there are equal number of ARG and negative sequences in the data. The distribution of ARG classes in data is shown in figure B.1

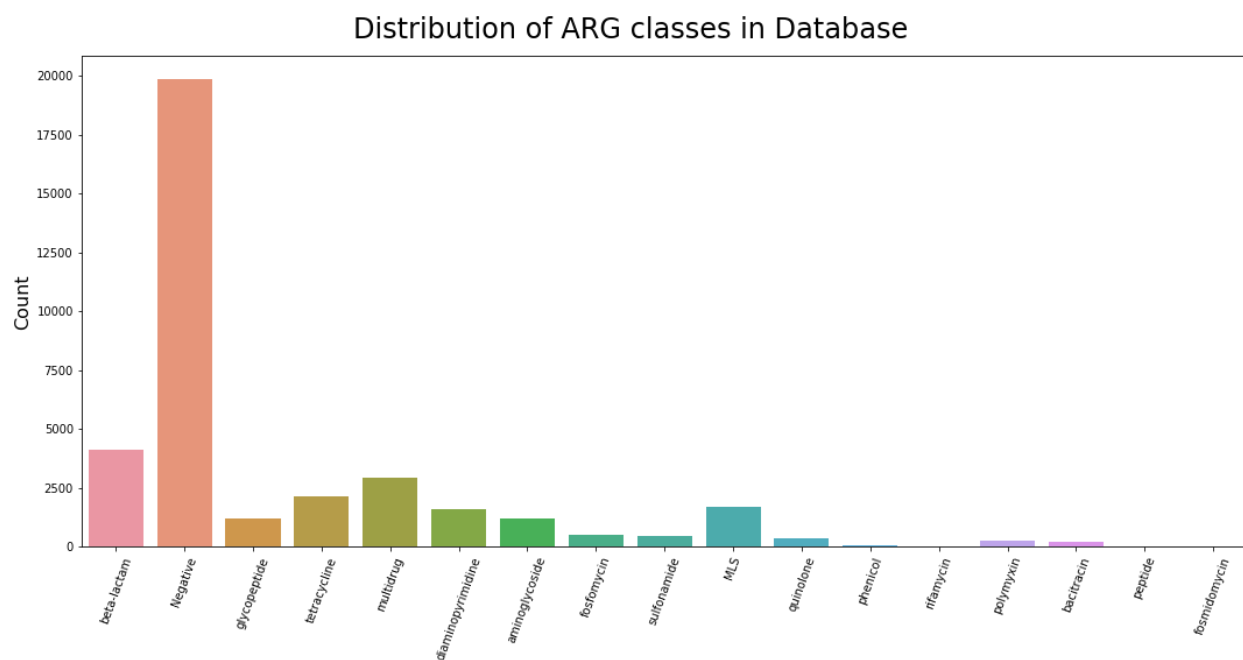


Figure B.1: Distribution of classes in database

B.2 Distribution of lengths of sequences

As discussed in section 2.2.3, we analyzed the lengths of all sequences in the database to decide the optimal length of the vector to be fed to a 1D CNN. The distribution lengths is shown in figure B.2. We can see that most sequences have length less than 1000 amino acids. Therefore, 1000 is chosen as the optimal length.

