UPEC
UNIVERSITÉ PARIS-EST CRÉTEIL

FACULTÉ
DE SCIENCES ÉCONOMIQUES
ET DE GESTION

Master thesis

Detecting antibiotic resistance

Authored by: Mohamed Amine BOUSSOUFA & Amine AYADEN

Under the direction of Ms. Sophie LARUELLE

2023 - 2024

Acknowledgements

We would like to extend our heartfelt thanks to Université Paris-Est Creteil for providing us with the essential support and resources to successfully complete our thesis within the Master 2 MASERATI program.

Our deepest gratitude goes to Ms. Sophie Laruelle for her exceptional supervision and unwavering encouragement throughout this project. Her dedication, availability, and expert guidance have been indispensable, and we are truly thankful for her commitment to our success.

We would also like to express our appreciation to the entire Maserati master's teaching staff for their passion and dedication to our learning experience and professional development. Their efforts have not only enriched our academic journey but have also equipped us with the skills and knowledge needed for our future endeavors.

Lastly, we would like to thank all those who have supported and accompanied us in the completion of this thesis. Their contributions have been invaluable, and their encouragement has been a significant factor in the success of our research work. We offer our sincerest thanks and appreciation to each and every one of them.

Table of contents

Acknowledgements 2

Introduction 4

I. Understanding Antimicrobial Resistance.. 5

A) Understanding Antimicrobial Resistance 5

1. Definition and Mechanisms 5

2. Causes of Resistance Development 5

B) The Global Impact of AMR.. 5

1. Health Impacts 5

2. Economic Burden... 5

3. Challenges in Low-Resource Settings 6

C) The Need for Improved Detection.. 6

1. Limitations of Current Methods.. 6

2. Role of Innovation 6

II. Literature review:... 7

A). Prediction of antimicrobial resistance 7

B). Neural embeddings 9

III. Methodology 12

A) Embedding method... 12

B) DATASET 15

C) Machine learning and model evaluation.. 18

IV. Results and discussion:.. 21

A) Results 21

Examining the Overall Performance and Additional Metrics of the Models: 23

B) Discussion:.. 25

C) Limits.. 26

V. Conclusion.. 28

V. References.. 29

VII. Appendices: 30

Introduction

Antibiotic resistance has emerged as one of the most formidable public health challenges of the 21st century. As bacteria evolve to resist the effects of antibiotics, the arsenal of effective treatments available to combat infections dwindles, leading to longer hospital stays, higher medical costs, and an increased mortality rate. The World Health Organization (WHO) has identified antibiotic resistance as a significant global health threat, which requires urgent and coordinated action across all sectors of society.

Traditional methods for detecting antibiotic resistance typically involve culturing bacteria and testing their response to various antibiotics, a process that can take days. This delay can be critical, as it hampers the timely administration of effective treatment, thereby increasing the risk of severe outcomes. Moreover, these methods require sophisticated laboratory equipment and highly skilled personnel, limiting their accessibility in resource-poor settings.

Deep learning, a subset of machine learning characterized by algorithms and neural networks that can learn from and make decisions based on large sets of data, offers a promising solution to these challenges. Its ability to identify patterns and anomalies in complex datasets has revolutionized fields ranging from natural language processing to image recognition, making it a potent tool for medical diagnostics.

Recent advancements in deep learning have paved the way for its application in detecting antibiotic resistance. By leveraging large datasets of bacterial strains and their antibiotic response profiles, deep learning models can potentially predict resistance patterns more quickly and accurately than traditional methods. These models can analyze complex biological data, such as genomic sequences or growth patterns, to identify markers of resistance. Furthermore, deep learning can enhance the interpretation of diagnostic images, such as electron microscopy scans of bacteria, to detect subtle features that indicate resistance.

This thesis aims to explore the application of machine learning techniques to detect antibiotic resistance, focusing on developing models that can accurately predict resistance from genomic data.

The significance of this study lies in its potential to contribute to the global fight against antibiotic resistance by providing healthcare providers with tools that allow for quicker and more accurate diagnosis. This could lead to better patient outcomes through the timely administration of appropriate therapies and could also help in curbing the spread of resistant bacterial strains.

I. Understanding Antimicrobial Resistance

A) Understanding Antimicrobial Resistance

1. Definition and Mechanisms

Antimicrobial resistance (AMR) occurs when microorganisms such as bacteria, fungi, viruses, and parasites evolve to resist the effects of medications, rendering standard treatments ineffective. This resistance can arise from genetic mutations or the acquisition of resistance genes from other microbes, often mediated by mobile genetic elements like plasmids. Such adaptations allow these organisms to survive and multiply even in the presence of antimicrobial agents.

2. Causes of Resistance Development

The escalation of AMR is influenced by several critical factors:

- Overuse and Misuse of Antibiotics: Excessive and inappropriate antibiotic use in both humans and animals is a primary driver of AMR. Studies suggest that more than 50% of antibiotic prescriptions in some healthcare settings are either unnecessary or incorrect (Source: CDC, 2019).
- Agricultural Factors: The widespread use of antibiotics in agriculture, particularly for prophylactic purposes and as growth promoters in livestock, significantly contributes to the development of AMR.
- Global Travel and Trade: These elements facilitate the swift movement of resistant organisms across borders, complicating the containment and management of AMR outbreaks globally.

B) The Global Impact of AMR

1. Health Impacts

AMR is responsible for approximately 700,000 deaths annually, with projections suggesting that this number could rise to 10 million by 2050 if current trends persist (Source: Review on Antimicrobial Resistance, 2016). The management of infections caused by resistant microbes often requires longer hospital stays, the use of more expensive or toxic drugs, and can lead to significantly higher mortality rates.

2. Economic Burden

The economic impact of AMR extends beyond direct healthcare costs, encompassing substantial losses in productivity and a profound effect on global economic stability. Estimates suggest that

AMR could cost the global economy up to $100 trillion by 2050 if no effective mitigation strategies are implemented (Source: World Bank, 2017).

3. Challenges in Low-Resource Settings

The burden of AMR is particularly acute in low- and middle-income countries where healthcare systems are often under-resourced and overburdened. Limited access to effective antimicrobials and diagnostic tools, coupled with insufficient healthcare infrastructure, exacerbates the spread of resistance.

C) The Need for Improved Detection

1. Limitations of Current Methods

Traditional methods for detecting AMR, such as culture and sensitivity testing, are labor-intensive, time-consuming, and often limited to a subset of bacterial species that can be cultured in lab settings. These methods can delay critical treatment decisions and allow for the further spread of resistance.

2. Role of Innovation

Innovative technologies, particularly deep learning, hold the potential to revolutionize the detection of AMR. By leveraging complex datasets, such as genomic sequences and diagnostic images, deep learning algorithms can identify patterns of resistance much faster than traditional methods. This could drastically reduce the time to diagnosis and improve the precision of treatment strategies.

The pressing global challenge of AMR demands innovative solutions for rapid and accurate detection. This thesis explores the potential of deep learning techniques to address this urgent need, aiming to enhance the effectiveness of antimicrobial interventions and significantly contribute to global health security.

II. Literature review:

A). Prediction of antimicrobial resistance based on whole-genome sequencing and machine learning (2021)

The study by Ren et al. (2021) titled "Prediction of antimicrobial resistance based on whole-genome sequencing and machine learning" conducted a comprehensive evaluation of four machine learning algorithms - convolutional neural networks (CNN), random forests (RF), and support vector machines (SVM) - for predicting antibiotic resistance in Escherichia coli using whole-genome sequencing data.

