Using Large Language Models for Drug Discovery

Laura O'Sullivan

4th Year Project Report
Computer Science and Mathematics
School of Informatics
University of Edinburgh

2024

Abstract

Drug discovery is a time-consuming and expensive process; it can take over ten years with a cost of more than $1 billion to develop a new drug. Artificial intelligence promises to expedite this process by finding better drug candidates. This project investigates the use of Large Language Models for drug discovery. We train the ChemBERTa-2 Large Language Model [1] using feature extraction and fine-tuning for antibiotic and senolytic prediction and compare the performance to that of Smer-Barreto et al. (2023) [2] and Liu et al. (2023) [3], who trained machine learning models for senolytic and antibiotic discovery respectively. We also compare the diversity of the highest predicted compounds by ChemBERTa-2 with the machine learning models. Our results show that ChemBERTa-2 is capable of finding senolytics and antibiotics, however, it proved more successful at detecting senolytics than antibiotics. The top predicted compounds chosen by ChemBERTa-2 were largely distinct from those chosen by the machine learning classifiers and exhibited more chemical diversity.

Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Laura O'Sullivan*)

Acknowledgements

I would like to thank my supervisor, Diego Oyarzún, for all his help during the project.
I would also like to thank Vanessa Smer-Barreto for her support and helping me get a more in-depth understanding of drug discovery and senolytics.
Finally, I would like to thank all my friends and family for their support.

Table of Contents

1 Introduction **1**
- 1.1 Motivation . . . 1
- 1.2 Objectives . . . 2
- 1.3 Achievements . . . 2

2 Background **3**
- 2.1 Biology and Chemistry Concepts for Drug Discovery . . . 3
  - 2.1.1 SMILES strings . . . 3
  - 2.1.2 Senolytics . . . 5
  - 2.1.3 Antibiotics . . . 5
- 2.2 Artificial Intelligence Methods . . . 6
  - 2.2.1 Previous work . . . 6
  - 2.2.2 Machine Learning Classifiers . . . 7
  - 2.2.3 Dataset Challenges and Pre-processing . . . 7
  - 2.2.4 Large Language Models . . . 9

3 Methodology **11**
- 3.1 Datasets . . . 11
  - 3.1.1 Senolytic Dataset . . . 11
  - 3.1.2 Antibiotic Dataset . . . 11
- 3.2 ChemBERTa-2 Training . . . 12
  - 3.2.1 Feature Extraction . . . 12
  - 3.2.2 Investigation of the ChemBERTa-2 embeddings . . . 14
  - 3.2.3 Fine-tuning ChemBERTa-2 . . . 15
- 3.3 Test Set Comparison and Analysis . . . 16

4 Results **18**
- 4.1 PCA ChemBERTa-2 Training Embeddings Investigation . . . 18
  - 4.1.1 Senolytics . . . 18
  - 4.1.2 Antibiotics . . . 19
- 4.2 Feature Extraction . . . 21
  - 4.2.1 Senolytics . . . 21
  - 4.2.2 Antibiotics . . . 22
- 4.3 ChemBERTa-2 Fine-Tuning . . . 23
  - 4.3.1 Senolytics . . . 24
  - 4.3.2 Antibiotics . . . 26

4.4 Test Compounds Analysis 28
4.4.1 Senolytics 28
4.4.2 Antibiotics 33

5 Conclusions **37**
5.1 Main Findings 37
5.2 Research Question 38
5.3 Future work 38
5.3.1 Fine-Tuning Improvements 38
5.3.2 Task-Adaptive Pre-Training 39
5.3.3 Chemical Compound Class Investigation 40

Bibliography **41**

Chapter 1

Introduction

1.1 Motivation

Drug discovery is an expensive and time-consuming process. It can take over ten years with a cost of more than $1 billion to develop a new drug [4]. Furthermore, 90% of drug candidates fail clinical trials [5]. In the case of cancer drug development for example, failed drug discovery projects account for approximately 70% of R&D expenses, totalling $60 billion per year [6].

Artificial intelligence promises to accelerate the drug discovery process by providing a more cost-effective and efficient method [7]. It can even find drugs using small and imbalanced datasets, common issues confronting medical data [8, 9].

The field of natural language processing has undergone major transformation in recent years with the emergence of large language models (LLMs) [10]. An increase in computation power and dataset sizes has enabled the training of LLMs which demonstrate strong abilities across a wide range of tasks, including question answering, natural language understanding, and machine translation [10, 11, 12]. Not only that, LLMs possess capabilities in applications outside of natural language, such as computer vision [13], programming [14], and protein generation [15], as well as drug discovery. LLMs have been used to predict various properties of molecules [16, 17, 18] and have demonstrated capabilities in drug discovery related tasks [19, 20]. The potential for LLMs to be used for antibiotic and senolytic detection has yet to be investigated.

Recent studies have demonstrated great success in finding new senolytics and antibiotics using machine learning and deep learning methods [2, 3]. Machine learning models were trained on datasets containing compound properties, which were then utilised to carry out a virtual screen on a set of unlabelled compounds [2, 3]. The compounds with the highest predicted probability were tested in a laboratory[2, 3]. This resulted in the discovery of new senolytic and antibiotic compounds [2, 3].

1.2 Objectives

This dissertation builds upon previous work by fine-tuning an LLM, ChemBERTa-2 [1], to predict the likelihood of senolytic and antibiotic activity. We carry out a comparative study examining the similarities and differences of ChemBERTa-2's results to those in Smer-Barreto et al. (2023) [2] and Liu et al. (2023) [3] which used machine learning classifiers for senolytic and antibiotic prediction respectively.

This comparison involves investigating various questions:

- How does ChemBERTa-2 perform at classifying antibiotics and senolytics?
- How do the prediction probability distributions of the test compounds vary between ChemBERTa-2 and relevant machine learning model?
- What is the overlap between the top compounds predicted by ChemBERTa-2 and the respective machine learning model?

We will also investigate the diversity of the top compounds predicted by ChemBERTa-2 and compare it to that of the top compounds of the relevant machine learning model. Chemical diversity is an important consideration in developing new drugs. The senolytics that are currently known have limitations and unwanted side effects [2, 21]. Newly found antibiotics are primarily members of existing classes, meaning their efficacy is limited due to antibiotic resistance [22]. Therefore, we must develop new, diverse antibiotics in order to counteract antibiotic resistance.

