Antimicrobial resistance prediction using deep convolutional neural networks on whole genome sequence data

Andrew Zhang, Roxbury Latin School
Mentors: Dr. Gil Alterovitz and Dr. Insung Na, Harvard Medical School

Abstract

Antimicrobial resistance threatens the effectiveness of antibiotics against bacteria worldwide, causing hundreds of thousands of deaths annually. Unsure of which antibiotics will be effective against particular strains, clinicians are in the dark for prescriptions. Traditional culture-based testing takes at least two days for a detailed report, during which time the patient's condition could significantly worsen. Thus, there arises a need for faster identification of a bacteria's resistances. We propose a method to determine whether a bacterial strain is resistant to an antibiotic based on its whole genome sequence data using deep machine learning – deep convolutional neural networks (DCNN). DCNN can quickly and accurately classify data by learning features from large data sets, as shown in other areas of research such as image classification.

The DCNN model developed in this research is shown to achieve an average AMR prediction accuracy of 94.7%. Each prediction takes less than a second. The model is verified with Klebsiella pneumoniae resistance to tetracycline data and Acinetobacter baumannii resistance to

carbapenem data from the public database PATRIC. The DCNN model is further tested with clinically collected genomic data of 149 strains of Mycobacterium tuberculosis, and achieves a prediction accuracy of 93.1% for resistance to pyrazinamide (PZA). To find genes that harbor mutations of PZA resistance, we build a Support Vector Machine (SVM) model tailored for VCF format genomic data, which has revealed two novel genes, *embB* and *gyrA,* that harbor mutations associated with PZA resistance besides the well-known *pncA* gene.

Our DCNN and SVM Machine Learning framework, if used together with the real-time genome sequencing machines, which are now already available, could make rapid AMR predictions, allowing for critical time to ensure good patient outcomes and preventing outbreaks of deadly AMR infections. Furthermore, the developed framework identifies pertinent resistance genes, helping researchers understand the mechanisms behind resistance. Finally, this research demonstrates how deep machine learning techniques can produce high accuracy predictive models accelerating the diagnosis of AMR.

Background

Antibiotics, since their conception, have been an invaluable tool in fighting bacteria, providing doctors with an extremely potent weapon against many previously untreatable diseases. Due to the overwhelming effectiveness of antibiotics, their use has quickly spread over the last 70 years, and has brought down the risk of death by infectious disease considerably [1]. In addition to human use, antibiotics are given in vast amounts to farm animals [2][3]. This widespread use has also brought about a dangerous side-effect: the development of antimicrobial resistance (AMR) in many bacteria species [1][2][3][4]. AMR is one of the greatest current health crises. Each year, in the US alone, at least 2 million people are infected with bacteria with some resistance to

antibiotics, and 23,000 die from those infections [5][6]. Globally, the yearly deaths from AMR may total over 700,000 [1]. If the current trends continue, the number of deaths will rise to a devastating 10 million people a year by 2050 [1].

One particular problem in dealing with AMR is that when a patient is infected with a strain of a bacteria, doctors do not know which antibiotics it is resistant to. They then must respond by either prescribing broad-spectrum antibiotics, a wide variety of antibiotics, or waiting for culture-based lab results. However, a detailed report takes at least two days and can even take up to a month [7]. Broad-spectrum prescription is dangerous, because the overuse of antibiotics is what causes the development of AMR in the first place and wrong antibiotics can lead to adverse patient outcomes [8][9]. Waiting for culture-based results is hazardous towards patients' health, as their conditions may worsen during the time when the cultures are being tested, and infections have more chances to spread. Thus, there arises a need for faster identification of bacterial resistance.

Genomic data has long been used as a tool in medical research. In 2010, Bierut et al. examined single nucleotide polymorphisms (SNP) associated with vulnerability to alcohol dependence using genome-wide association strategy [10]. In 2012, Lavender et al. evaluated individual effects and complex interactions among 172 apoptotic SNPs in relation to prostate cancer risk [11].

Genome data is more readily available today with recent developments in whole genome sequencing (WGS) technology. The prices for sequencing are steadily going down, to $25 per million base pairs [7]. This advancement makes it possible for WGS of bacterial strains to be collected routinely in individual clinics by doctors. If we can quickly associate antimicrobial resistance with genome-wide WGS data, it will not only save valuable time for doctors to prescribe the right antibiotics to help save patient lives, but also prevent the further outbreak of life-threatening bacteria.

One highly successful method of using big data for classification is Deep Convolutional Neural Networks (DCNN) [12]. For example, DCNNs have enabled massive strides in image classification software in the early 2000s, a period known as the deep learning renaissance [13]. In the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012, Krizhevsky et al. trained a DCNN to classify 1.2 million high-resolution images into 1000 categories with record breaking accuracy [14]. DCNN have since become dominant for all sorts of classifications with powerful GPU, large available datasets and better algorithms.

Advancements in machine learning (ML), including its success in image classification and more recently in chemical design [15], have attracted the attention of biological researchers. As a case study of using ML to perform analysis on AMR using publicly available AMR datasets and ML tools, Santerre et al. [16] demonstrated the benefits and power of ML in approaching the practical problem of genotype to phenotype classification. The authors use the large-scale biological data aggregation platform Pathosystems Resource Integration Center (PATRIC) [17]

for AMR genotype and phenotype data in their ML case study. They use Random Forest (RF) from the off-the-shelf ML tool scikit-learn [23] as the classifier. Microbial genome data is divided into k-mers, which are features for RF. Their analysis was performed using Python on a 32 core machine with 1TB RAM. Their results are very impressive with accuracy as high as 92% in AMR phenotypes prediction, but there are hurdles to overcome to deploy the method in clinics. As microbial genomes typically have many millions of base pairs, the k-mers count can be huge, e.g, with k at 14, there are millions of k-mers as shown in the paper. The RF model would have millions of features. When using the off-the-shelf RF classifier, the authors run the model on a high-end computer with large memory and a high end CPU, and did not provide the time it takes to produce a prediction.