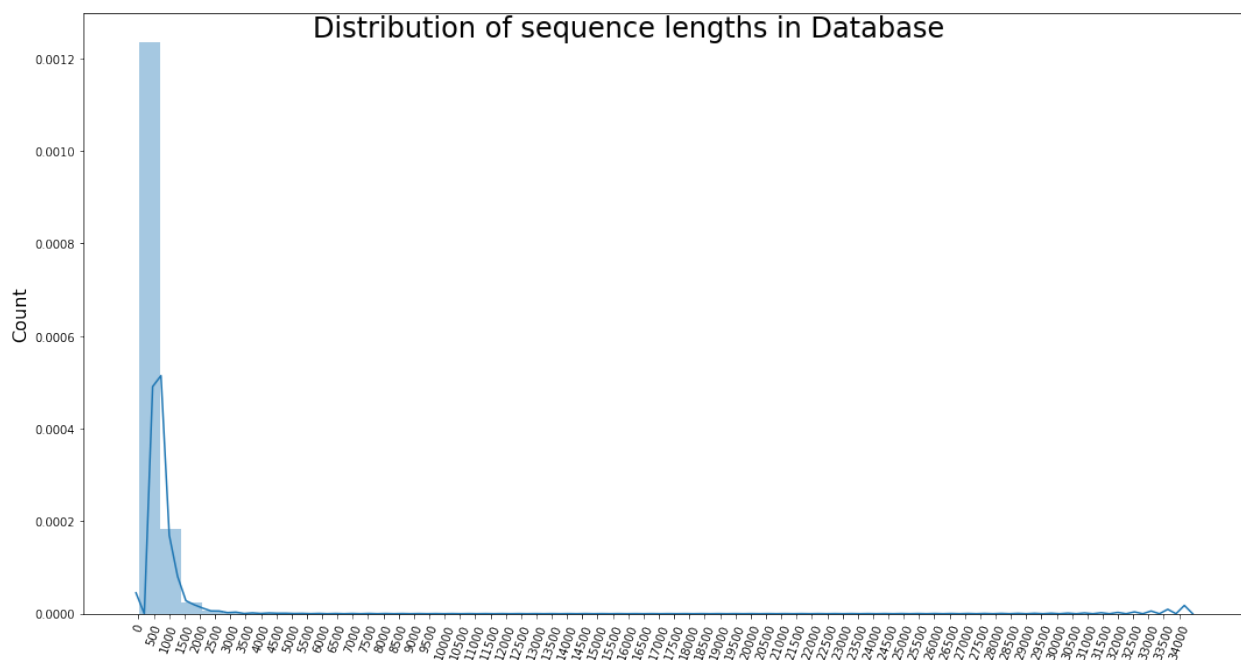


Figure B.2: Distribution of lengths of all sequences in database

B.3 Experiments

B.3.1 Independent feature vector based models

In order to assess the quality of feature vectors based on fastText word embeddings and different physio-chemical properties, we built models having input only of either of the em-

beddings. Convolutional neural networks were used for the models based on embeddings of physio-chemical properties. Models based on embeddings based on physio-chemical properties like hydrophobicity, molecular weight, isoelectric point and EIIP were built and the results were evaluated on the 77 novel beta lactams as discussed in section 3.2. Feed forward neural networks were used to test the fasttext embedding based model.

B.3.2 Ensemble based models

Each physio-chemical property has its own advantage and helps in predicting resistance classes. In order to utilize the advantage of all the physio-chemical properties, we built models based on different physio-chemical properties and ensembled them using the voting classifier. A hard voting classifier, polls each of these individual models and predicts the output class that is predicted by most the individual models. A soft voting classifier polls the individual models and predicts class based on probabilities predicted by the individual model.

B.3.3 Multichannel CNNs

A single convolutional neural network (CNN) can be used to build models that encode only one physio-chemical property. Different physio-chemical properties capture different aspects of the sequence and are helpful for classification. Therefore, to incorporate more features in data we need to either build more 1D CNNs (one for each CNN) or build a multichannel CNN. A multichannel CNN is a network in which each channel represents one encoding based on the physio-chemical property of the sequence. Such a technique helped us build simpler and flexible models in which features can be added easily for expansion.

Table B.1: Model Optimization

Model	Physio-chemical Property	Architecture	Prediction
Only fastText embeddings based model	None	4D	11
Only 1 physiochemical property	H	4C	64
Only 1 physiochemical property	MW	2C	70
Only 1 physiochemical property	PKa	2C	70
Ensemble Model- Hard voting	H, MW, PKa	NA	67
Ensemble Model- Soft voting	H, MW, PKa	NA	28
Multimodal Analysis	H	4C, 1D	51
Multimodal Analysis	H,EIIP	[4C*2], 1D	51
Mulichannel Analysis	h,EIIP	6C, 1D	68