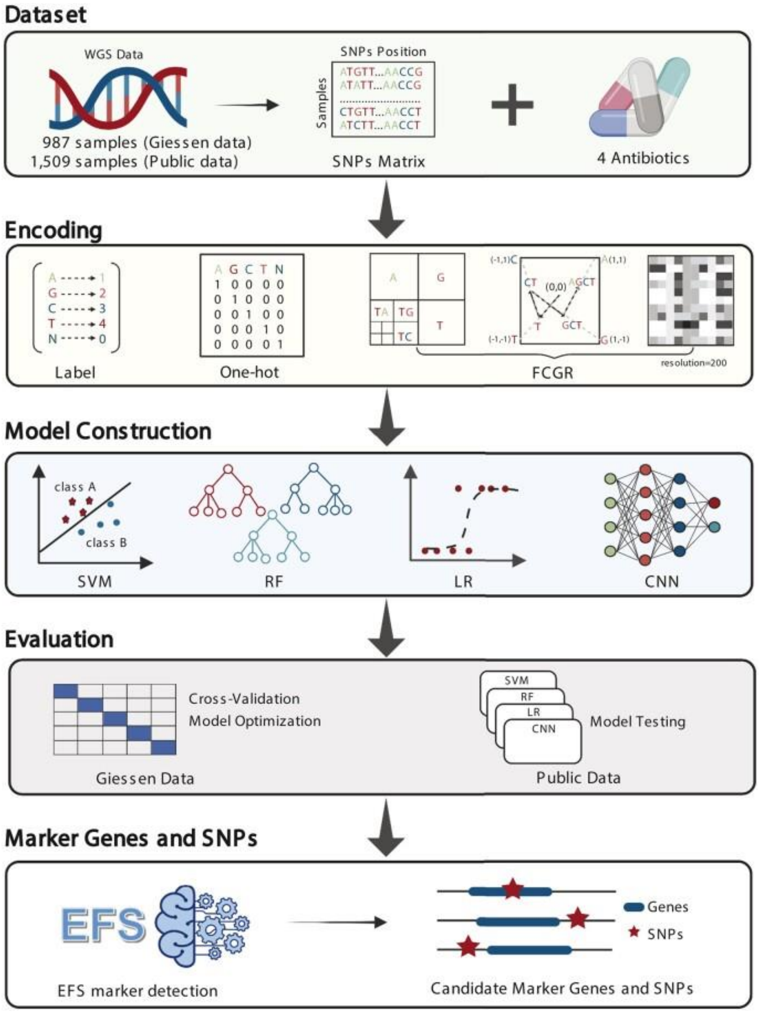


Fig.1: Workflow of the study Prediction of antimicrobial resistance based on whole-genome sequencing and machine learning (2021)

The workflow of the study is shown in Figure 1.

The authors used two datasets - an in-house dataset from Giessen University of 987 E. coli isolates (Giessen data) and a public dataset of 1,509 isolates from Moradigaravand et al. (2018). They focused on predicting resistance to four important antibiotics: ciprofloxacin (CIP), cefotaxime (CTX), ceftazidime (CTZ), and gentamicin (GEN).

Three different encoding schemes were evaluated to represent the genomic sequences as input features for the machine learning models:

- Label encoding
- One-hot encoding
- Frequency matrix chaos game representation (FCGR) encoding

The model performance was assessed using 5-fold cross-validation on the Giessen data and validated on the independent public dataset. Precision, recall, and area under the ROC curve (AUC) were the key evaluation metrics.

For the Giessen data, the results with label encoding showed that:

RF models achieved the highest precision scores across all four antibiotics, significantly outperforming CNN, LR, and SVM for CIP resistance prediction.

RF also exhibited the highest recall values for CTX, CTZ, and GEN resistance.

CNN and LR demonstrated comparable performance to RF for certain antibiotics.

With one-hot encoding on the Giessen data:

- RF again showed the highest precision for CIP, CTX, and CTZ resistance prediction.
- For recall, RF matched or exceeded the other algorithms across antibiotics.
- CNN and LR were competitive, with CNN having the highest recall for GEN resistance.

For FCGR encoding on the Giessen data:

- RF displayed the highest precision for all four antibiotics.
- RF also achieved the best recall for CIP, CTZ, and GEN resistance prediction.
- CNN demonstrated the highest recall for CTX resistance prediction.

When evaluated on the balanced public dataset (down-sampled for class imbalance):

- RF consistently showed the highest precision and recall across most antibiotics and encoding schemes.
- CNN, LR, and SVM were generally competitive but did not outperform RF.

Furthermore, the authors conducted an SNP association analysis to identify putative resistance-linked genetic markers without relying on prior knowledge of resistance genes. Several of the top-ranked SNPs mapped to genes previously implicated in antibiotic resistance, such as nhaA, rlmC, fliI, pepB, and murB, providing biological validation.

The key strengths of this rigorous study include:

- Comprehensive benchmarking of four machine learning algorithms on whole-genome data across multiple antibiotics and encoding representations.
- Evaluation using nested cross-validation on the primary dataset and validation on an independent test set.
- Unbiased discovery of potential resistance-associated genetic variants through data-driven SNP association analysis.
- Identification of known resistance genes among the top SNP markers, confirming the approach's validity.
- Demonstration of RF models' consistent superior performance across different antibiotics and encoding schemes.

Overall, this study by Ren et al. (2021) provides a robust machine learning framework, notably highlighting the efficacy of random forest models, for leveraging whole-genome sequencing to predict and elucidate antibiotic resistance determinants in the major pathogen E. coli. Their findings pave the way for improved clinical decision support and antimicrobial stewardship efforts.

B). Neural embeddings for efficient DNA data compression and optimized similarity search (2024)

The study titled "NeuralBeds: Neural embeddings for efficient DNA data compression and optimized similarity search" by Sarumi et al. (2024) explores the potential of deep learning techniques, specifically Convolutional Neural Networks (CNNs) and Fully Connected Networks

(FCNs), to create neural embeddings for DNA sequences. These embeddings aim to capture sequence similarity as a distance measure, enabling efficient DNA data compression and optimized similarity search. The study's motivation stems from the ever-growing repositories of DNA sequences generated by high-throughput sequencing technologies, which necessitate improved computational methods for tasks such as DNA similarity search, sequence alignment, and antimicrobial resistance (AMR) gene detection.

Sarumi et al. (2024) utilized a dataset comprising 33,860 DNA plasmids from 263 pathogens, obtained from the Comprehensive Antibiotic Resistance Database (CARD). After preprocessing and filtering, the cleaned dataset contained 3,549 DNA sequences from 47 distinct pathogens, with sequence lengths ranging from 162 to 3,594 nucleotides. The authors transformed the DNA sequences into Chaos Game Representations (CGRs), a technique that maps one-dimensional sequences into two-dimensional polygons, enabling the application of machine learning algorithms.

The study employed two neural network architectures, CNNs and FCNs, trained with two loss functions: triplet loss and ladder loss. The triplet loss function aims to minimize the distance between an anchor sequence and positive samples (sequences from the same class) while maximizing the distance from negative samples (sequences from different classes). The ladder loss function, on the other hand, ranks the sequences based on their similarity to the anchor sequence, allowing for a more flexible approach to ranking.

Sarumi et al. (2024) evaluated the performance of their neural embedding methods by comparing them with three alternative techniques: Locality-Sensitive Hashing (LSH), Principal Component Analysis (PCA), and the baseline approach of using unprocessed CGRs. The evaluation was based on a ranking task, where DNA sequences were ordered based on their distance from a query DNA sequence. The authors employed the Normalized Discounted Cumulative Gain (NDCG) and positional rank scores as evaluation metrics.

The results demonstrated that neural embeddings trained using the ladder loss function, particularly those generated by a CNN, outperformed all other approaches, including the baseline CGR method, in capturing DNA sequence similarity. Specifically, the CNN with ladder loss achieved the highest NDCG rank scores and positional rank scores for ranks one, two, and three, surpassing the baseline CGR method's performance.

Furthermore, the study assessed the models' ability to compress DNA sequences by examining the rank scores for various embedding dimensions. The results indicated that neural embeddings, especially those generated by CNNs with ladder loss, excelled in embedding sequences into lower-dimensional latent spaces while retaining essential information. However, it is noteworthy that the CNN approach required a longer processing time compared to other methods, such as PCA and LSH, highlighting the trade-off between embedding quality and computational efficiency.