Another approach, by Pesesky et al., used the ML method of Logistic Regression (LR) to build an application called Genotype Based Antibiotic Susceptibility Prediction (GBASP) to predict antibiotics susceptibility using WGS [18]. However, a typical WGS covers several million base pairs, so genes were first annotated using an antibiotic resistance database to find known resistance genes which also reduces the input size for the LR. This subset of data, not the WGS, was used as input for the LR model, significantly reducing the number of inputs. The accuracy of prediction depended on the database being used for annotation and the bacterial strains being tested, and ranged from 57.7% to 94.9%. It should be noted that the elastic net regularization, which is also part of the scikit-learn [28] package, can handle larger input size, but it was not used in this literature.

This research uses DCNN to diagnose AMR using WGS data with high accuracy and high speed, even running on a personal computer to show practicality for daily clinical use. We find that DCNNs' high efficiency in handling large inputs make them especially well-suited for the AMR diagnosis problem. We use the large scale public biological database, PATRIC [17], for the AMR phenotype data of bacterial strains. PATRIC provides the phenotypes with accuracy over 93%, which we use directly. We use publicly available WGS data in FASTA format from the National Center for Biotechnology Information (NCBI). PATRIC and NCBI together provide abundant data, from which we used data on K. pneumoniae resistance to tetracycline, and A. baumannii resistance to carbapenem to train and validate the DCNN model. The FASTA format bacterial sequence data has about 6 million base pairs per individual strain. After tuning and training the DCNN model, it achieves an accuracy of 94.7% in AMR prediction for the evaluation datasets, while training datasets always reach 100% accuracy. With a Nvidia GPU 1080 that has 8GB of memory, it takes about 25 minutes to train the model. After training, the diagnosis, or classification in machine learning terms, takes less than one second with the aligned sequence data as input. For daily clinical usage, only the classification part is needed, so our method can provide antibiotic resistance predictions as soon as aligned WGS data is available.

Besides testing genomic data available from public databases, we also verify the DCNN model using clinically collected genomic data of Mycobacterium tuberculosis strains that are resistant or susceptible to the first-line drug pyrazinamide (PZA). The DCNN model achieved a prediction accuracy of 93.1% for PZA resistance. The M. tuberculosis genomic data is in VCF format. This test also illustrates the versatility of the model in handling different genomic data formats. To

identify genes that harbor PZA-resistant mutations, a SVM machine learning model tailored for VCF format data is built. The SVM model has identified two genes, *embB* and *gyrA*, as well as the well-known *pncA* gene, which are significantly associated with PZA resistance. The *embB* gene was also shown to associate with PZA resistance in independent research with a different dataset using statistical analysis [19].

Results

Test case 1: prediction of K. pneumoniae resistance to tetracycline

K. pneumoniae is a bacteria that normally lives inside human intestines, where it doesn't cause disease. However, if the bacteria spreads to other areas of the body, it can cause a range of different illnesses, including pneumonia. Doctors typically use antibiotics to treat K. pneumoniae infections, but with the increase of antibiotic-resistant strains, they face the difficulty of deciding what antibiotics to use. We use the DCNN model to predict resistance of K. pneumoniae strains to tetracycline using their WGS in FASTA format.

The WGS data and AMR phenotypes of 96 strains of K. pneumoniae are downloaded from PATRIC and NCBI databases. The numbers of base pairs (BP) of these strains range between 5 and 6 million, with the statistics shown in Table 1. As DCNN requires the same input size for all the strains, 0 padding is applied to make the length 6 million for all strains. Note that with classic images, cropping is frequently used to change image size; however, cropping can't be used because removing base pairs will lead to significant information loss.

Table 1: WGS data statistics of K. pneumoniae used by DCNN for training and evaluation

Minimum number of BPs/ Assembly accession	Maximum number of BPs/ Assembly accession	Mean number of base pairs	Resistant strains	Susceptible strains
4557999/GCA_003057745.1	5997388/GCA_900092975	5617511	63	33

The dataset is divided with an 80/20 ratio into training and evaluation subsets. The model is trained with the training dataset (80% of strains), and validated with evaluation dataset (20% of the strains). The evaluation data is not used in the training process. In training the model, ADM [METHOD] is used to minimize the loss function, which we observe with respect to epochs using Tensorboard Scalars, as shown in Fig. 1. It can be seen that the loss goes down rapidly toward 0. Zero loss means the model output matches the label obtained from bacterial database for all the 76 strains in the training subset, showing the convergence of the DCNN model. As each label represents the AMR phenotype of a strain, this means the model produces the correct AMR phenotypes for all the strains in training, reaching 100% accuracy for the training dataset.

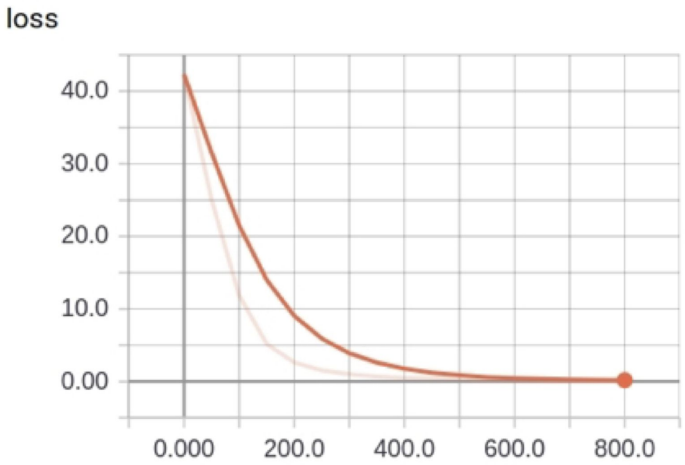

Fig. 1: loss function converges to 0 when training the DCNN model with K. pneumoniae and tetracycline pairs. The X-axis is epochs, and Y-axis is loss function.