B.3.4 Model optimization

As discussed in the preceding sections we built various models that incorporate physio-chemical properties. We used these models for resistant gene annotation and got interesting results. All the models discussed here were optimized using the Adam optimizer with 0.001 learning rate. In the below table ‘C’ stands for convolutional layers used for embeddings of physio-chemical properties while ‘D’ stands for dense layers used for fasttext embeddings. Only the results of the best models of each category are described in this section. The model optimization is described in table B.1. In table B.1 H stands for hydrophobicity, MW for molecular weight, PKa for isoelectric point and EIIP for Electron ion interaction potential. The prediction is from 77 novel beta-lactams discussed in chapter 3

B.3.5 Analysis of fastText Feature Vectors

fasText has some hyperparameters like word size (kmer size in our domain) and embedding size. These parameters need to be tuned to obtain the best set of feature vectors that can achieve maximum initial separation between class vectors. Figure shows the word embed-

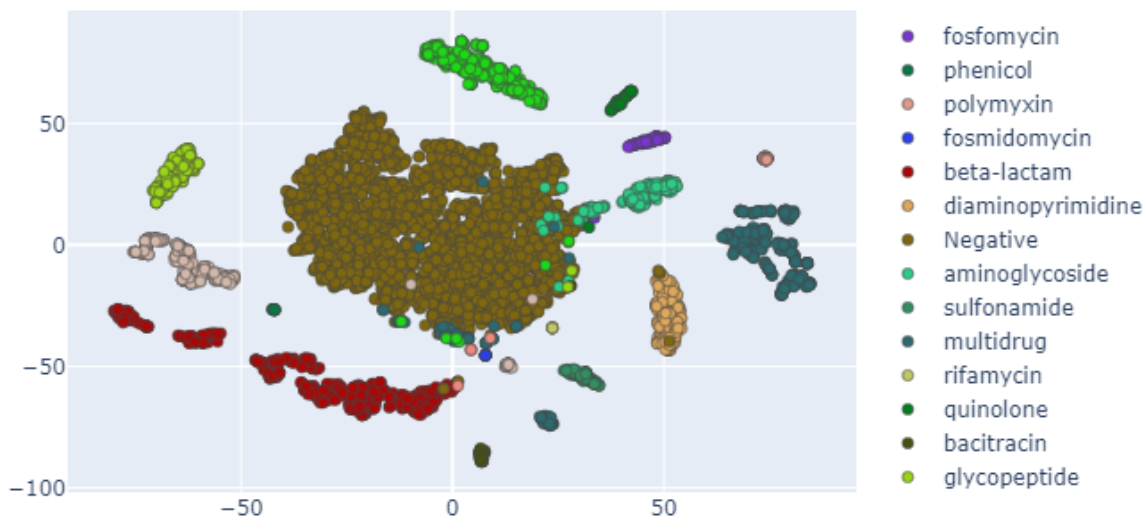


Figure B.3: fastText vectors on validation data for kmer size 20

dings generated from the test data. TSNE [69] algorithm is used to reduce dimensionality to facilitate the 2D visualization of embeddings (kmer size 20, 8 and 6). Figures B.3, B.4, B.5 shows the separation among the embeddings of sequences belonging to different classes. Therefore, we can say that our embeddings of kmer size 20 are powerful enough to get separation of the classes.

B.4 Results on soil data

As discussed in section 3.3, DeepARG+ predicted 1661 ARGs belonging to 9 classes from the data collected in the soil environment. In the section 3.3 we discussed evaluation on DeepARG+ against all the pipelines. The results were shown in figures 3.2 and 3.3. Here,