1.3 Achievements

The main contributions of this dissertation are as follows:

- Investigated ChemBERTa-2's embeddings of the antibiotic and senolytic training compounds to determine the effectiveness of transfer learning to predict senolytic and antibiotic activity
- Implemented feature extraction by training classifiers on the antibiotic and senolytic embeddings from ChemBERTa-2 and assessed the performance
- Fine-tuned and evaluated ChemBERTa-2 for senolytic and antibiotic discovery
- Performed a comparative study with machine learning methods used to detect antibiotics (Liu et al. (2023) [3]) and senolytics (Smer-Barreto et al. (2023) [2]):
  - Compared the predicted probability distributions outputted by ChemBERTa-2 for antibiotic and senolytic prediction to the relevant study
  - Compiled the top test set compounds predicted by ChemBERTa-2 and investigated the overlap and predictions made compared to the lab-tested compounds from Smer-Barreto et al. (2023) [2] and Liu et al. (2023) [3]
  - Investigated the diversity of the top compounds predicted by ChemBERTa-2 and the compounds tested in the wet lab and showed ChemBERTa-2 chooses more diverse compounds

Chapter 2

Background

This chapter is a literature review of the relevant background information for this dissertation on drug discovery using large language models (LLMs). To start with, senolytics and antibiotics are outlined in conjunction with key biology and chemistry topics. Next, we summarise previous studies using machine learning techniques for antibiotic and senolytic discovery and explain the various machine learning models implemented in this project. We then discuss the data pre-processing techniques used in this dissertation and how class imbalance is addressed in classification tasks. Finally, the ChemBERTa-2 LLM is presented alongside the RoBERTa and BERT LLMs on which ChemBERTa-2 is based.

2.1 Biology and Chemistry Concepts for Drug Discovery

2.1.1 SMILES strings

The simplified molecular-input line-entry system (SMILES) is a chemical notation language developed during the 1980s by David Weiniger to represent molecular structure [23]. This paper states the following:

SMILES was developed to make molecular structure representations that both humans and computers can interpret (p.32). The molecular structure is encoded using a string of characters (p.32). Like natural language, SMILES consists of a simple vocabulary, in this case of atoms and bonds, and grammar rules for how the atoms and bonds may be combined (p.32).
Atoms are depicted using their atomic symbols contained in square brackets (p.32). Hydrogen atoms need not be included (p.32). For elements Boron (B), Carbon (C), Nitrogen (N), Oxygen (O), Phosphorus (P), Sulphur (S), Fluorine (F), Chlorine (Cl), Bromine (Br) and Iodine (I), the square brackets can be excluded provided the number of attached hydrogens has to the lowest normal valence according to its explicit bonds (p.32).
Lowercase letters rather than uppercase letters are utilised for the atomic symbols of atoms in aromatic rings (p.33) (example in Figure 2.1).
Formal charge is indicated using + or – and an optional digit can be used afterwards

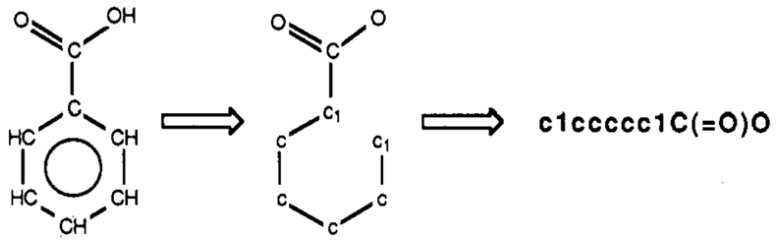

Figure 2.1: SMILES string formation for benzoic acid featuring an aromatic ring taken from [23]

(p.32). Periods denote the separation between disconnected compounds (p.32). The symbols - = # are used to encode single, double, triple and aromatic bonds respectively (p.32). Single and aromatic bonds can be left out in the SMILES string (p.32). Branches are represented using parentheses () as shown in Figure 2.2 (p.32). To encode cycling

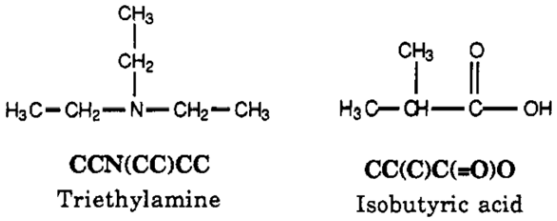

Figure 2.2: Examples of compounds with branches and their respective SMILES strings taken from [23]

structures, one of the single or aromatic bonds is severed, resulting in a connected non-cyclic graph structure (p.33). The bonds that open and close the ring are indicated in the SMILES string with the same digit after the atomic symbol. This can then be represented using the previous rules (p.33).
One of the major limitations is that SMILES strings are not necessarily unique, for example possible SMILES strings for 6-hydroxy-1,4-hexadiene can be seen in Figure 2.3 (p.32). Furthermore, the algorithm for SMILES strings is proprietary and other

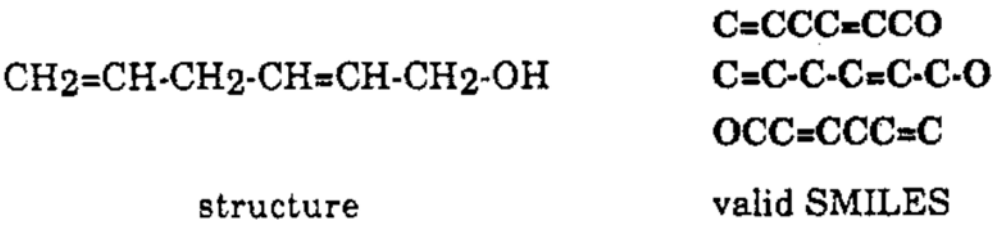

Figure 2.3: Possible valid SMILES strings for 6-hydroxy-1,4-hexadiene taken from [23]

open-source and commercial products have been developed which include their own SMILES algorithms that differ from each other [24].
In the context of machine learning, models that generate SMILES strings could generate strings that do not follow the rules of SMILES notation [25]. Models also might learn the syntax of SMILES strings instead of actually learning the characteristics of the molecules [25].