After training finishes, the WGS data from evaluation dataset are fed into the model to calculate the output labels. The output labels are typically not exactly [0, 1] or [1, 0]. For example, the resistant strain GCA_900093365.1 has an output label of [0.08, 0.92]. These numbers represent the confidence of the prediction. In the given example, the DCNN model predicts the strain is resistant to tetracycline with a confidence level of 92%.

In validation of the model using evaluation dataset, 19 out of the 20 strains have correct prediction, resulting in an accuracy of 95%. The confidence level of the predictions is very high with a mean of 93.3%. As the dataset is not very balanced, we also calculate the Matthews Correlation Coefficient (MCC) for the evaluation dataset using the formula below:

$$MCC = (TP * TN - FP * FN)/\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)},$$

where TP is the number of true positives, TN the number of true negatives, FP the number of false positives and FN the number of false negatives.

Results are summarized in Table 2.

Table 2: DCNN test result for K. pneumoniae resistant to tetracycline

Training datasets (Num of strains)	**Evaluation datasets** (Num of strains)	**Training accuracy**	**Prediction Accuracy**	**Prediction Confidence Mean**	**MCC for training dataset**	**MCC for evaluation dataset**	**Prediction Time** (seconds)
76	20	100%	95%	93.3%	1.0	0.892	0.15

Test case 2: prediction of M. tuberculosis resistant to pyrazinamide (PZA)

Tuberculosis, caused by M. tuberculosis infections, have proven to be one of the deadliest and most widespread global diseases. PZA has long been one of the most important antibiotics for treating tuberculosis, but the efficacy of this essential drug is threatened as more and more M. tuberculosis strains develop resistance to PZA.

This test case will demonstrate how the DCNN model can determine whether a strain is PZA-resistant using its VCF genomic data. The SVM model is then used to find genes that are associated with resistance.

The genomic data from 149 *M. Tuberculosis* strains from clinical isolates are in VCF format. The dataset is very balanced with 75 resistant strains, and 74 susceptible strains. They are aligned with reference strain H37Rv. The resistance or susceptibility to PZA of these strains has previously been identified in culture based lab tests. The details on the dataset collection are described in the method section. Mutations in the resistance genes of each strain are first located from the VCF file to produce the Genomic Image as described in Data Preprocessing section.

The dataset is divided with an 80/20 ratio into training and evaluation subsets. We train the DCNN model using 120 strains from the training dataset. Tensorboard shows fast convergence of the model, with loss reaching 0 in about 5000 epochs. Loss of 0 means the model produces the correct labels for all strains in the training dataset, reaching 100% accuracy for the training dataset.

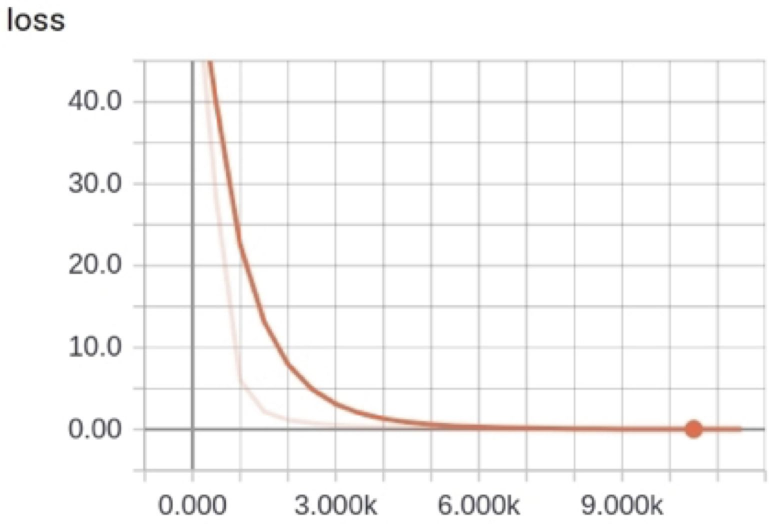

Fig. 2: Loss converges to 0 when training DCNN with M. Tuberculosis and PZA data. The X-axis is epochs, and Y-axis is the loss function. The lighter colored curve represents the real data, and the darker colored curve represents the smoothed version created by tensorboard.

For validation of the model, we then use the trained model to predict resistance for the 29 strains in the evaluation dataset. Two strains, BTB12-304 and BTB04-416, have wrong predictions, resulting in an accuracy of 93.1%.

To find the genes with mutations that contribute to PZA resistance, we then use another machine learning method, Support Vector Machines (SVM), to predict PZA resistance. We pay special attention to the below genes as listed in TB Drug Resistance Database [22]: Rv3795, Rv1267c, Rv0341, Rv3854c, Rv0006, Rv0005, Rv1694, Rv1908c, Rv2245, Rv1854c, Rv2427A, Rv2428, Rv1483, Rv1484, Rv3919c, Rv0682, Rv2043c, Rv3793, Rv3794, Rv3795, Rv0667, Rv1908c, Rv0667.

There are 298 unique mutations on these genes in the 149 M. tuberculosis strains. With 298 input features, the SVM overfits and results in poor prediction accuracy. So, we first use SVM to iteratively reduce the number of genes by removing a gene one by one if it does not contribute to the prediction accuracy. The feature reduction leads us to identify mutations on three genes that affect the PZA resistance of *M. tuberculosis*, Rv3795 (*embB*), Rv0006 (*gyrA*) and Rv2043c (*pncA*), with *pncA* being a well-known gene. There are 104 unique mutations on the three resistance genes.

Using the mutations on the three genes as feature input, the SVM model achieved prediction accuracy of 80% for the 29 strains in the evaluation dataset. Table 3 shows the prediction performance improvement by starting with only *pncA*, then adding *embB* and *gyrA*. To further show *embB* and *gyrA* are harboring PZA resistance mutations, we use mutations on them as input, and the SVM model predicts PZA resistance with 72% accuracy.