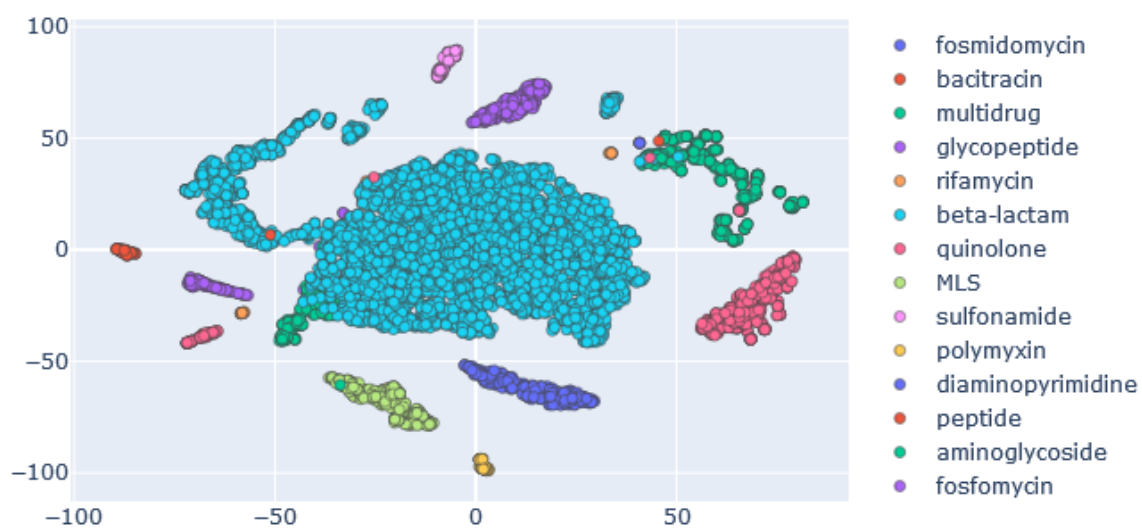


Figure B.4: fastText vectors on validation data for kmer size 8

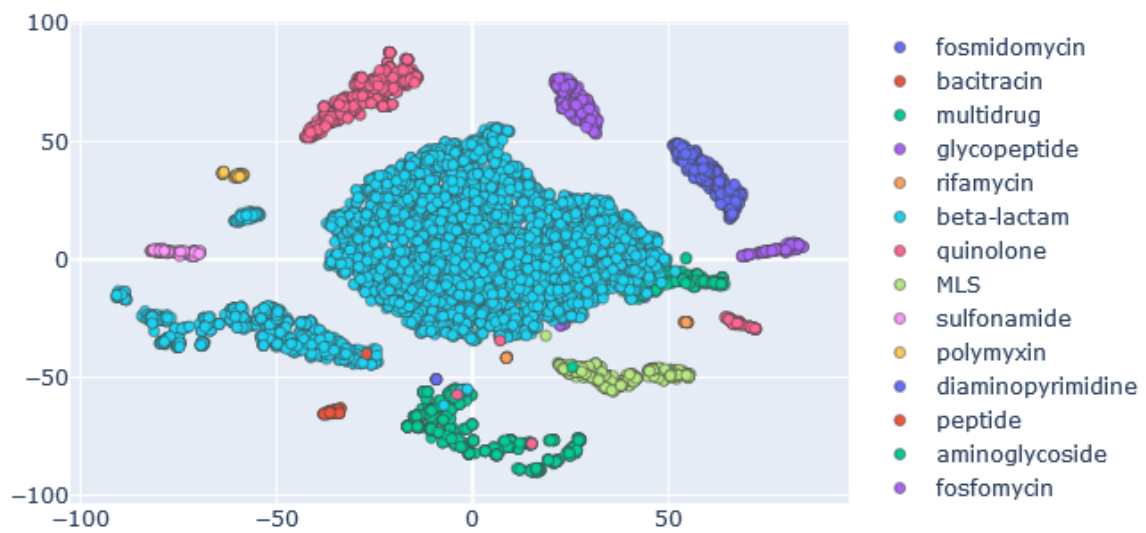


Figure B.5: fastText vectors on validation data for kmer size 6

we will discuss the detailed analysis of the soil data. The confusion matrix of the results is as shown in figure [B.6](#).

Table B.2: Metrics for soil data

Accuracy	94.94%
Precision	97%
Recall	95%
F1-score	96%

B.5 Evaluation on 71 validated ARGs by Ruppe *et al.*

[\[67\]](#)

As discussed in section [3.4.1](#), Ruppe *et al.* validated 71 genes from their predictions in the laboratory. These genes had varying percentage identity to the existing genes in the database. In this section, we discuss about percentage identity of the correctly classified genes by DeepARG+, DeepARG-LS [\[10\]](#) and PCM [\[67\]](#) from the 71 validated genes. DeepARG+ and PCM take the structure of protein in account, while DeepARG-LS is a sequence similarity based method. From figures [B.7](#) and [B.8](#) we can see that the structure based methods are able to classify genes irrespective of the identity with existing database. On the other hand figure [B.9](#) shows the inability of sequence similarity based method to predict genes having less than 30 percent identity with existing databases. Thus, the use of structural properties, help in improving the accuracy of prediction