As a use case, the authors compared the performance of their CNN neural embedding with ladder loss (NeuralBeds) to the widely used BLAST (Basic Local Alignment Search Tool) algorithm in terms of retrieval speed, disk storage usage, and quality of the retrieved sequences. The results showed that NeuralBeds achieved a shorter retrieval time than BLAST, with an increased sensitivity of 89% compared to BLAST's 74% sensitivity. Additionally, the NeuralBeds database required only 1.8 GB of disk space, whereas the BLAST database consumed 15 GB.

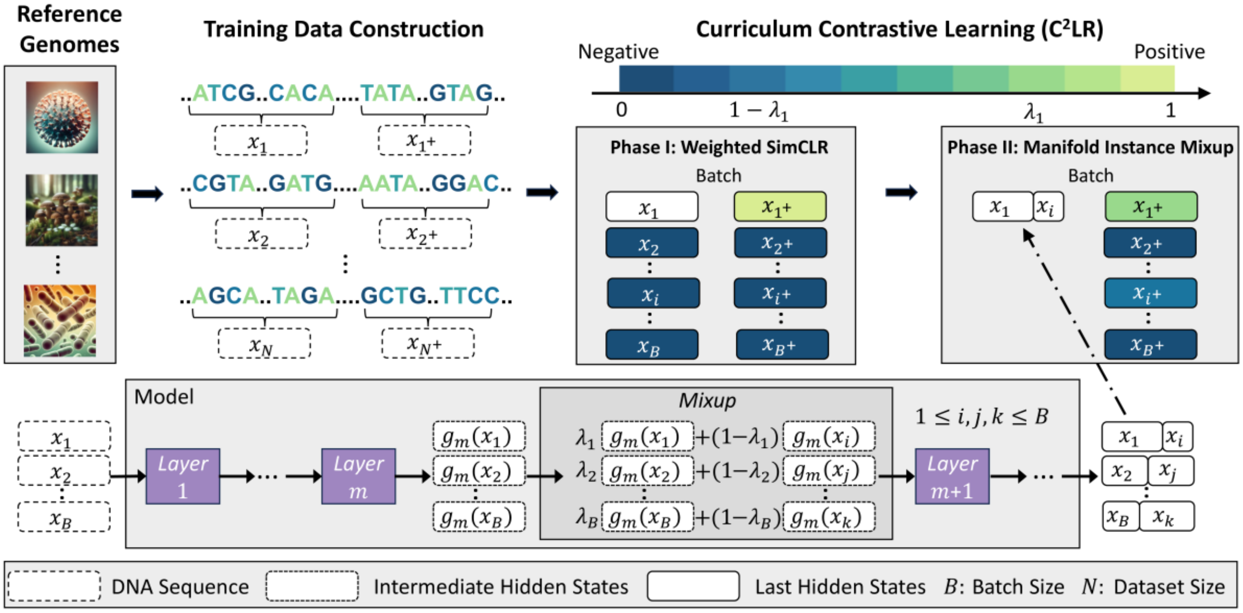
In conclusion, the study by Sarumi et al. (2024) demonstrated the feasibility and potential of using deep learning techniques, particularly CNNs trained with ladder loss, for generating semantic embeddings of DNA sequences. These embeddings not only optimize DNA similarity search but also enable efficient data compression while preserving essential sequence information. The authors' findings highlight the promise of neural embeddings as a viable approach for various bioinformatics tasks, such as DNA similarity search, sequence alignment, and AMR gene detection, in the face of rapidly growing genomic data repositories.

III. Methodology

A) Embedding method

Effective DNA sequence embedding is crucial for applying deep learning to genomic analysis tasks such as detecting antibiotic resistance in bacteria. Traditional embedding methods like one-hot encoding, tetra-nucleotide frequency (TNF), and pre-trained k-mer embeddings have certain limitations. These techniques fail to capture complex semantic relationships within DNA sequences that are essential for discerning patterns indicative of antibiotic resistance. Additionally, genome foundation models pre-trained with generic language modeling objectives often produce suboptimal embeddings for specific genomic tasks due to their inability to adapt to the nuanced differences within DNA sequences.

To overcome these limitations, we have chosen to implement DNABERT-S, a specialized genome foundation model designed to generate species-aware DNA embeddings. This choice is driven by the model's ability to effectively learn and distinguish DNA sequences through advanced embedding techniques. DNABERT-S employs a novel Curriculum Contrastive Learning (C2LR) strategy coupled with a Manifold Instance Mixup (MI-Mix) training objective. This combination enhances the model's capability to learn nuanced features of DNA sequences by initially differentiating similar from dissimilar sequences using weighted SimCLR loss and subsequently increasing the learning complexity through the MI-Mix loss that blends hidden representations at random layers, as depicted in Figure 2.



Reference Genomes
Training Data Construction
..ATCG..CACA....TATA..GTAG..
x_1
x_1+
..CGTA..GATG....AATA..GGAC..
x_2
x_2+
..AGCA..TAGA....GCTG..TTCC..
x_N
x_N+
Curriculum Contrastive Learning (C²LR)
Negative
Positive
0
$1-\lambda_1$
λ_1
1
Phase I: Weighted SimCLR
Batch
x_1
x_1+
x_2
x_2+
x_i
x_i+
x_B
x_B+
Phase II: Manifold Instance Mixup
Batch
x_1
x_i
Model
Layer 1
Layer m
$g_m(x_1)$
$g_m(x_2)$
$g_m(x_B)$
Mixup
$\lambda_1 g_m(x_1) + (1-\lambda_1) g_m(x_i)$
$\lambda_2 g_m(x_2) + (1-\lambda_2) g_m(x_j)$
$\lambda_B g_m(x_B) + (1-\lambda_B) g_m(x_k)$
$1 \leq i,j,k \leq B$
Layer m+1
x_2
x_j
x_k
DNA Sequence
Intermediate Hidden States
Last Hidden States
B: Batch Size
N: Dataset Size

Fig. 2: Overview of DNABERT-S's training process

The DNABERT-S model has demonstrated exceptional performance across various genomic datasets and tasks including species clustering, classification, and metagenome binning. Notably, it has shown to double the clustering performance and effectively identify species bins, outperforming leading baseline models. This superior performance is particularly relevant for embedding bacterial genomes in the context of antibiotic resistance detection, where differentiating between species and strains with distinct antibiotic resistance profiles is crucial. The visualization in Figure 2 illustrates the model's pronounced ability to cluster and segregate different species within the embedding space, highlighting its effectiveness.

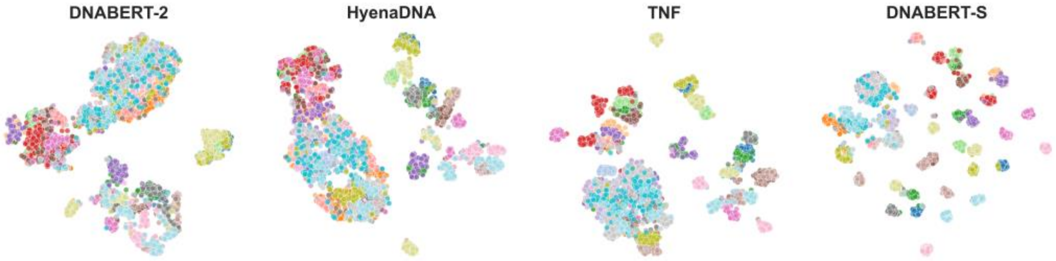


Fig. 3: TSNE visualization of the DNA embeddings generated by different methods on a CAMI2 [Meyer et al., 2022] dataset with 50 different species. Each point represents an individual DNA sequence, with the color coding indicating the species affiliation. Notably, DNABERT-S demonstrates a pronounced ability to cluster and segregate different species within the embedding space.

The species-awareness of DNABERT-S embeddings makes it particularly suitable for capturing crucial genomic features associated with antibiotic resistance. Unlike traditional methods that might miss critical genotypic signals, the embedding strategy of DNABERT-S is uniquely poised to classify antibiotic resistance effectively. Through the C2LR and MI-Mix training methodologies, DNABERT-S can detect both broad genomic differences across species and subtle variations within strains, which are indicative of resistance or susceptibility to antibiotics.