With the SVM model, we have found two novel genes, *embB* and *gyrA*, that harbor mutations for PZA resistance. The *embB* gene encodes an arabinosyltransferase involved in cell wall biosynthesis [24], and *gyrA* is a type II topoisomerase that negatively supercoils closed circular double-stranded DNA in an ATP-dependent manner to modulate DNA topology and maintain chromosomes in an underwound state [25]. Their biological relevance to PZA resistance will need further study; this paper focuses on building a framework that uses DCNN model to achieve high resistance prediction accuracy, and uses SVM model to find the genes that contribute to the resistance.

Table 3: Result of PZA resistance prediction using SVM model

Genes	Accuracy with training data	Accuracy with evaluation data	Total mutations	Unique mutations
pncA	87%	64%	77	42
pncA+embB	85%	78%	199	79
pncA+embB+gyrA	89%	80%	704	104
embB+gyrA	79%	72%	627	62

First column is the genes whose mutations are used as input for the SVM model for PZA resistance prediction. Second column is the accuracy for the training dataset. Third column is the accuracy for the evaluation dataset. Fourth column is the total mutations on the genes. Fifth column is the unique mutations on the genes, which is also the features of the SVM model. It can be seen that the inclusion of the two novel genes makes PZA resistance prediction much more accurate.

Test case 3: prediction of A. baumannii resistance to carbapenem

A. baumannii infections typically occur in intensive care units and healthcare settings housing very ill patients. It causes a variety of diseases, ranging from pneumonia to serious blood or wound infections, and can cause or contribute to death. Carbapenem antibiotics are one of the most important therapeutic options for serious infections caused by A. baumannii, but many strains are now resistant to them. This test case shows the prediction of A. baumannii strains resistance to carbapenem using the DCNN model.

86 strains of A. baumannii are used in this test, of which 31 strains are resistant and 55 strains are susceptible. The dataset is divided with an 80/20 ratio into training and evaluation subsets. In training, the ADM algorithm (METHOD) reduces the loss toward zero rapidly as shown in

Fig. 3. The loss of 0 means the model produces the correct label for all 68 strains in the training dataset, reaching 100% accuracy for the training dataset.

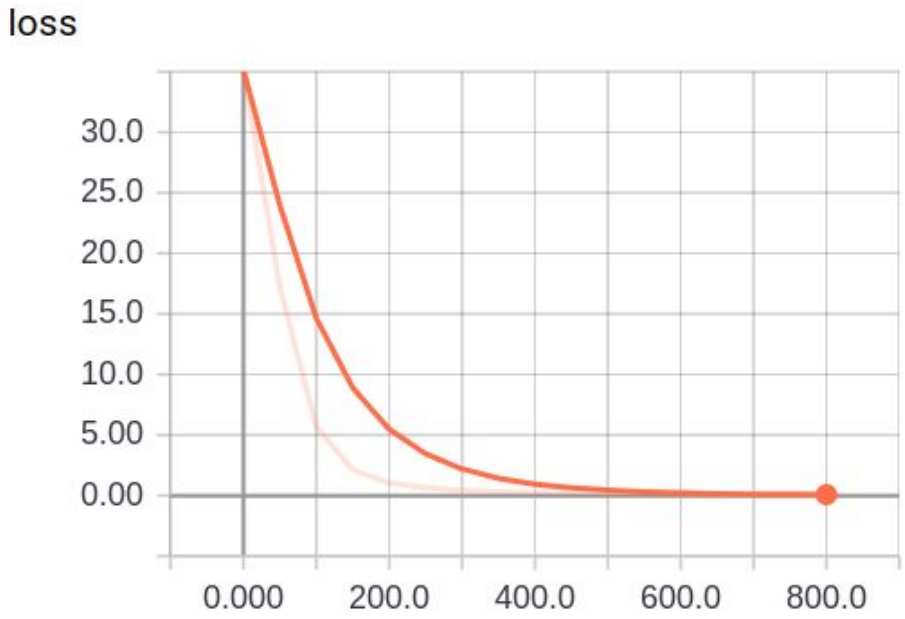

Fig. 3: loss function converges to 0 when training the DCNN model with A. baumannii and carbapenem pairs. The X-axis is epochs, and Y-axis is the loss function. The lighter colored curve represents the real data, and the darker colored curve represents the smoothed version created by tensorboard.

To validate the model, the WGS data from evaluation dataset are fed into the trained model to predict resistance. 1 out of 18 strains has wrong prediction, achieving an accuracy of 94.4%. Detailed results, including Matthews Correlation Coefficient, are listed in Table 4.

Table 4: DCNN test result for A. baumannii Resistance to carbapenem

Training datasets (Num of strains)	**Evaluation datasets** (Num of strains)	**Training accuracy**	**Prediction accuracy**	**Prediction confidence mean**	**MCC for training dataset**	**MCC for evaluation dataset**	**Prediction time** (seconds)
68	18	100%	94.4%	93%	1.0	0.892	0.15

Discussion

AMR is an urgent threat to human health as antibiotics, a critical tool for treating diseases, become ineffective. The problem is particularly dangerous as diagnosis of AMR takes days or even a month with a traditional culture-based method. Thus, there is a need to rapidly determine a strain's AMR phenotypes by clinics. This research aims to diagnose AMR infections rapidly and accurately using bacterial WGS. By converting WGS to Genomic Images, we tackle the diagnosis problem as an image classification problem and use the advanced DCNN to achieve this goal.

Data from three bacteria and antibiotics pairs are used to tune, train, and evaluate the DCNN model. By experimentation with different architectures, we build a DCNN model that fits the unique characteristics of images encoded from WGS. We divide our datasets with a ratio of 80/20 to training and evaluation subsets. The model is shown to converge rapidly in training with training datasets. In validation with the evaluation datasets, the model reaches an average accuracy of 94.2% on AMR prediction and each prediction takes just 0.15 seconds on a PC. The fact that the model worked for all three randomly picked bacteria and antibiotic pairs shows that the DCNN can be used to accurately and rapidly diagnose AMR phenotype.