Figure B.6: Confusion Matrix of DeepARG+ on Soil Data

% Identity of correctly classified ARGs with reference ARGs

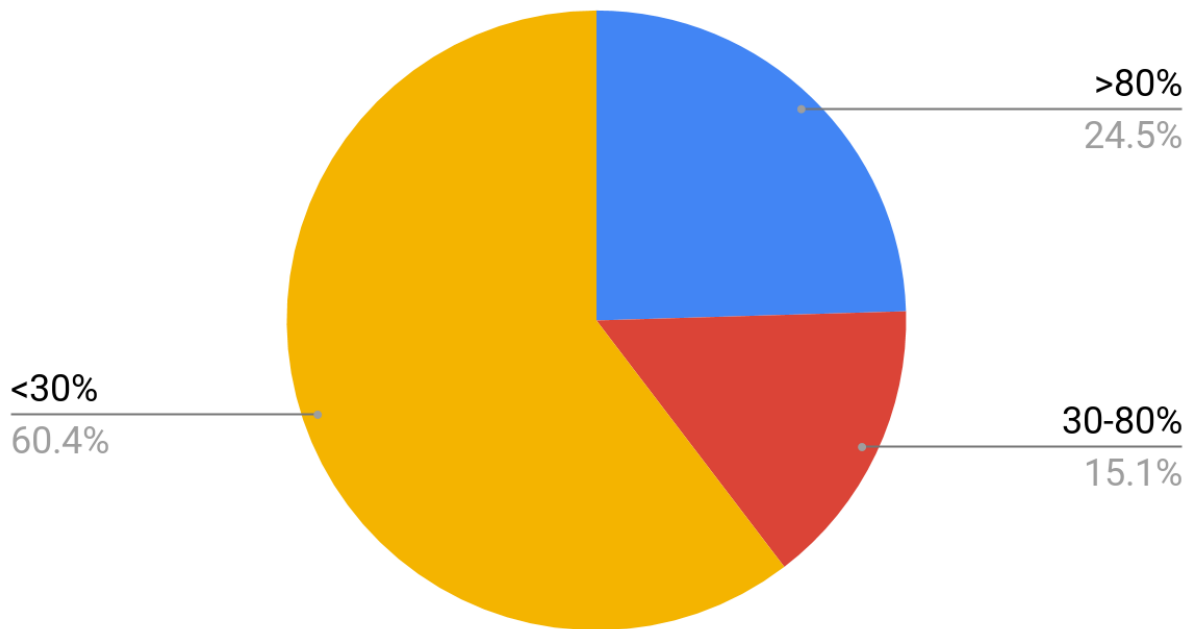


Figure B.7: Percentage Identity of correctly classified genes by DeepARG+

% Identity of correctly classified ARGs with reference ARGs