In our thesis, DNABERT-S serves as the core embedding method for analyzing DNA sequences from bacterial isolates. The species-aware embeddings generated by this model are used as input features for training machine learning models that predict antibiotic resistance. This approach is anticipated to significantly enhance the accuracy and efficiency of antibiotic resistance detection compared to existing methods. Nevertheless, to ensure a comprehensive analysis, we will also employ one-hot encoding as another embedding method for comparative evaluation.

Model Initialization and implementation:

DNABERT-S is available via the Huggingface Transformers library. To integrate this mode into our research pipeline, we leveraged the Huggingface platform's capacity to facilitate the easy deployment of state-of-the-art machine learning models. The DNABERT-S model and its corresponding tokenizer were initialized for our application. The tokenizer preprocesses the DNA sequences, adapting them into a suitable format for the model. The model then processes these formatted inputs to generate fixed-length embeddings.

Embedding process:

Each input DNA sequence, regardless of its original length, was embedded into a fixed-length vector of 768 dimensions. This dimensionality constraint stems from the transformer architecture underlying DNABERT-S, designed to manage sequences up to a predetermined length due to computational efficiency and memory considerations. This standardization ensures uniform handling of sequence data, which is critical for maintaining consistent analytical methodologies across varied datasets.

The choice of keeping the length of the DNA sequence to 768 in models like DNABERT-S, which are based on transformer architectures adapted from NLP (Natural Language Processing) models such as BERT (Bidirectional Encoder Representations from Transformers), is primarily driven by computational and architectural considerations.

For sequences shorter than 768 bases, DNABERT-S uses padding to extend the sequences to the required length. Padding involves adding extra, non-informative tokens (often zeros or a special padding token) to the end of the sequence until it reaches the required length. This ensures that all sequences input into the model are of uniform length, a necessity for batch processing in neural networks.

Additionally, a masking mechanism is employed alongside padding. This mask indicates to the model which parts of the sequence are real and which are padding. The model then ignores the padded sections during processing, focusing only on the real, biologically relevant parts of the sequence. This mechanism ensures that the padding does not affect the model’s output, preserving the integrity and relevance of the embeddings.

Uniform embedding length offers several significant benefits that enhance both the efficiency and efficacy of data handling and model performance, particularly in complex fields such as genomic studies. One of the primary advantages is the consistency it brings to data processing. By converting all input sequences to a uniform length, models like DNABERT-S can simplify the preprocessing steps and ensure consistent handling across diverse datasets and analytical scenarios. This uniformity is essential for maintaining the validity of comparisons and aggregations, which are critical in genomic research where precise and reliable data interpretation is necessary.

Moreover, setting a fixed sequence length of 768 optimizes the model architecture, particularly for transformer models designed to balance computational load and performance. This optimization is crucial for efficient training and inference processes, especially when managing large genomic datasets. Additionally, padding shorter sequences to a uniform length allows the model to learn from a standard data "format," enhancing its capability to extract valuable insights even from limited data. This aspect is particularly important in genomics, where sequence length can vary naturally due to biological factors like gene length variability. The use of a masking strategy further refines the learning process by enabling the model's attention mechanism to focus only on relevant features of the sequence, thereby improving its ability to perform specific tasks, such as predicting antibiotic resistance, with higher accuracy.

In practical terms, this approach allows DNABERT-S to produce high-quality embeddings that are informative and tailored to the specific needs of genomic research, such as antibiotic resistance prediction. The embeddings reflect the biological significance of each sequence, irrespective of its original length, thereby enabling more accurate downstream analysis, such as clustering, classification, or direct prediction tasks.

By standardizing the input sequence length and employing intelligent masking, DNABERT-S ensures that each sequence, no matter its length, contributes meaningfully to the model's training and inference processes, thus enhancing the overall quality and reliability of the genomic analysis.

B) DATASET

For our analysis, the Comprehensive Antibiotic Resistance Database (CARD) (https://card.mcmaster.ca/) was selected as the primary dataset. The Comprehensive Antibiotic Resistance Database (CARD) is an extensive, publicly accessible database dedicated to the curation of high-quality information concerning antibiotic resistance elements. It comprises a rich collection of data including detailed annotations of resistance genes, their protein sequences, associated resistance mechanisms, and affected drugs. Each entry in the CARD is meticulously annotated with metadata that encompasses a variety of variables such as gene ontology terms, links to associated scientific literature, and clinical relevance. Additionally, the database includes an ontology of antibiotic resistance elements, providing a hierarchical classification that spans from molecular structures to mechanisms of action and the genetic environments where these genes are found. The dataset also features tools for comparative analysis and visualization, which facilitate the exploration of genomic contexts and the identification of novel resistance determinants. This robust framework not only supports the identification of known resistance genes in bacterial DNA sequences but also assists in the predictive modeling of resistance based on gene presence, mutation, and expression patterns, making it an ideal resource for applying deep learning methodologies to the study of antibiotic resistance.

The CARD dataset includes several key files, each serving a unique purpose in the analysis of AMR:

1. **ARO Categories and Index Files (TSV format):**
    - aro_categories.tsv and aro_categories_index.tsv provide categorizations of the AMR genes based on the Antibiotic Resistance Ontology (ARO). These files help in identifying gene families, target drug classes, and resistance mechanisms.

- aro_index.tsv lists the GenBank accessions tagged with ARO terms, facilitating cross-referencing with other genomic databases.
- TSV (Tab-Separated Values) files are plain text files that use tabs to separate values. They are commonly used for storing structured data in a simple, readable format and are easily imported into various data analysis tools and software, making them highly relevant for bioinformatics research.

2. **FASTA Files:**
    - The dataset includes various nucleotide and protein FASTA files, such as nucleotide_fasta_protein_homolog_model.fasta, nucleotide_fasta_protein_knockout_model.fasta, and protein_fasta_protein_variant_model.fasta. These files contain sequences of resistance genes and their protein products.
    - The FASTA files are categorized based on different model types. For instance, the "protein homolog" model includes sequences of AMR genes without mutation data.
    - FASTA (Fast-All) is a text-based format for representing nucleotide or protein sequences. Each entry in a FASTA file begins with a single-line description starting with a '>' character, followed by lines of sequence data. For example:

      ```
      >gene_name description
      ATGCGTACGTAGCTAGCTAG
      ```
    - The description line provides information about the sequence, such as the gene name and additional annotations. The sequence lines contain the actual nucleotide (DNA/RNA) or protein sequences. FASTA files are widely used in bioinformatics for sequence alignment, similarity searches, and other analyses due to their simplicity and compatibility with various bioinformatics tools and databases. This makes them crucial for genomic research, including the study of antimicrobial resistance.

3. CARD JSON File:
    - card.json is a comprehensive file containing data for all of CARD's AMR detection models, including reference sequences, SNP mapping data, model parameters, and ARO classification. This file is integral to the Resistance Gene Identifier (RGI) software used for identifying AMR genes.
    - JSON (JavaScript Object Notation) is a lightweight data-interchange format that is easy for humans to read and write and easy for machines to parse and generate. It is used to store and exchange data between a server and a web application, making it suitable for storing complex datasets in a structured format.

4. SNP Data:

- The file snps.txt lists specific SNPs associated with the detection models, crucial for understanding the genetic variations that contribute to resistance, we will not be using it.

5. Shortname Files (TSV format):
    - shortname_antibiotics.tsv and shortname_pathogens.tsv contain CARD-specific abbreviations for AMR gene names and pathogen names, respectively. These abbreviations are used for programmatic purposes and ensure compatibility across different datasets and tools.

After consolidating our dataset by merging all the necessary files, we have compiled a dataset containing 5,000 DNA sequences from various bacteria. Initially, we eliminate outliers by removing sequences shorter than 10 nucleotides since the majority of sequences exhibit similar sizes, clustering around 800 to 1,200 nucleobases, as highlighted in the boxplot diagram [figure 4].