The DCNN model developed for AMR prediction has a major difference from the ones for classic image classification – pooling layers are not used. In our modeling finetune, we observed

that pooling layers would significantly reduce AMR prediction accuracy. An explanation is, with traditional images, max-pooling acts as a max filter that selects the max value of a subregion, which results in an abstract representation of the "brightest" parts of the image. However, with GI, every value, max or minimum of a subregion, has equal importance as they are encoded from different base pairs which are all critical. A max filter would cause information loss with GI. To reduce training parameters and to avoid overfitting, we use a large stride in the last convolutional layer instead of using pooling at each layer.

RF with K-mers is another ML based approach for AMR prediction problems [16] in the literature. Comparing DCNN with the method presented in [16], the advantage of DCNN is its very short prediction time on a regular PC with each prediction calculation under a second.

Besides building a DCNN model that predicts AMR phenotypes rapidly and accurately, this research applied a SVM model to find genes that harbor resistant mutations. The SVM model has identified two novel genes, *embB* and *gyrA*, besides the well-known *pncA* gene, which are all strongly related to PZA resistance of M. tuberculosis.

With next-generation sequencing machine becoming cheaper and faster, WGS of bacteria has steadily become more available. The DCNN model can be run on a PC together with modern sequencing machines in clinics to get AMR results in a patient's first visit, instead of waiting two days or even up to a month for a detailed culture-based test report. Fast diagnosis will help save patients' lives and prevent the outbreak of AMR infections. It is likely that culture-based

diagnosis will co-exist sometimes, but the DCNN model can be a valuable tool to guide initial treatment, which is the most crucial for patient recovery.

Conclusions

This research demonstrates the potential of using DCNNs, a deep learning method, to diagnose AMR based on genomic data. By tackling the AMR diagnosis problem as a Genomic Image-classification problem, our machine learning framework makes a diagnosis in under a second on a PC, with an average accuracy of 94.2%. We verified the model using bacterial genomic data from public databases, as well as genomic data collected from clinical isolates. For genomic data represented in the VCF format, we have built an SVM model to identify genes that harbor resistance mutations, and identified two novel genes besides the known *pncA* gene that are related to PZA resistance.

In the future, work could be done to find which mutations on the novel genes caused the PZA resistance, and the biological mechanisms involved.

Methods

Collect Training and Evaluation Data for DCNN model

For a DCNN model to make accurate predictions, high quality training and evaluation data are needed. AMR phenotype data is retrieved from PATRIC [17], and WGS data from NCBI. PATRIC provides resistance or susceptible with accuracy over 93%, which we use directly. Biological research grants typically require that data collected be made available to the public [16]. PATRIC provides a large-scale integration platform for researchers to share their data,

which also facilitates access to high-quality data accumulated and verified by past researchers. We retrieve the AMR phenotype data of bacterial strains from PATRIC, and then cross-reference the NCBI database to download its FASTA format WGS file, which is a text file representing the nucleotide sequences of the strain.

Besides the FASTA data from PATRIC and NCBI, we also use clinically collected VCF format genomic data from *M. tuberculosis* strains that are resistant or susceptible to PZA to test the model. The genomic data are collected internationally from clinical isolates of patients infected with TB, and the resistance phenotypes were obtained from laboratory, using method LMBM (BACTEC), and MIC level was 100 ug/ml at PH 6.0.

The H37Rv reference genome was used to generate the VCF files. Details of the VCF file generation tools are as below.

Mapping: BWA-MEM.

Conversion to BAM: SAMTOOLS VIEW.

Sorting: SAMTOOLS SORT.

Indexing: SAMTOOLS INDEX.

Converting bam file to vcf: PILON.

Annotation: SNPEFF.

All the above packages were downloaded from online and were widely used for WCS pipelines. Parameters are defaults.

Data Preprocessing

FASTA format file starts with a single-line description, followed by sequence data. The description line is distinguished from the sequence data by a greater-than (">") symbol in the first column. A snapshot from a FASTA file is as follows:

>NC_016845.1 Klebsiella pneumoniae subsp. pneumoniae HS11286 chromosome, complete genome

GGTGGTCTGCCTCGCATAAAGCGGTATGAAAATGGATTGAAGCCCGGGCCGTGGATTCTACTCAACTTTCGT

C

As only sequence data determines AMR phenotype, a simple parser was developed using Python 3 to remove all the description lines and extract the sequence data. The resulting sequence data can be up to six million characters long. Up to six bases – each base being a character of A, C, G, or T – are then encoded to a number to keep the input dimension of the DCNN more manageable at 1 million. Each base has only four variants so the encoded numbers range from 1 to 4^6 = 4096 when six bases are encoded to a number.

The one-dimensional integer array of size 1,000,000 is then converted to 1000x1000 two-dimensional array to make it an "image", and thus suitable for two dimensional convolutions in DCNN. The two-dimensional representation of genomic data will be referred to as "genomic image" or "GI" in the rest of the paper. This genomic image retains all original FASTA sequence information as decoding the integers will recover the original sequence data. However, the genomic image is in a format ready for use as the DCNN model's input.

The label of each genomic image represents either that the strain is resistant or that is susceptible to a particular antibiotic. We use vectors to represent the label, [0, 1] for resistant and [1, 0] for susceptible. As an illustration in Fig. 4, a short section of a WGS (24 out of 6 million bases) is converted to a 2x2 GI, and then mapped to a label.