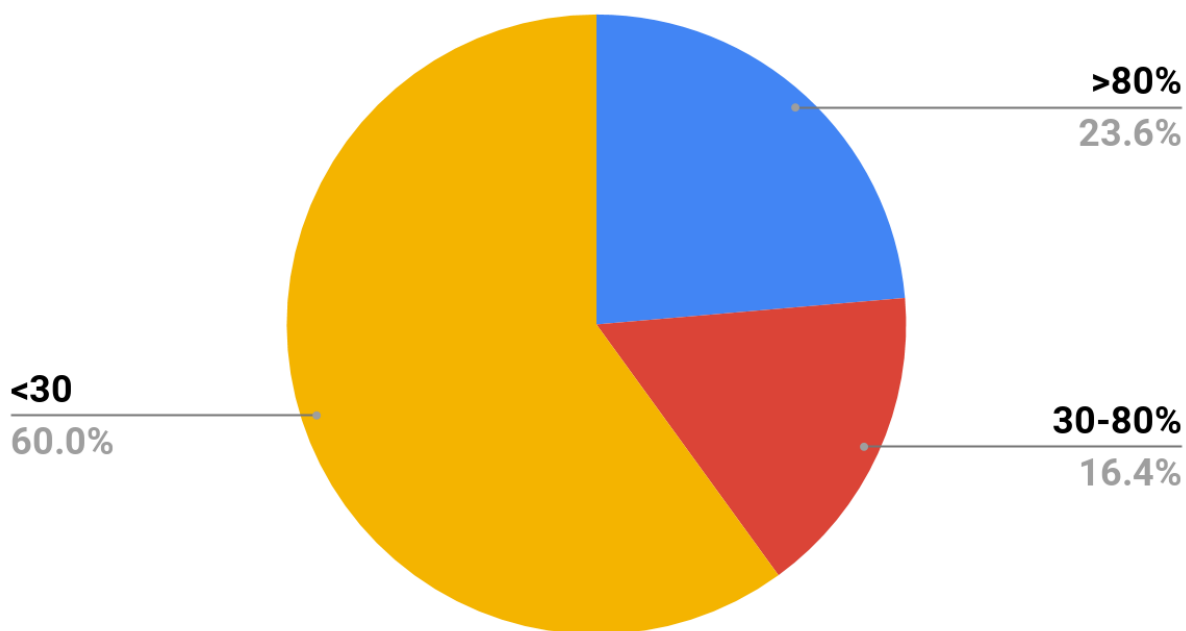


Figure B.8: Percentage Identity of correctly classified genes by PCM

% Identity of correctly classified ARGs with reference ARGs

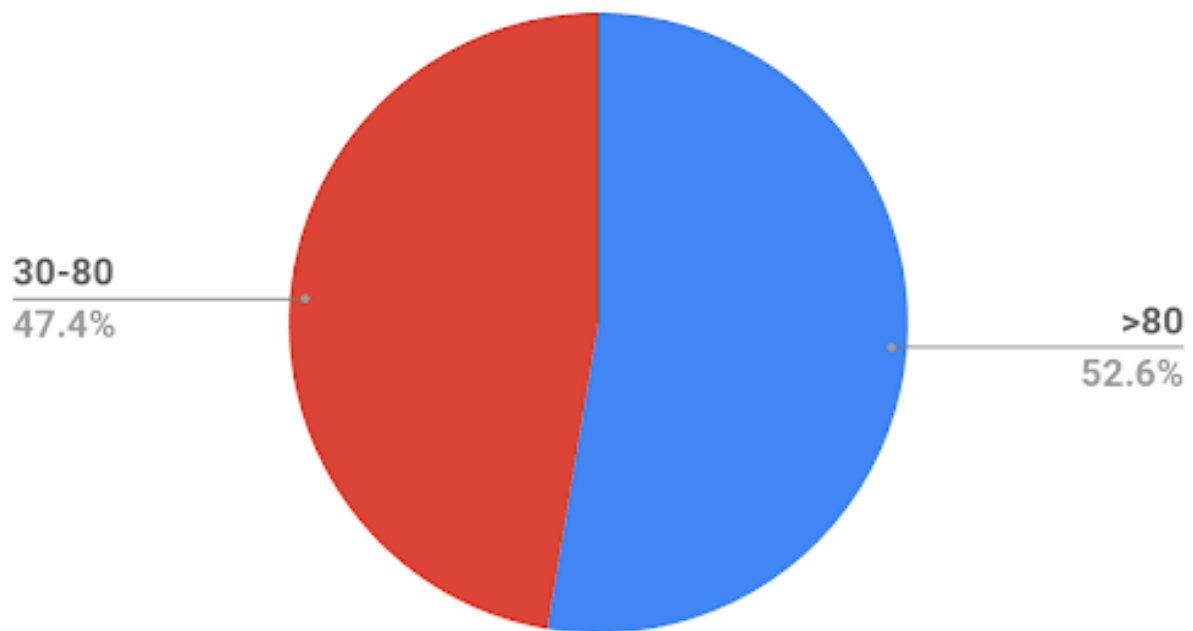


Figure B.9: Percentage Identity of correctly classified genes by DeepARG-LS