2.1.2 Senolytics

Cellular senescence is an irreversible state of cell cycle arrest where the cell no longer duplicates or divides [26, 27]. It can be caused by oxidative and genotoxic stress, oncogenic activation, mitochondrial dysfunction, exposure to radiation, or chemotherapy [28]. Cellular senescence has demonstrated positive impacts such as developing the placenta and foetus [29] and aiding wound healing [30], however senescent cells are linked to diseases such as osteoporosis, pulmonary fibrosis, hepatic steatosis, neurodegeneration, [31], SARS-CoV-2 infection [32], as well as ageing and cancer [33].

Senolytics are drugs that remove these senescent cells [31]. There is a clear need to discover new senolytics since they show potential for treating ageing-related diseases [26], yet not many have been discovered [2], and there are limitations or unwanted side effects associated with a lot of senolytics that have so far been discovered [21].

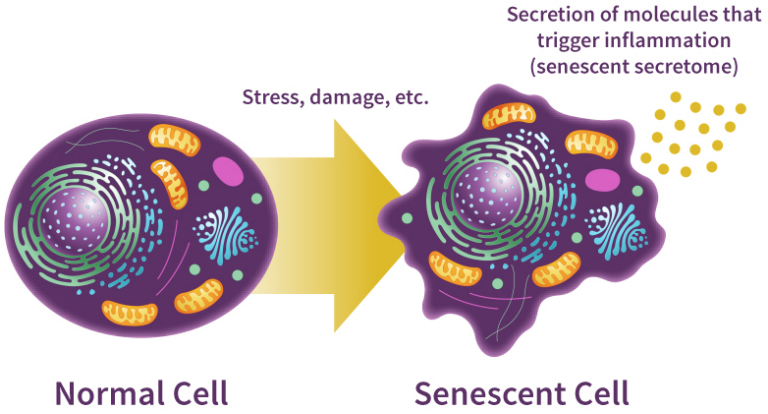

Figure 2.4: Cell undergoing senescence taken from [34]. The senescent cell no longer multiplies, it secretes chemicals that cause inflammation, which can damage nearby cells [34]

2.1.3 Antibiotics

Antibiotics are medications used to treat infections caused by bacteria. They kill or inhibit the growth of the bacteria [35]. This dissertation focuses on antibiotics that target acinetobacter baumannii, a Gram-negative bacterium which is pathogenic to humans [36]. Infections caused by acinetobacter baumannii include skin and soft tissue infections, pneumonia, urinary tract infections, bacteremia, and meningitis [37].

Acinetobacter baumannii infections are commonly found in hospitals, particularly in Intensive Care Unit patients [38], however, it posed major issues in wounded soldiers during the Iraq war, which contributed to the spread of acinetobacter baumannii in hospitals across the world [39, 36].

Antimicrobial resistance, where bacteria, viruses, parasites and fungi no longer respond to treatment, is proving to be one of the major healthcare challenges of the 21st century [40, 41]. In 2019, 4.95 million deaths were associated with drug-resistant infections, with 1.95 million of these being attributed directly to their drug-resistant nature [40]. Acinetobacter baumannii is a known multi-drug resistant bacteria and is one of the

six main bacteria contributing to deaths related to antibiotic resistance [36, 40]. It is estimated that acinetobacter baumannii infections alone were associated with 452,000 deaths worldwide in 2019 [42]. The mortality rate estimates of patients with infections from acinetobacter baumannii range from 26.0% to 55.7%, with an estimated range of 8.4% to 36.5% of patients dying directly as a result of the infection [43].

Factors such as overuse and inappropriate prescribing of antibiotics, agricultural use and low numbers of new antibiotics being developed have contributed to antibiotic resistance [44]. It has been estimated that up to 10 million deaths annually could occur due to antimicrobial resistance by the year 2050 if new treatments are not found to tackle antimicrobial resistance [41]. Therefore, it is vital we find new antibiotics so that we can overcome antibiotic resistance.

2.1.3.1 Tanimoto Similarity

Tanimoto (or Jaccard) similarity is a method used to calculate the similarity between molecules [45]. The compounds are represented numerically using a molecular ”fingerprint” i.e. a series of compound values [45]. In this dissertation, 200 physicochemical features calculated from RDKit are utilised as fingerprints [46]. The similarity between two molecular fingerprints, A and B, is calculated as follows [47]:

$$Tanimoto\ Similarity = 1 - Tanimoto\ Distance$$

where

$$Tanimoto\ Distance = \frac{A \cdot B}{\|A\|^2 + \|B\|^2 - A \cdot B}$$
$$A \cdot B = \sum_i A_i B_i$$
$$\|A\|^2 = \sum_i A_i^2$$

Tanimoto similarity values range from 0 to 1, where a similarity score of 0 means there are no elements in common, and a score of 1 indicates identical fingerprints [48].

2.2 Artificial Intelligence Methods

2.2.1 Previous work

2.2.1.1 Machine Learning for Senolytic and Antibiotic Discovery

Previous studies have used machine learning [2] and graph neural network [21] techniques to discover new senolytics. Machine learning classifiers have been widely used for antibiotic discovery [49]. This dissertation focuses in particular on antibiotics targeting acinetobacter baumannii, which have previously been classified using an ensemble model of directed message-passing neural networks [3]. The potential for LLMs to find senolytics and antibiotics has yet to be investigated. LLMs have demonstrated abilities to determine various properties of molecules [16, 17, 18] as well as in drug discovery related tasks [19, 20].

2.2.2 Machine Learning Classifiers

2.2.2.1 Random Forest

Random forest is an ensemble model that votes across the outputs of the individual decision trees [50]. Each decision tree is trained on a different bootstrap sample of the dataset and a subset of features is used to split the nodes of the tree [50]. This is so that the errors made by trees are independent of each other [51]. Random forest can demonstrate the importance of features for classification and is less computationally expensive than other ensemble models using trees such as boosting methods [52]. The performance of random forest can be poor for imbalanced datasets [53].

2.2.2.2 XGBoost

EXtreme Gradient Boosting (XGBoost) is another ensemble model that utilises decision trees [54, 55]. XGBoost involves using a technique called gradient boosting to train a series of decision trees known as "weak learners" [54, 56, 57]. These "weak learners" predict the residuals of prior models which are concatenated to make a final prediction [54, 56]. XGBoost is efficient and allows for parallel computing however it demonstrates poorer performance when applied to imbalanced datasets [58].