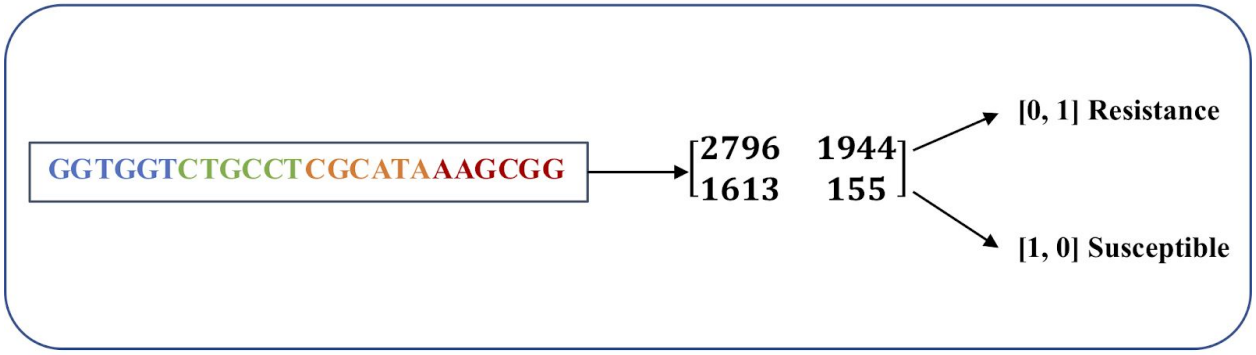

Fig. 4 Example of WGS data to Genomic Image Conversion and Map to Classification Labels

The conversion of VCF file to genomic image is slightly different. The VCF file contains mutations and their positions in a strain. A VCF parser was developed to extract all the mutations and positions on the resistance genes as listed in the resistance gene database [20]. A snapshot of the parsed result for strain BTB12-304 is as below:

POS, MUTATION (reference alternate)
7362, G/C
7585, G/C
8688, G/T

After parsing all the VCF files, the union of all the mutations is the set of features for the DCNN. For each strain, a feature is set to 1 if the mutation exists or 0 if the mutation does not exist. The

arrays consisting of 0s and 1s of all the strains are the input for the DCNN model, so the Genomic Image consists of 0s and 1s for VCF data. The label of each strain is mapped similarly as for the FASTA case: [0, 1] for resistant and [1, 0] for susceptible.

Design a Deep Convolutional Neural Network for AMR Prediction

We will design a DCNN model that solves the AMR prediction problem. The genomic data from the bacterial strains are first converted to genomic images, which are used as inputs to the DCNN.

The DCNN model, as shown in Fig. 5, consists of four convolutional layers, each with five 4x4 filters to extract features from the input genomic images. At the last convolutional layer, a pooling layer with a stride of 20x20 is used to reduce the genomic image dimension significantly, down to 50x50, which is then reshaped to a 2500 one-dimensional array before a fully connected layer that generates the output label.

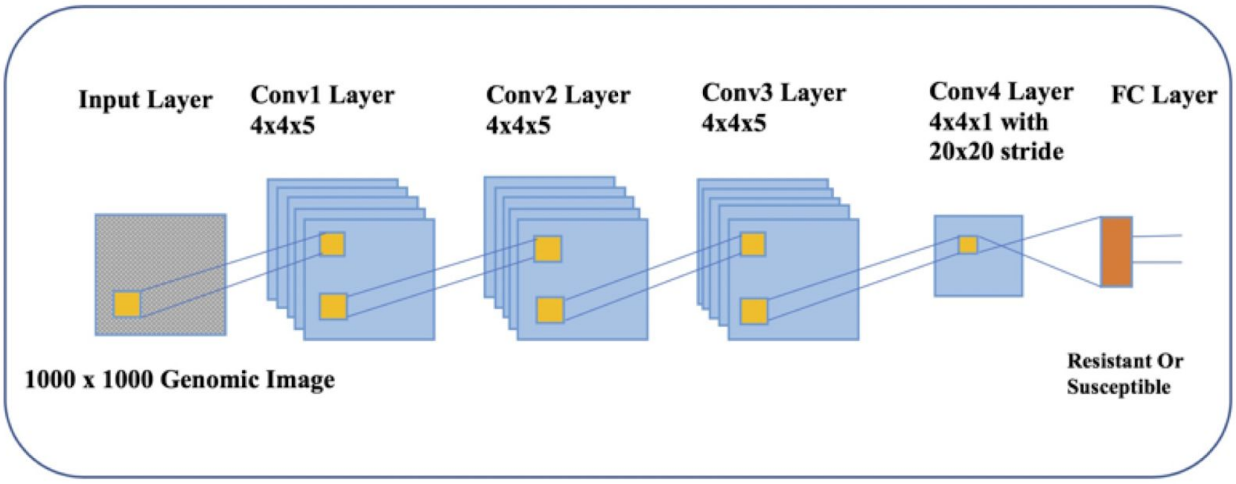

Fig. 5 DCNN Model for AMR Classification

The DCNN model is designed with the below characteristics to fit bacterial WGS data:

1. A flexible WGS encoding scheme limits the genomic image size to 1 million. When a genome has 6 million base pairs (BP), 6 BPs are encoded to a number. The encoder ensures that input size of the DCNN is always limited to one million, regardless of the bacteria's genome size, so the same model is applicable to any bacteria species.
2. Numerous small filters (also called kernels) are used in the four convolutional layers. The small filter sizes use a smaller number of training parameters. Each 4x4 filter uses only 16 parameters. Less parameters allow for faster training and a smaller memory requirement.
3. Before the fully connected (FC) layer, the last convolutional layer has a large stride to reduce the image size. As the output has two categories, resistant or susceptible, the FC layer has a small parameter size. The large stride aids the model in handling the large genomic image size (1000x1000).
4. AMR phenotypes are represented with one-hot vectors, a representation of categorical variables as binary vectors that is widely used in ML for categorization problems [14].

Implement the DCNN Using Tensorflow

Using the training and evaluation datasets, and the DCNN model previously laid out, we implement and test the model. Google's machine learning platform Tensorflow is chosen for implementation due to its various tutorials, large inventory of open source code for reference, and rich APIs for DCNN that can be directly used to speed up implementation [21].

At each convolutional layer, the tensorflow API nn.conv2d() performs two dimensional convolution using specified filter size and number of filters to the input. Neural activation function API, nn.sigmoid(), is used to generate neural output at every layer. Sigmoid activation is chosen because its output range is always between 0 and 1; we do not need to do normalization at every layer, which greatly helps speedy convergence.