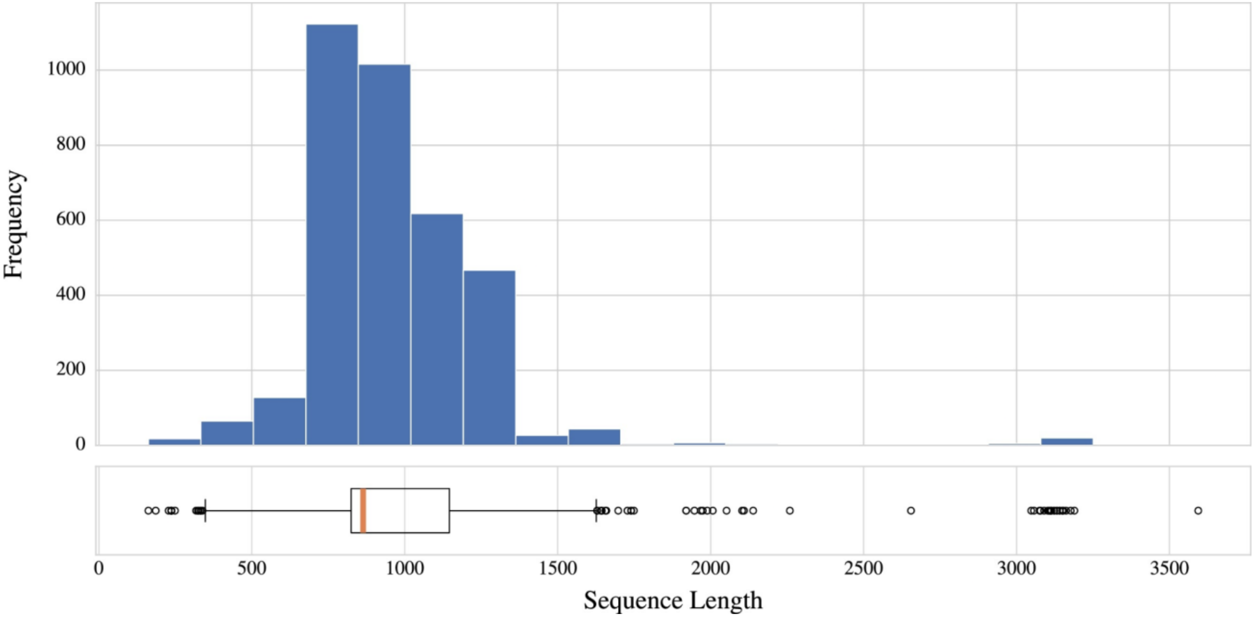


Fig. 4: Visualization of the DNA data distribution with sequence lengths spanning

Next, we clean the DNA sequences to address the presence of unfamiliar letters such as 'N', which serve as placeholders for unknown nucleotides. These characters are often used to represent ambiguities or mixed signals detected during sequencing when the specific nucleotide cannot be confidently identified. In our cleaning process, we either replace these ambiguous bases with a consensus nucleotide or remove the sequence entirely if necessary. This approach ensures that our analysis is based on accurate and reliable genetic data.

After cleaning, we apply the embedding technique as previously described.

Upon completion, we have a refined dataset of 5107 sequences. For our study, we use two primary variables: the DNA sequence (independent variable) and the antibiotics to which each bacterium is resistant (dependent variable). Given that bacteria can be resistant to multiple antibiotics, we encode this information by creating dummy variables for each antibiotic, ensuring our dataset is well-prepared for machine learning analysis.

Presented below is a structured example from the dataset employed in this research. Each row illustrates a unique bacterial sequence alongside its numerical embedding vector and the corresponding antibiotic resistance profile. The columns under each antibiotic represent the presence ('1') or absence ('0') of resistance for that particular antibiotic.

Sequence	Embedding	Aminocoum- -arin Antibiotic	Aminoglycos -ide Antibiotic	Antibacter -ial FFA	Bicyclomyc -in Antibiotic	Carbapen -em Antibiotic
ATGAAGA CA...	[0.127, - 0.231, 0.214, ...]	0	0	1	0	1
ATGCGTT AT...	[0.113, 0.197, 0.273, ...]	1	0	0	0	0
ATGATAG GT...	[0.506, -0.034, 0.040, ...]	0	1	0	1	1

Table 1: Representative sample of dataset observations

C) Machine learning and model evaluation

We employed three machine learning methods, including Random Forest (RF), Support Vector Machine (SVM) with a linear kernel, and Convolutional Neural Networks (CNN). For both RF and SVM, we utilized the Scikit-learn Python package (Pedregosa et al., 2011). The RF model was implemented with the default parameters but configured to use 200 trees. SVM was applied using a linear kernel and default parameters.

Our CNN is specifically tailored for binary multi-label classification of one-dimensional sequence data. The model architecture features two convolutional layers, each followed by max pooling to reduce dimensionality and enhance feature extraction. The first convolutional layer has 32 filters, while the second employs 64 filters, both utilizing ReLU activation functions to introduce non-linearity. Post-feature extraction, the network includes a flattening step, a dense layer with 100 neurons, and a dropout layer set at 10% to mitigate overfitting. The output layer consists of 46 neurons, corresponding to the number of labels, and uses a sigmoid activation function to provide the probability of each label. The model is compiled with the Adam optimizer and trained over 100 epochs using binary cross-entropy loss, focusing on accuracy and recall metrics. This configuration underscores the CNN's capability to efficiently process sequential data and accurately handle multiple binary classifications simultaneously.

To handle the multiclass classification task effectively, we created dummy variables from the 'Drug Class' column in our dataset. Each bit represents a possible category. If the original value was that category, the bit will be set to 1, otherwise 0. This approach allows us to transform the categorical variable, which is inherently non-numeric, into a format that can be easily used by machine learning algorithms to perform multiclass classification.

In our evaluation of multiclass classification models, we prioritized the use of precision, recall, and specifically focused on Micro Recall to select the best model. Precision is defined as $Precision = \frac{TP}{FP + TP}$, where it measures the accuracy of the positive predictions our model makes. Recall, calculated by $Recall = \frac{TP}{FN+TP}$, assesses the model’s ability to identify all relevant instances correctly. The F1 Score, which harmonizes these two metrics, is computed using the formula $F1 = 2 \times \frac{Precision \times Recall}{Precision + Recal}$.

For multiclass scenarios, particularly those involving class imbalances, Micro Recall is highly valuable. These metric computes recall globally across all classes by aggregating the total counts of true positives (TP) and false negatives (FN) from each class, using the formula:

$$\boldsymbol{Micro\ Recall = \frac{Total\,TP}{Total\,FN +\ Total\,TP}}$$

Where:

- $\boldsymbol{Micro\ Precision = \frac{Total\,TP}{Total\,FP + Total\ \ TP}}$

- $\boldsymbol{Micro\ F1 = 2 \times \frac{Micro\ Precision \times\ Micro\ Recall}{Micro\ Precision +\ Micro\ Recal}}$

In our evaluation of multiclass classification models for antibiotic resistance detection, we emphasized the importance of Micro Recall as our primary metric for model performance. This focus is especially crucial in contexts such as ours, where the consequence of missing a true positive—failing to identify a bacterial strain resistant to antibiotics—can be far more severe than the inconvenience of a false positive.

Emphasizing Micro Recall ensures that our chosen model maximizes the detection of all resistant strains, providing robustness against the potentially disastrous consequences of failing to identify resistant infections. This is critical in antibiotic resistance classification, where overlooking resistant strains could lead to ineffective treatment regimens and contribute to the broader issue of antibiotic resistance spread.

By focusing on Micro Recall, we ensure the model not only predicts accurately but also minimizes the risk of missing resistant cases, thereby enhancing its overall reliability and utility in clinical settings. This approach has been pivotal in helping us select the most effective model for our dataset, thus enabling better treatment decisions and contributing to more effective management of antibiotic stewardship programs.