2.2.2.3 Support Vector Machine

Support vector machine (SVM) is a linear machine learning classifier that determines a hyperplane separating two classes by maximising the distance between it and the closest training examples of each class, which are known as the support vectors [59]. SVM can be used for non-linear classification by first transforming the data into a high dimensional space in which the classes can be separated linearly, for example, the Radial Basis Function kernel [60].

2.2.2.4 K-Nearest Neighbours

The K-nearest neighbours (KNN) classifier is a non-parametric model where the k nearest points in the training set are obtained and their majority-class is outputted [61]. KNN generally has good performance and is interpretable [62]. However, it is sensitive to the number of nearest neighbours considered, is quite slow and requires a large amount of memory since the entire training dataset must be stored (hence it is non-parametric) and the distance to each training example must be calculated for predicting a new example [62].

2.2.3 Dataset Challenges and Pre-processing

2.2.3.1 Imbalanced Datasets

Imbalanced datasets pose challenges to the performance of many standard machine learning classifiers [63]. Accuracy alone is not a suitable metric for evaluating the performance of models since an algorithm can get high accuracy scores by simply predicting the value of the majority class for every test example [63]. Metrics that are

commonly used to evaluate models on imbalanced datasets include [63]:

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

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

$$F1\ score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Precision is a measure of how many of the examples labelled as positive are actually positive, whereas recall calculates how many of the positive class examples were labelled correctly [63]. F1 score is the harmonic mean of these two metrics [63]. These values can be calculated from the confusion matrix which outlines how many examples were predicted per class according to their actual label as shown in Table 2.1.

		Predicted	
		0	1
True	0	# True Negatives	# False Positives
	1	# False Negatives	# True Positives

Table 2.1: Confusion matrix showing the predictions outputted compared to their actual classes adapted from [63]

Precision recall curves can also give an insight into the performance of a model on imbalanced data [63]. The corresponding recall value is plotted for different precision values as you the threshold for what a model considers positive is changed [64]. The area under this curve provides a metric for evaluating model performance [64].

2.2.3.2 Dimensionality reduction: Principal Component Analysis

Datasets with a large number of features can pose problems for machine learning models since having more features increases the risk of overfitting [65]. Additionally, not all features may be relevant for prediction [66]. Dimensionality reduction techniques are commonly implemented to overcome this challenge.
Principal component analysis (PCA) is a dimensionality reduction method that aims to preserve most of the variation of the dataset [67]. PCA is also useful for visualising data [68]. PCA uses linear combinations of the original variables of the dataset called principal components as the new features [68].

The dataset must be standardised first such that the features have a mean value of 0 and a standard deviation of 1 [68].

The principal components can then be found using Singular Value Decomposition [68]:

$$X = P\Sigma Q^T$$

where X is the $n \times d$ standardised dataset, the columns of $n \times n$ matrix P are the orthonormalised eigenvectors of XX^T, known as the left singular vectors, the columns of

$d \times d$ matrix Q are the orthonormalised eigenvectors of $X^T X$, known as the right singular vectors, and Σ is an $m \times n$ diagonal matrix with the square root of the eigenvalues of X ordered from largest to smallest on the diagonal [69]. The principal components correspond to the columns of Q where the first principal component is the direction in which the dataset that has maximum variance [68].
The new dataset with reduced dimensions can be calculated as follows:

$$Z = XQ_k$$

where Z is the new $n \times k$ dataset with the reduced dimensions and Q_k is the $d \times k$ matrix with the first k columns (principal components) of Q [68].

PCA is sensitive to outliers [70] and assumes a linear relationship between the features of the input dataset [71].

2.2.3.3 Uniform Manifold Approximation and Projection (UMAP)

UMAP is another dimensionality reduction method which utilises manifold learning techniques [72]. Firstly, UMAP creates a high-dimensional graph of the data. It assumes that the data lies on a manifold, which is approximated and used to form the high-dimensional representation of the data, known as a ”fuzzy topological” representation [72]. From this, UMAP optimises a low-dimensional representation of the high-dimensional representation by minimising the cross-entropy between the high- and low-dimensional representations [72]. In comparison to PCA, it is less interpretable and the UMAP dimensions do not have any particular meaning, however, it is non-linear and it is faster and scales better for larger datasets than t-SNE, another state-of-the-art dimensionality-reduction technique that uses manifold learning [72].

2.2.3.4 Transfer Learning

Transfer learning is a machine learning technique where a pre-trained model is applied to a new but related task [73]. Transfer learning is commonly implemented when limited data is available as the quality of the latent feature representation is quite poor [73]. The model is pre-trained on a large, general dataset and this learned knowledge can be ”transferred” to a previously unseen problem [73].
In the context of LLMs, two approaches are normally used. In one method, known as feature extraction, the output from the final layer of the LLM before the classifier is taken and inputted into a separately trained classifier [73]. This does not involve retraining the parameters of the LLM [73]. Alternatively, the LLM can be fine-tuned by changing all of the parameters or only some by freezing layers of the LLM [73]. Lower learning rates are more suitable when fine-tuning LLMs in order to capitalise on the pre-trained parameters [73].

2.2.4 Large Language Models

2.2.4.1 Bidirectional Encoder Representations from Transformers (BERT)

Bidirectional Encoder Representations from Transformers (BERT) is an LLM developed by researchers at Google [11]. BERT’s model architecture is a multi-layer bidirectional

transformer with 12 transformer blocks, hidden layer size of 768 and 12 self-attention heads, giving a total of 110 million parameters [11]. The model was pre-trained on BooksCorpus [74] and English Wikipedia [11]. The inputs to BERT are tokenised and embedded contextually [11]. One significant advantage of BERT is its bidirectional nature where it can look both left and right in a sentence allowing access to the preceding and succeeding tokens [11]. This was trained using masked language modelling where 15% of the input tokens were masked at random and the model must predict the masked tokens [11]. BERT is also trained for next sentence prediction by inputting two sentences A and B and predicting whether B followed A in their original dataset [11].

BERT can be fine-tuned for a wide range of tasks [11]. State-of-the-art results were achieved on the GLUE [75], SQuAD [76, 77] and SWAG [78] benchmarks [11]. The BERT model proved to be revolutionary in the natural language processing field [79].