At the last convolutional layer, pooling is added to reduce the dimension of the genomic image size. After pooling, the genomic image is reshaped back to a one dimensional array using the tensorflow API reshape().

After the pooling and reshaping, the last layer is a fully connected layer that generates the output. In a fully connected layer, each input neuron is connected to all output neurons, so this layer requires the largest number of training parameters in the network. The pooling helps to minimize this number.

The loss function is defined to minimize the difference between the model output and the label. A quadratic loss function is used, taking the difference between model output and label, squaring, and adding together. The loss function is implemented using the predefined quadratic and sum functions in Tensorflow, tf.square and tf.reduce_sum.

The predefined optimization method in Tensorflow, tf.train.AdamOptimizer, performs adaptive moment estimation. We use it to adjust the training parameters to minimize the loss function. Adam is a popular method of optimization with DCNNs due to its adaptive learning rate [24].

Implement SVM using SKLEARN

We use off-the-shelf SVM from sklearn module in Python [23]. The fit API is used to train the model, and the predict API is used for prediction. The parameters for the SVM model is: kernel = linear, and tol = 0.0000001 (tolerance), and all other parameters are kept as default.

Train and Evaluate the DCNN for AMR Prediction Model

For FASTA format data, we use 96 strains of K. pneumoniae with corresponding tetracycline resistance data from PATRIC and NCBI databases; for VCF format data, we use 149 clinical isolates of M. tuberculosis with corresponding PZA resistance data.

DCNN Graph

Tensorflow has a utility called Tensorboard graph that shows the Neural Network graph, a visualization of the DCNN. We will break up our DCNN graph into two parts to explain.

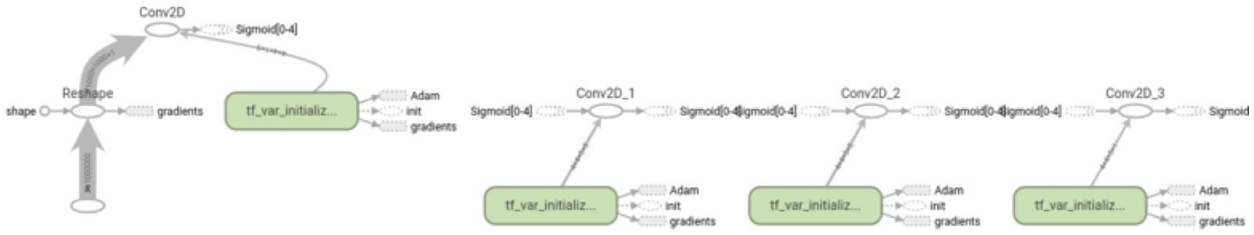

Fig. 6: The Tensorboard graph that shows the convolutional layers of the DCNN.

The left most node performs reshaping. Encoded genome data is reshaped into a genomic image, which is similar to images used in classic Image Classification, albeit with much larger dimensions at 1000x1000. After reshaping, there are four convolutional layers, Conv2D, Conv2D_1, Conv2D_2, and Conv2D_3, each with five filters. Each filter is a 4x4 matrix whose

values are initialized by a tensorflow provided function tf_var_initializer, and then optimized iteratively using the predefined optimization method AdamOptimizer. The last convolutional layer also includes pooling function to reduce the size of the genomic image before fully connected layer, though it is not shown in the graph because it is part of Conv2D_3.

The graph of the fully connected layer and the loss function is shown in Fig. 7.

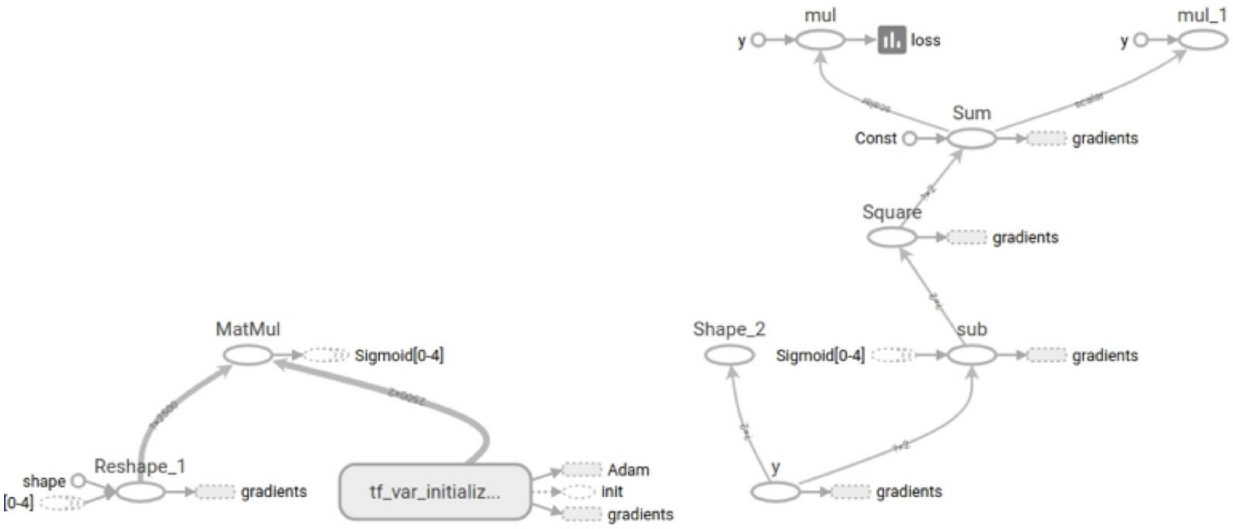

Fig. 7: The Tensorboard graph that shows the fully connected layer and the loss function

The output from Conv2D_3 layer in Fig. 6 is first reshaped to a one-dimensional array by the Reshape_1 node on the leftmost side, whose output is multiplied by a matrix shown as the node MatMul. This is the fully connected (FC) layer. The graph on the right is the loss function. The difference of the output from FC layer and labels y is calculated by the node “sub” for subtraction, which is then used to calculate the square loss function, represented by the Square and Sum nodes.