IV. Results and discussion:

A) Results

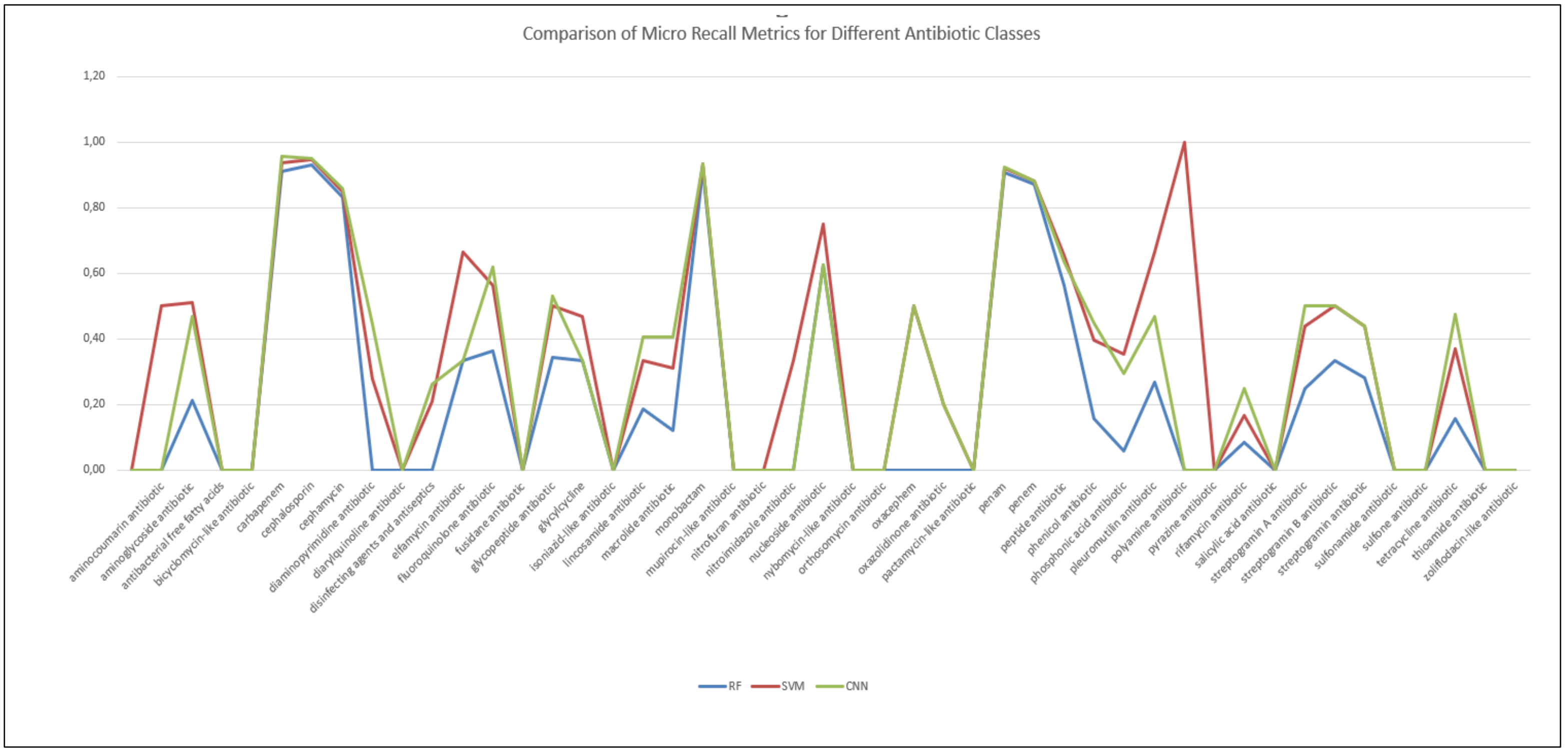


Fig. 5: Comparison of Micro Recall Metrics for Different Antibiotic Classes

Reviewing the "micro avg" recall values reveals that all three models—CNN, SVM, and RF—demonstrate high recall, with CNN and SVM outperforming RF. This high recall suggests that these models are generally effective at identifying relevant instances.

Variability in recall scores is evident across different antibiotic classes. For example, the recall for "carbapenem" is exceptionally high in all models, often nearing perfection, which suggests these models are particularly adept at recognizing this class, aided by a substantial number of observations.

	Random Forest			SVM			CNN			
Drug Class	**Precision**	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**	**Precision**	**Recall**	**F1-score**	**Support**
aminocoumarin antibiotic	0,00	0,00	0,00	0,43	0,50	0,46	0,00	0,00	0,00	6
aminoglycoside antibiotic	0,91	0,21	0,34	0,53	0,51	0,52	0,52	0,47	0,49	47
antibacterial free fatty acids	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0
bicyclomycin-like antibiotic	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	1
carbapenem	0,99	0,91	0,95	0,96	0,94	0,95	0,94	0,96	0,95	505
cephalosporin	0,98	0,93	0,96	0,95	0,95	0,95	0,95	0,95	0,95	659
cephamycin	0,99	0,83	0,90	0,88	0,85	0,86	0,95	0,86	0,90	119
diaminopyrimidine antibiotic	0,00	0,00	0,00	0,50	0,28	0,36	0,42	0,44	0,43	18
diarylquinoline antibiotic	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0
disinfecting agents and antiseptics	0,00	0,00	0,00	0,36	0,21	0,27	0,50	0,26	0,34	19
elfamycin antibiotic	1,00	0,33	0,50	1,00	0,67	0,80	1,00	0,33	0,50	3
fluoroquinolone antibiotic	0,87	0,36	0,51	0,63	0,56	0,60	0,67	0,62	0,64	55
fusidane antibiotic	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	2
glycopeptide antibiotic	1,00	0,34	0,51	0,84	0,50	0,63	0,89	0,53	0,67	32
glycylcycline	1,00	0,33	0,50	0,78	0,47	0,58	0,83	0,33	0,48	15
isoniazid-like antibiotic	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	1
lincosamide antibiotic	1,00	0,19	0,31	0,56	0,33	0,42	0,69	0,41	0,51	27
macrolide antibiotic	0,83	0,12	0,21	0,48	0,31	0,38	0,49	0,40	0,44	42
monobactam	0,99	0,91	0,95	0,96	0,93	0,95	0,98	0,93	0,96	197
mupirocin-like antibiotic	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	2
nitrofuran antibiotic	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	1
nitroimidazole antibiotic	0,00	0,00	0,00	0,50	0,33	0,40	0,00	0,00	0,00	3
nucleoside antibiotic	1,00	0,63	0,77	0,86	0,75	0,80	1,00	0,63	0,77	8
nybomycin-like antibiotic	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	4
orthosomycin antibiotic	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0
Oxacephem	0,00	0,00	0,00	0,25	0,50	0,33	0,50	0,50	0,50	2
oxazolidinone antibiotic	0,00	0,00	0,00	1,00	0,20	0,33	0,33	0,20	0,25	5
pactamycin-like antibiotic	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0
Penam	0,98	0,91	0,94	0,93	0,92	0,93	0,94	0,92	0,93	483
Penem	1,00	0,87	0,93	0,93	0,88	0,91	0,99	0,88	0,93	94
peptide antibiotic	1,00	0,56	0,72	0,95	0,65	0,77	0,95	0,64	0,76	55
phenicol antibiotic	1,00	0,16	0,27	0,65	0,39	0,49	0,71	0,45	0,55	38
phosphonic acid antibiotic	1,00	0,06	0,11	1,00	0,35	0,52	1,00	0,29	0,45	17
pleuromutilin antibiotic	1,00	0,27	0,42	0,91	0,67	0,77	0,78	0,47	0,58	15
polyamine antibiotic	0,00	0,00	0,00	0,50	1,00	0,67	0,00	0,00	0,00	1
pyrazine antibiotic	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0
rifamycin antibiotic	1,00	0,08	0,15	0,15	0,17	0,16	0,38	0,25	0,30	12
salicylic acid antibiotic	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0
streptogramin A antibiotic	1,00	0,25	0,40	0,64	0,44	0,52	0,80	0,50	0,62	16
streptogramin B antibiotic	1,00	0,33	0,50	0,60	0,50	0,55	0,86	0,50	0,63	12
streptogramin antibiotic	1,00	0,28	0,44	0,48	0,44	0,46	0,73	0,44	0,55	25
sulfonamide antibiotic	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	4
sulfone antibiotic	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	1
tetracycline antibiotic	0,86	0,16	0,27	0,67	0,37	0,47	0,56	0,47	0,51	38
thioamide antibiotic	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0
zoliflodacin-like antibiotic	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0
Micro avg	**0,98**	**0,77**	**0,87**	**0,90**	**0,83**	**0,86**	**0,90**	**0,84**	**0,87**	**2584**
Macro avg	**0,49**	**0,22**	**0,27**	**0,45**	**0,36**	**0,39**	**0,44**	**0,32**	**0,36**	**2584**
Weighted avg	**0,95**	**0,77**	**0,83**	**0,88**	**0,83**	**0,85**	**0,89**	**0,84**	**0,86**	**2584**
Samples avg	**0,75**	**0,74**	**0,74**	**0,81**	**0,81**	**0,80**	**0,82**	**0,82**	**0,81**	**2584**