2.2.4.2 A Robustly Optimized BERT Pretraining Approach (RoBERTa)

Researchers from Facebook AI and the University of Washington improved on BERT with their Robustly optimized BERT approach (RoBERTa) model [80]. RoBERTa has the same architecture as BERT however it was trained on more data for longer with larger batch sizes and sentence lengths [80]. Unlike BERT, RoBERTa was only trained for masked language modelling (not for next sentence prediction) [80]. RoBERTa also implements dynamic masking of tokens where instead of having a static rate of masking tokens, a new pattern of masking is created each time a sequence is inputted into the model [80]. RoBERTa demonstrated cutting-edge results comparable to or better than BERT's for the GLUE [75], RACE [81] and SQuAD [76, 77] benchmarks.

2.2.4.3 ChemBERTa-2

ChemBERTa-2 further trains RoBERTa to predict molecular properties from SMILES strings [1]. This is the second ChemBERTa model; it improves on the first by optimising the pre-training process and training on a larger number of compounds [1]. A dataset of 77 million SMILES strings from PubChem, an open-source database of chemical structures [82], was used to train ChemBERTa-2 [1]. Two training methods were implemented - masked language modelling where the model must predict the masked tokens in the input, and multi-task regression, where 200 molecular properties are predicted simultaneously for each compound [1]. Hyperparameter optimisation is carried out on a dataset of 5 million SMILES strings to ensure convergence [1]. ChemBERTa-2 is fine-tuned and evaluated on regression and classification tasks for molecular properties from the MoleculeNet benchmark [83], for which it achieved competitive results [1].

Chapter 3

Methodology

This chapter outlines the datasets used to train ChemBERTa-2 for antibiotic and senolytic prediction, the methods utilised for ChemBERTa-2 training, and how the results were compared to the machine learning classifiers in Smer-Barreto et al. (2023) [2] and Liu et al. (2023) [3]. First, we give an overview of the datasets of senolytic and antibiotic compounds and how they were collated. Secondly, the two methods used to train the ChemBERTa-2 model to classify senolytics and antibiotics are introduced and we show the analyses of their performance. To conclude the chapter, we explain how the test set results for ChemBERTa-2 are compared with those of the machine learning classifiers in Smer-Barreto et al. (2023) [2] and Liu et al. (2023) [3] on probability predictions, performance and compound diversity.

3.1 Datasets

3.1.1 Senolytic Dataset

The dataset for senolytic prediction was compiled by Smer-Barreto et al. (2023) [2] for the purpose of training machine learning classifiers for senolytic prediction. The training dataset consists of 2,523 training compounds and is significantly imbalanced, with only 58 positive senolytics (2.30%). This imbalance was intentionally chosen by the authors since it illustrates the reality that a compound is very unlikely to be a senolytic [2]. The compounds are represented using their SMILES string [23]. The study collected a list of 4,340 previously unlabelled compounds, and using the best-performing machine learning classifier on the training dataset XGBoost, made predictions for the likelihood of senolytic activity of these compounds. Laboratory experiments were conducted on the top 21 compounds with the highest predicted probability by XGBoost to determine senolytic activity, resulting in 3 positive and 18 negative compounds.

3.1.2 Antibiotic Dataset

Data on antibiotics were collated in Liu et al. (2023) [3]. The target in question for the antibiotics is acinetobacter baumannii, one of the six main bacteria contributing to worldwide deaths related to antibiotic resistance [40]. The dataset consists of 7,684

training compounds represented by their SMILES string [23], 480 of which are positive (6.38%). 6,690 test compounds were screened using an ensemble of ten deep learning models using Chemprop, a package that provides directed message-passing neural networks for chemical property prediction [3, 84]. The best 240 compounds were chosen based on average prediction probability and dissimilarity to the positive training compounds and tested in a laboratory for antibiotic activity [3]. 40 of these compounds proved to be true positives.

3.2 ChemBERTa-2 Training

The ChemBERTa-2 model is provided in two formats - one version trained for masked language modelling and the other to simultaneously predict 200 physicochemical properties of each molecule. Since drug discovery is a classification task, the latter model was chosen for the purpose of this project.

Transfer learning was implemented to classify senolytic and antibiotic compounds using the pre-trained ChemBERTa-2 model. Two approaches were used for this: feature extraction and fine-tuning. Feature extraction involved extracting the embeddings of the dataset from the final layer of the pre-trained ChemBERTa-2 model before the classifier and training a separate classifier on these embeddings. For fine-tuning, the weights of the ChemBERTa-2 model were retrained on the senolytic and antibiotic training sets.

3.2.1 Feature Extraction

The dataset was tokenised using the ChemBERTa-2 tokeniser and is fed through the pre-trained ChemBERTa-2 model using Hugging Face [85]. ChemBERTa-2 was originally trained using the RoBERTa model [80] on chemical compounds so they share the same model architecture: 2 transformer blocks with hidden layer size of 768 and 12 self-attention heads, totalling 110 million parameters [80, 11]. The output from the last layer prior to the classification layer was obtained. This output is known as the embeddings of the dataset. These embeddings were then used to train machine learning classifiers with scikit-learn [86] for antibiotic and senolytic activity.

The embeddings of the ChemBERTa-2 model are 384 dimensions, so PCA was applied to the embeddings to reduce their dimensionality to be used for classification. 52 components were retained for the senolytics dataset and 62 components for the antibiotics dataset so that 90% of the variance was retained. The scree plots for the senolytic and antibiotic embeddings can be seen in Figures 3.1 and Figure 3.2.

Various classifiers were trained on the output from PCA applied to the embeddings: Support Vector Machine (SVM), Random forest, XGBoost and K-Nearest Neighbours (KNN) (due to computational constraints Support Vector Machine (SVM) was not trained on the antibiotic embeddings). Since the datasets are very imbalanced, the classifiers which allow for class weights, SVM and Random Forest, used balanced class weights which equally weighted positive and negative examples during classification.