Loss Function Value Observation

In training the DCNN model, the objective is to minimize the loss function. Ideally, the loss function, which represents how close the model output matches the labels of the training dataset, should trend towards 0. During initial development, while model is being tuned and debugged, it is important that the loss can be monitored while training is going on. The Tensorboard Scalars tool is used to monitor scalars in the DCNN model. We use tf.summary.scalar to define the loss as the scalar to be monitored, and then use tf.summary.FileWriter to write the scalars to files, which can then be viewed from tensorboard either during training or after training finishes. The Tensorboard scalar monitoring tool is helpful for DCNN model debugging and tuning.

References

1. The Economist, Antibiotic use is rapidly increasing in developing countries, 2018

2. Lucas IA (1972) The use of antibiotics as feed additives for farm animals. Proc Nutr Soc 31:1–8

3. Landers TF, Cohen B, Wittum TE, Larson EL (2012) A review of antibiotic use in food animals: Perspective, policy, and potential. Public Health Rep. doi: 10.1177/003335491212700103

4. Wang Y, Hu Y, Zhu B, Jiao X, Gao GF (2018) Antibiotic resistome in farm animals and their related environments: a review. Sheng Wu Gong Cheng Xue Bao 34:1226–1233

5. The Centers for Disease Control and Prevention, Antibiotic/Antimicrobial Resistance, 2018. https://www.cdc.gov/drugresistance/index.html

6. United Nations meeting on antimicrobial resistance. Bull World Health Organ. 2016;94(9):638-9.

7. Didelot X, Bowden R, Wilson DJ, Peto TEA, Crook DW. Transforming clinical microbiology with bacterial genome sequencing. Nat Rev Genet. 2012;13(9):601-12.

8. Lin RY, Nuruzzaman F, Shah SN (2009) Incidence and impact of adverse effects to antibiotics in hospitalized adults with pneumonia. J Hosp Med 4:E7–E15

9. Costelloe C, Metcalfe C, Lovering A, Mant D, Hay AD (2010) Effect of antibiotic prescribing in primary care on antimicrobial resistance in individual patients: systematic review and meta-analysis. BMJ 340:c2096–c2096

10. Bierut LJ, Agrawal A, Bucholz KK, Doheny KF, Laurie C, Pugh E, et al. A genome-wide association study of alcohol dependence. Proc Natl Acad Sci U S A. 2010;107(11):5082-7.

11. Lavender NA, Rogers EN, Yeyeodu S, Rudd J, Hu T, Zhang J, et al. Interaction among apoptosis-associated sequence variants and joint effects on aggressive prostate cancer. BMC

12. LeCun Y, Bottou L, Bengio Y, Haffner P, Gradient-based learning applied to document recognition, 1998; https://ieeexplore.ieee.org/document/726791

13. Rawat W, Wang Z. Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. Neural Comput. 2017;29(9):2352-449.

14. Krizhevsky A, Sutskever I, Hinton G., ImageNet Classification with Deep Convolutional Neural Networks, Advances in neural information processing systems. 2012

15. Gómez-Bombarelli R, Wei JN, Duvenaud D, Hernández-Lobato JM, Sánchez-Lengeling B, Sheberla D, et al. Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. ACS Cent Sci. 2018;4(2):268-76.

16. Santerre JW, Davis JJ, Xia F, Stevens R, Machine learning for antimicrobial resistance. arXiv preprint arXiv:1607.01224. 2016. https://arxiv.org/abs/1607.01224.

17. Gillespie JJ, Wattam AR, Cammer SA, Gabbard JL, Shukla MP, Dalay O, et al. PATRIC: the comprehensive bacterial bioinformatics resource with a focus on human pathogenic species. Infect Immun. 2011;79(11):4286-98.

18. Pesesky MW, Hussain T, Wallace M, Patel S, Andleeb S, Burnham CD, et al. Evaluation of Machine Learning and Rules-Based Approaches for Predicting Antimicrobial Resistance Profiles in Gram-negative Bacilli from Whole Genome Sequence Data. Front Microbiol. 2016;7:1887.

19. Sheen P, Requena D, Gushiken E, Gilman RH, Antiparra R, Lucero B, et al. A multiple genome analysis of Mycobacterium tuberculosis reveals specific novel genes and mutations associated with pyrazinamide resistance. BMC Genomics. 2017;18(1):769.

20. Sandgren A, Strong M, Muthukrishnan P, Weiner BK, Church GM, Murray MB. Tuberculosis drug resistance mutation database. PLoS Med. 2009;6(2):e2.

21. Abadi M, Barham P, Chen J, et al (2016) TensorFlow: A System for Large-Scale Machine Learning. 265–283

22. Kingma DP, Ba J (2014) Adam: A Method for Stochastic Optimization.

23. Pedregosa F, Varoquaux G, Gramfort A, et al (2011) Scikit-learn: Machine Learning in Python. J Mach Learn Res 12:2825–2830

24. Jia B, Raphenya AR, Alcock B, Waglechner N, Guo P, Tsang KK, et al. CARD 2017: expansion and model-centric curation of the comprehensive antibiotic resistance database. Nucleic Acids Res. 2017;45(D1):D566-D73.

25. Ramaswamy SV, Amin AG, Göksel S, Stager CE, Dou SJ, El Sahly H, et al. Molecular genetic analysis of nucleotide polymorphisms associated with ethambutol resistance in human isolates of Mycobacterium tuberculosis. Antimicrob Agents Chemother. 2000;44(2):326-36.

26. Lau RW, Ho PL, Kao RY, Yew WW, Lau TC, Cheng VC, et al. Molecular characterization of fluoroquinolone resistance in Mycobacterium tuberculosis: functional analysis of *gyrA* mutation at position 74. Antimicrob Agents Chemother. 2011;55(2):608-14.