Table 2: Comparative Performance Metrics of Machine Learning Models for Antibiotic Resistance Prediction

Certain classes, such as "aminocoumarin antibiotic", "polyamine antibiotic," and "isoniazid-like antibiotic," exhibit zero recall in the RF and CNN models, suggesting that the genetic markers of resistance for these classes may be complex or insufficiently represented in the training data. This highlights the need for more focused genomic research to better understand the resistance mechanisms in these bacteria. Remarkably, the SVM model achieves 100% recall for both "aminocoumarin antibiotic" and "polyamine antibiotic," which is notable given the limited data of only 39 and 12 observations respectively as shown in Appendix 1. This indicates that SVM is more adept at handling the complex genetic patterns in these classes compared to RF and CNN.

The CNN model exhibits consistently strong performance across various antibiotic classes, highlighting its capability to navigate the intricate patterns in DNA sequence data, which can differ greatly among bacterial classes.

The graph also indicates a decrease in recall for classes like "fluoroquinolone antibiotic" in all models, potentially due to inherent data characteristics that challenge modeling efforts. Additionally, "bicyclomycin-like antibiotic" shows a 0% recall, and that is because there is only one observation available as shown in table 2, emphasizing the challenges of data scarcity.

Examining the Overall Performance and Additional Metrics of the Models:

1. Random Forest:

Random Forest model has shown strong performance in certain areas but still faces challenges, particularly with handling class imbalance and sensitivity in detecting less frequent classes. The high micro-average precision of 0.98 indicates that the model is quite accurate when it predicts a class label, but a recall of 0.77 suggests it misses about 23% of true positive instances across the board. This discrepancy points to a need for improving the model's ability to detect positive instances.

Several classes exhibit zero precision, recall, and F1-score, especially those with very few examples in the training set. This issue highlights the model's difficulty in learning from underrepresented data. On the other hand, the model performs exceptionally well with classes such as "carbapenem," "cephalosporin," and "penam," where both precision and recall are high. This indicates effective learning where sufficient data is available.

However, the model's performance is inconsistent across various classes. For example, in classes like "fluoroquinolone antibiotic" and "macrolide antibiotic," the model shows high precision but significantly lower recall. This inconsistency suggests that while the model's predictions are often correct, it fails to identify many actual instances of these classes.

Overall, the model achieved a commendable micro recall score of 77%, which is quite good. It's important to highlight that this performance was achieved using DNABERT embeddings. Notably, the model performed exceptionally well in comparison to other embeddings such as one-hot encoding, which only achieved a recall of 0.69 as shown in Appendix 2. This underscores the effectiveness of DNABERT embeddings in enhancing model performance in this context.

2. SVM

The SVM model demonstrates a robust performance across various antibiotic classes, as indicated by a micro-average precision of 0.90, recall of 0.83, and an F1-score of 0.86, suggesting high overall accuracy and capability in identifying positive instances. This performance is particularly strong in well-represented classes such as "carbapenem," "cephalosporin," and "monobactam," where precision and recall exceed 0.90. However, the model struggles with classes having minimal or zero instances in the dataset, showing no predictive accuracy in classes like "aminocoumarin antibiotic" and "bicyclomycin-like antibiotic." In comparison to the Random Forest model, the SVM appears to handle well-represented classes more effectively, as reflected in its higher weighted average scores. Despite this, both models exhibit challenges with class imbalance, particularly in accurately predicting less frequent classes.

DNABERT embeddings continue to perform exceptionally well, delivering results that are remarkably better than those obtained with one-hot encoding, which achieved a recall of 0.68.

3. CNN

The performance of your Convolutional Neural Network (CNN) model showcases both strengths and areas for improvement, bearing similarities to the Random Forest (RF) and Support Vector Machine (SVM) models. CNN model achieves impressive micro-average metrics with a precision of 0.90, recall of 0.83, and an F1-score of 0.86, indicating robust overall accuracy in identifying and classifying antibiotic classes across all samples. It excels particularly with classes such as "carbapenem," "cephalosporin," and "monobactam," where both precision and recall exceed 0.90. This is similar to the performance seen in the SVM model, suggesting that the CNN is effective at handling well-represented classes.

However, like the RF and SVM, the CNN struggles with classes that have very few instances or zero support, indicating a common challenge of class imbalance affecting all three models. The CNN's weighted and samples average metrics are comparable to those of the SVM, implying that both models effectively manage class imbalance by leveraging the weight of each class according to its prevalence. Despite this, all models—including the RF—show lower macro averages, highlighting difficulties in dealing with sparsely represented classes.

In comparison, while the RF model shows slightly lower recall, suggesting it is less effective at identifying positive instances, the CNN and SVM compensate for this with better overall recall metrics. The CNN particularly demonstrates a competitive edge in handling individual sample predictions effectively, as reflected by the sample's average metrics. To further improve the CNN model, enhancing feature extraction and refining strategies to better handle underrepresented classes could be beneficial, perhaps incorporating techniques like data augmentation or advanced oversampling methods to improve learning outcomes from smaller classes. This holistic approach could help elevate the CNN's performance to even higher levels, ensuring more consistent and reliable predictions across all classes.

The potential for the CNN model to deliver even better performance is significant, especially if the right parameters can be finely tuned and the model complexity increased to enhance its learning capacity. However, a key limitation in achieving these improvements is the available compute power. Optimizing a CNN, particularly for tasks like antibiotic classification, often requires substantial computational resources to experiment with various architectures, train on larger datasets, or implement more sophisticated features. The lack of sufficient compute power restricts our ability to explore these enhancements fully, which could potentially unlock higher accuracy and more robust generalization across underrepresented classes. This underscores the need for better resources or alternative strategies that can maximize the model's performance within the available infrastructure constraints.

Our CNN model utilizing DNABERT embeddings also outperformed the one-hot encoding CNN which attained a recall of 0.77, compared to 0.83 recall achieved with the DNABERT embeddings.

B) Discussion:

The overall results from our models (CNN, RF, SVM) demonstrate robust performance for well-represented classes but also highlight the persistent challenge of class imbalance, particularly evident in the performance on sparsely represented classes. A significant enhancement in our modeling approach is the use of DNABERT-S embeddings, which have shown to considerably improve the models' ability to capture and classify complex patterns in genetic sequences over traditional methods like one-hot encoding and k-mer, particularly enhancing the performance of the CNN model.

Addressing the issue of class imbalance effectively requires a multifaceted approach. One strategy could involve data augmentation, where techniques like SMOTE or specific augmentations applicable to our data type could increase the number of samples for underrepresented classes. Another approach could be the implementation of advanced sampling techniques such as stratified sampling, ensuring that every split of the data maintains the same proportion of each class as found in the full dataset. This could be crucial for ensuring minority classes are adequately represented during training and validation.