Hyperparameter optimisation was carried out using a random search of the hyperparameter space for 30 trials. Initially, it was planned to use Bayesian hyperparameter

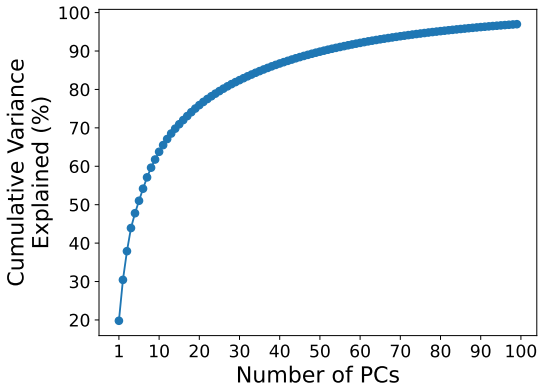

Figure 3.1: PCA cumulative scree plot for ChemBERTa-2 embeddings of senolytics dataset showing how much of the variance is explained cumulatively for each principal component (PC) up to 100 PCs

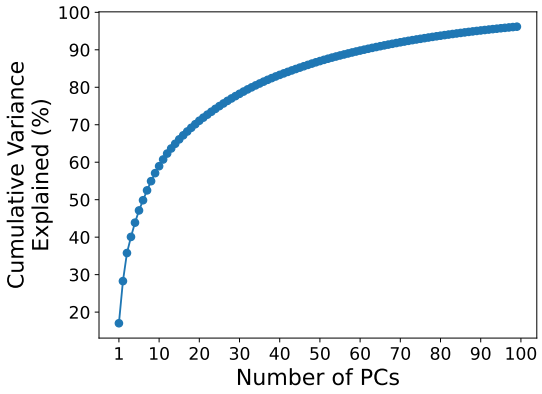

Figure 3.2: PCA cumulative scree plot for ChemBERTa-2 embeddings of antibiotics dataset depicting how much of the variance is explained cumulatively for each principal component (PC) up to 100 PCs

optimisation for this, however the best hyperparameters found performed extremely poorly on the senolytics dataset, and random search found good hyperparameters.

Hyperparameter search spaces for each classifier:
Random forest

- No. of estimators : 1, 10, 50, 100, 1000
- Criterion: gini, entropy, log_loss
- Maximum tree depth: None, 1, 10, 100
- Max features: None, log2, sqrt
- Minimum number of samples for splitting an internal node: 2, 5, 10
- Minimum number of samples for a leaf node: 1, 5, 10

KNN

- No. of Neighbours: 2, 5, 10
- Algorithm to compute nearest neighbours: auto, ball_tree, kd_tree, brute
- Weight function: uniform, distance

SVM

- Regularisation parameter C: 0.001, 0.01, 0.1, 1, 10, 100
- γ, inverse of the radius of influence of support vectors: 0.001, 0.01, 0.1, 1, 10, 100
- Kernel type: linear, poly, rbf

XGBoost

- Learning rate: 0.05, 0.10, 0.3, 0.5, 0.75, 1
- γ, minimum loss reduction a further partition can be made on a leaf: 0.001, 0.01, 0.1, 1, 10, 100
- Max tree depth: 3, 5, 10, 20
- Minimum sum of instance weight required for a child: 1, 3, 8, 15
- Subsample ratio of columns during tree construction: 0.1, 0.5, 0.9

Hyperparameter optimisation was carried out on the entire training dataset. Normally, the training set is split into a training and validation set for hyperparameter optimisation, however since the datasets are small with few positive samples, the senolytics dataset especially, the entire antibiotic and senolytic training sets were used to fully leverage the positive samples available. Given the class imbalance, the best hyperparameter configuration was chosen based on the mean F1 score across 4 stratified cross-validation folds. The standard deviation between folds was calculated to evaluate the reliability of the mean F1 score as an indication of the true F1 score.

The model with the best F1 score during hyperparameter optimisation was trained and evaluated on a training and validation set which was obtained by splitting the training set into 70% training samples and 30% validation samples whilst ensuring the split was stratified. We evaluated the model based on validation accuracy, precision, recall, and F1 score. The confusion matrices and precision-recall curve alongside its area under the curve were also graphed and analysed.

3.2.2 Investigation of the ChemBERTa-2 embeddings

Feature extraction takes advantage of the learned representation of the dataset by the model and allows us to train our own classifier on the dataset. However, the training of ChemBERTa-2 must have contained a substantial amount of these training samples for the ChemBERTa-2 learned knowledge to be transferred to these drug discovery tasks and the embeddings to be a good representation of the training set. We investigated the PCA and UMAP plots for two components of the training sets embeddings to determine the effectiveness of the embeddings at representing the compounds and to see what information they captured.

3.2.3 Fine-tuning ChemBERTa-2

Fine-tuning a large language model involves retraining some or all of the weights on a new task [73]. We chose to fully train all the weights. To get good results when only retraining some weights, a sufficient number of training compounds for the antibiotic and senolytics dataset must have appeared in the ChemBERTa-2 training dataset [73]. Unfortunately, the full training set of 77 million training compounds for ChemBERTa-2 is not available, however, a subset of the training set containing 10 million molecules which was used to train the original ChemBERTa model [82] was previously released. We investigated the compounds in this subset and found that it contained very few of the antibiotic and senolytic compounds from our datasets, especially positive samples - no positive training senolytic examples and only 4 positive training antibiotics were in this set. As a result, we decided it was best to fine-tune all parameters of the ChemBERTa-2 model as the embedding space of the ChemBERTa-2 model may not be sufficiently expressive and representative of the antibiotic and senolytic training compounds and transfer of knowledge may be quite poor.

Optuna [87] was used to carry out hyperparameter tuning, which uses Bayesian optimisation to select hyperparameter configurations. I adapted code used by Vanessa Smer-Barreto for hyperparameter optimisation on a separate task using machine learning classifiers. 50 trials were carried out for the antibiotic dataset and 80 for the senolytic dataset, the antibiotic dataset used fewer trials due to RAM limitations. Like with the feature extraction classifiers, hyperparameter optimisation was carried out on the entire training set to take advantage of all the positive training compounds. The best hyperparameters selected were used to train a model and validated on a training set and validation set obtained from a 70:30 stratified split of the full training set.

The dataset was tokenised and inputted into ChemBERTa-2 using the same ChemBERTa-2 tokeniser as used for feature extraction. A custom cross-entropy loss function is implemented using PyTorch that takes the balanced class weights into account [88]. I followed the Hugging Face tutorials as outlined in the Hugging Face documentation [89, 90].