Moreover, adopting cost-sensitive learning could adjust the learning algorithm to penalize the misclassification of minority classes more heavily, shifting the model's focus towards these classes. Ensemble methods such as bagging or boosting might also improve handling of class imbalances by combining multiple models' predictions, thus reducing both variance and bias.

Implementing rigorous cross-validation, particularly stratified k-fold cross-validation, will ensure the model's performance is reliably tested across different subsets of data, providing a more accurate measure of its ability to generalize. This method is particularly beneficial in datasets with significant class imbalance as it ensures minority classes are consistently represented in each fold.

Additionally, exploring different neural network architectures, varying layers, and depths, or experimenting with different types of layers could potentially capture complex patterns more effectively. Employing feature selection techniques to retain the most informative features might also enhance model performance. By combining these data management strategies with sophisticated modeling techniques and robust validation methods, we can significantly enhance the predictive capabilities of models in fields such as antibiotic resistance classification, ultimately leading to more reliable and actionable insights.

C) Limits

While our modeling efforts have yielded promising results, there are notable limitations impacting the scope and potential of our project that need to be addressed. These include computational power and data availability, both of which play critical roles in the development and scaling of our predictive models.

Computational Power: One of the primary constraints we face is the limited computational power available to us. Advanced models, especially those using sophisticated embeddings like DNABERT-S and deep learning architectures like CNNs, require significant computational resources for training and optimization. This limitation restricts our ability to experiment with larger, more complex models or to iterate our models as extensively as we might prefer. Enhanced computational resources would allow us to explore a broader array of modeling techniques, conduct more extensive hyperparameter tuning, and potentially improve our models' accuracy and efficiency.

Data Availability: Another significant challenge is the availability of comprehensive and diverse datasets. Currently, the amount of data we have to train and test our models, particularly data covering underrepresented classes of antibiotics, is limited. This scarcity of data can lead to models that do not perform well on rare but clinically significant antibiotic resistance patterns, affecting the generalizability and applicability of our findings. More extensive and varied datasets would enable us to train models that are not only more robust and accurate but also more representative of the real-world variability in antibiotic resistance.

Despite these limitations, we are highly confident in the potential of our approach to revolutionize research on antibiotic resistance. If provided with more comprehensive datasets and increased computational power, we believe that our models could achieve exceptional performance improvements. Such advancements could profoundly impact the field, enabling more accurate predictions of resistance patterns and facilitating the development of targeted, effective treatments. This would not only optimize therapeutic outcomes but also significantly advance the global effort to combat antibiotic resistance, marking a pivotal shift in our ability to manage and treat infectious diseases.

V. Conclusion

This thesis has explored the promising application of deep learning techniques in the detection and analysis of antibiotic resistance, a pressing challenge in global health. By leveraging advanced machine learning models and sophisticated genomic embeddings, notably DNABERT-S, we have demonstrated a potential paradigm shift in how healthcare systems can diagnose and respond to antimicrobial resistance (AMR). The use of deep learning not only accelerates the identification of resistance patterns but also enhances the precision of diagnostic outputs, crucially shortening the time to appropriate treatment intervention.

Our findings affirm that machine learning can effectively bridge the gap between extensive genomic data and actionable clinical insights, offering a robust toolset for combating AMR. The models tested, including Random Forest, Support Vector Machine, and Convolutional Neural Networks, show substantial promise in recognizing and classifying diverse bacterial strains under varied resistance profiles, particularly when empowered by species-aware embeddings. These capabilities are essential for developing targeted therapies and managing AMR more dynamically and responsively.

Despite facing challenges such as class imbalance and computational constraints, the adaptive strategies proposed—ranging from data augmentation to advanced sampling and cost-sensitive learning—provide a foundation for future research to refine and enhance predictive modeling in AMR. The continuous evolution of machine learning and its application in genomics and microbiology is poised to play a pivotal role in the global strategy against antibiotic resistance, aligning with the broader objectives of public health and safety.

In conclusion, as we advance our understanding and technology, the integration of deep learning in the fight against AMR not only represents a significant scientific advancement but also a vital public health imperative. With further development and the scaling of these technologies, we can look forward to more robust, timely, and clinically relevant interventions that will ultimately save lives and curb the spread of resistant infections worldwide.

V. References

[1] Yunxiao Ren, Trinad

Chakraborty Prediction of antimicrobial resistance based on whole-genome sequencing and machine learning, Bioinformatics, Volume 38, Issue 2, Pages 325–334, | Oxford Academic

[2] Oluwafemi A. Sarumi , Maximilian Hahn, Dominik Heider
NeuralBeds: Neural embeddings for efficient DNA data compression and optimized similarity search - ScienceDirect

[3] Gustavo Arango-Argoty, Emily Garner, Amy Pruden, Lenwood S Heath, Peter Vikesland, Liqing Zhang
DeepARG: a deep learning approach for predicting antibiotic resistance genes from metagenomic data | PubMed (nih.gov)

[4] Yanrong Ji, Zhihan Zhou, Han Liu
DNABERT: pre-trained Bidirectional Encoder Representations from Transformers model for DNA-language in genome | Oxford Academic

[5] Almeida J.S. et al. (2001)
Analysis of genomic sequences by chaos game representation. Bioinformatics, 17, 429–437.

[6] O.A. Sarumi, C.K. Leung (2022)
Adaptive machine learning algorithm and analytics of big genomic data for gene prediction, Intelligent systems reference library, intelligent systems reference library, Springer International Publishing, Cham (2022), pp. 103-123 | Google Scolar

[7] Hoang T. et al. (2016)
Numerical encoding of DNA sequences by chaos game representation with application in similarity comparison. *Genomics*, 108, 134–142 | PubMed (nih.gov)

[8] H J Jeffrey (2006)
Chaos game representation of gene structure | PubMed (nih.gov)

VII. Appendices:

Appendix 1: Frequency of Antibiotic-Resistant Bacterial DNA sequences per antibiotic.

Drug Class	**Frequency**
cephalosporin	3233
carbapenem	2415
penam	2303
monobactam	947
cephamycin	618
penem	489
aminoglycoside antibiotic	305
fluoroquinolone antibiotic	294
peptide antibiotic	246
tetracycline antibiotic	236
macrolide antibiotic	218
glycopeptide antibiotic	175
phenicol antibiotic	166
lincosamide antibiotic	138
streptogramin antibiotic	130
diaminopyrimidine antibiotic	90
glycylcycline	89
streptogramin A antibiotic	85
disinfecting agents and antiseptics	82
streptogramin B antibiotic	69
nucleoside antibiotic	69
pleuromutilin antibiotic	64
rifamycin antibiotic	58
phosphonic acid antibiotic	51
aminocoumarin antibiotic	39
nybomycin-like antibiotic	20
oxazolidinone antibiotic	18
sulfonamide antibiotic	17
nitroimidazole antibiotic	16
isoniazid-like antibiotic	15
polyamine antibiotic	12
fusidane antibiotic	11
elfamycin antibiotic	8
sulfone antibiotic	7
oxacephem	6
thioamide antibiotic	5
pyrazine antibiotic	4
mupirocin-like antibiotic	4

antibacterial free fatty acids	3
nitrofuran antibiotic	3
salicylic acid antibiotic	3
diarylquinoline antibiotic	2
orthosomycin antibiotic	1
pactamycin-like antibiotic	1
bicyclomycin-like antibiotic	1
zoliflodacin-like antibiotic	1

Appendix 2: Results achieved through one-hot encoding.

Random Forest

Drug Class	**precision**	**recall**	**f1-score**
micro avg	0,84	0,59	0,69

SVM

Drug Class	**precision**	**recall**	**f1-score**
micro avg	0,75	0,63	0,68

CNN

Drug Class	**precision**	**recall**	**f1-score**
micro avg	0,70	0,77	0,73