Hyperparameter Search Space for ChemBERTa-2:

- Learning rate: Log uniform distribution from 10^{-6} to 10^{-3}
- Weight decay (regularisation): Log uniform distribution from 4×10^{-5} to 10^{-1}
- Number of epochs: 1-4

The number of epochs was kept low as earlier experiments with a higher upper limit for the number of epochs indicated there was overfitting occurring after only 2 or 3 epochs. Despite this, the best hyperparameter configuration found by Optuna would choose a high number of epochs on the upper end of the range, and the best models found were overfitting. Upon further investigation, it was revealed that the F1 scores do not change much as the number of epochs increases despite the model clearly overfitting based on the training-validation loss curves. The best hyperparameters chosen based on optimising validation loss proved to have poor F1 and precision metrics, so we decided to limit the search space to simultaneously counteract overfitting and maintain high F1

values.

The best hyperparameter configuration was chosen based on the mean F1 score across 4 stratified cross-validation folds. The standard deviation between folds was calculated to examine the reliability of the mean F1 score to indicate the true F1 score. We trained the best-obtained hyperparameters on a stratified 70% 30% training validation split of the training set.

The Hugging Face model has an argument `load_best_model_at_end` which uses the best model seen across epochs based on a chosen metric rather than loading the final epoch necessarily. This argument was utilised to choose the best model based on the F1 score for the senolytics dataset and precision for the antibiotics dataset. Note this feature was not used for hyperparameter optimisation as it would give us a false result for the total number of epochs.

3.3 Test Set Comparison and Analysis

We carried out a comparative study on the predictions given by ChemBERTa-2 and the machine learning methods from Smer-Barreto et al. (2023) [2] for senolytic prediction and Liu et al. (2023) [3] for antibiotic prediction.

As outlined above, it is possible that ChemBERTa-2's training set does not contain many of the senolytic or antibiotic training compounds. For this reason, we decided to only carry out our test set analysis on the fine-tuned ChemBERTa-2 model since the embeddings obtained during feature extraction may not be a good representation of the training compounds. This also gives a better comparison between ChemBERTa-2 and machine learning methods since feature extraction involved using machine learning classifiers.

Firstly, we trained the best hyperparameters found during hyperparameter optimisation for antibiotic and senolytic prediction on the entire training set and used the obtained models to make predictions on the test set compounds. The probability scores and predictions given by the fine-tuned ChemBERTa-2 model for the senolytic and antibiotic test datasets were graphed to investigate their distribution and are compared with their respective machine learning model results from Smer-Barreto et al. (2023) [2] and Liu et al. (2023) [3] .

Since we do not have the correct labels for all of the test compounds, we focused our analysis on the wet lab compounds results i.e. the compounds that were tested in a laboratory for senolytic or antibiotic activity by Smer-Barreto et al. (2023) [2] and Liu et al. (2023) [3]. 240 compounds were tested for antibiotic activity from the test set, of which 40 were positive [3], and 21 compounds for the senolytic test dataset, with 3 compounds proving to be positive senolytics [2]. We simulated the process with ChemBERTa-2 by obtaining the 21 senolytic test compounds with the highest predicted probability and the top 240 antibiotic test compounds by probability prediction. Liu et al. (2023) [3] chose their wet lab compounds slightly differently by also taking compound similarity with respect to the positive training examples into account (they favoured dissimilar samples to the positive training examples for

drug diversity purposes) [3], however for simplicity we chose to just take the top 240 predicted compounds by ChemBERTa-2. Our comparison of the overlap between the top 240 compounds predicted by the ensemble model in Liu et al. (2023) [3] and the actual wet lab compounds showed that the compounds not tested in the lab all had a probability prediction of less than 37.2%. Since we were unable to test ChemBERTa-2's top compounds in a lab, we examined the overlap between them and their respective machine learning method comparison wet lab compounds. This also gives us an insight into how different the compounds chosen by ChemBERTa-2 are.

To investigate the diversity of the compounds, we looked at how similar the ChemBERTa-2 top compounds and the machine learning classifier wet lab compounds are to the positive training examples. The Tanimoto similarities between the positive training compounds and the wet lab compounds as well as between the top ChemBERTa-2 compounds were calculated. We analysed ridge plots of the Tanimoto similarity with the positive training examples to compare the positive training examples' similarity to the top test set compounds as predicted by ChemBERTa-2 and the machine learning classifier. The colour of the ridge plots is calculated using the min-max normalised weighted mean of every density. The minimum and maximum x-values from each density are used as the normalisation minimum and maximum values, respectively [91].

PCA was also applied to the datasets to investigate how much variance is explained by the PCA principal components, with the idea that if the components cumulatively explain less of the variance and the cumulative plot grows slower, the compounds are more diverse.

We also combined the positive training examples, the top compounds from ChemBERTa-2 and the machine learning method wet lab compounds together and applied PCA to investigate the distribution of compounds chosen by each method in the PCA space and how they compare to the positive training examples. Compounds that are considered diverse with respect to the positive training examples are expected to be located in different regions of the PCA space.

Chapter 4

Results

We present the results of using ChemBERTa-2 for senolytic and antibiotic prediction and the comparative study with Smer-Barreto et al. (2023) [2] and Liu et al. (2023) [3].

First, we evaluate how well the ChemBERTa-2 embeddings represent the training compounds. After that, we present the hyperparameter optimisation results for ChemBERTa-2 fine-tuning and feature extraction and evaluate the performance of the best models. Next, we perform an analysis on the test set compound results for ChemBERTa-2 and benchmark the results against those from Smer-Barreto et al. (2023) [2] and Liu et al. (2023) [3] to determine how well ChemBERTa-2 classifies compounds and how chemically diverse the compounds it chooses are.

4.1 PCA ChemBERTa-2 Training Embeddings Investigation

We investigate the distribution of positive and negative compounds in the PCA space to investigate the abilities of the ChemBERTa-2 embeddings to represent the compounds.

4.1.1 Senolytics

We plot the first two principal components for the training embeddings in Figure 4.1. We also label the positive senolytics by compound class where known with the non-senolytic compounds being labelled as ”Negative”. We observe that the senolytics are dispersed throughout the PCA space with senolytics of the same compound class being close to each other.

The plot of the first two UMAP components in Figure 4.2 also exhibits a similar trend where the senolytics are spread throughout the PCA space and senolytic classes are grouped together, albeit with a few outliers

These results are expected and demonstrate that ChemBERTa-2’s embeddings are representing the training compounds well. Smer-Barreto et al. (2023) [2] shows that the senolytics chosen are diverse. The fact that the compound families are close to one

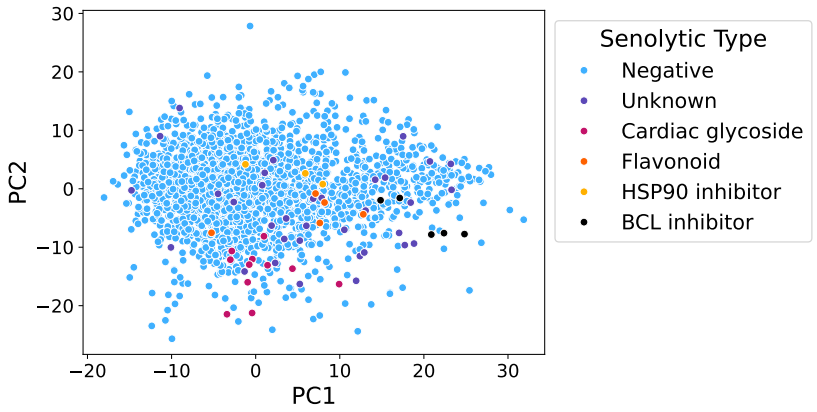

Figure 4.1: Plot of first 2 principal components (PCs) of PCA for ChemBERTa-2 embeddings of senolytics dataset with senolytic type labelled

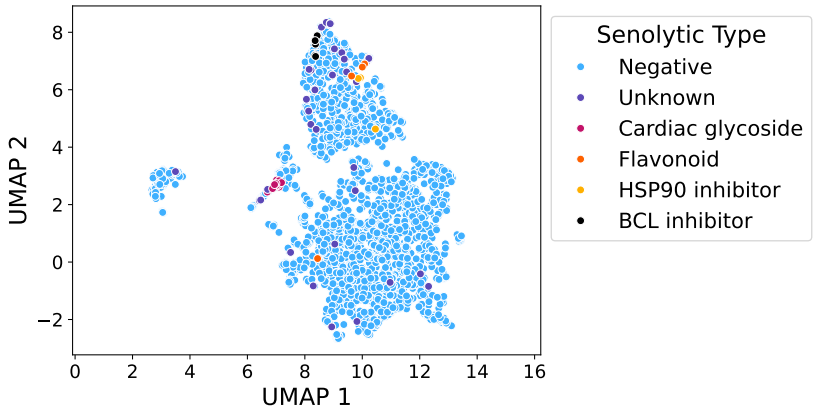

Figure 4.2: Plot of first 2 UMAP components for ChemBERTa-2 embeddings of senolytics dataset with senolytic type labelled

another indicates a good representation of the compounds since compounds of the same family are more similar to each other and share characteristics.

4.1.2 Antibiotics

Unfortunately, the antibiotic compound classes were not readily available so we plot the first two principal components and label the training compounds simply as positive or negative for antibiotic activity in Figure 4.3. The compounds are spread throughout the PCA space including the positive compounds as shown in Figure 4.3.

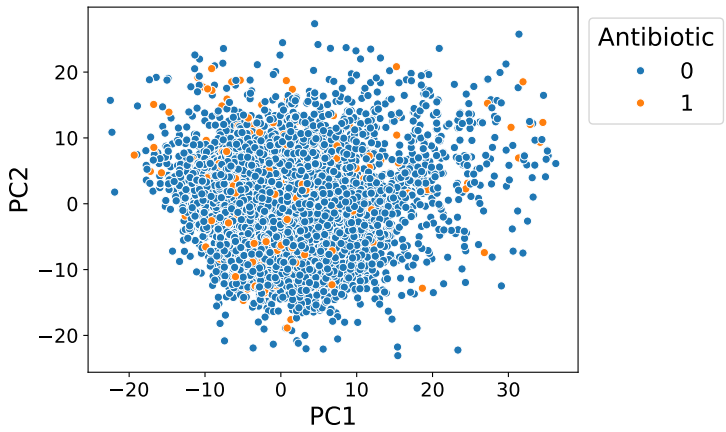

Figure 4.3: Plot of first 2 principal components (PCs) of PCA for ChemBERTa-2 embeddings of antibiotic training dataset with positive compounds and negative labelled

The positive compounds are not as spread out however in the plot of the first two UMAP components (Figure 4.4). All but seven are located within one central group, with the seven outliers scattered outside the group. Most negative compounds are also found in this central group with some scattered around the periphery.

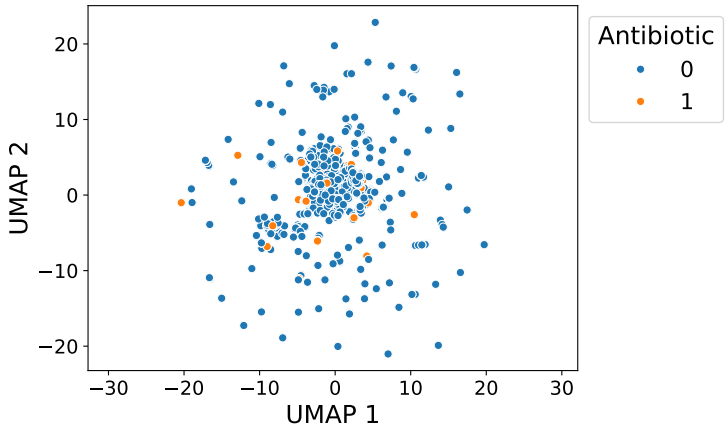

Figure 4.4: Plot of first 2 UMAP components for ChemBERTa-2 embeddings of antibiotic training dataset with positive compounds and negative labelled

The ChemBERTa-2 embeddings may be a poorer representation of the antibiotic training compounds than for the senolytic training examples. Liu et al. (2023) state the antibiotics and negative compounds are structurally diverse [3], however the UMAP plot of the ChemBERTa-2 embeddings demonstrates a clear group in the centre with a few outliers outside, which indicates that ChemBERTa-2's embeddings may not be able to capture the compound diversity very well.

4.2 Feature Extraction

We train classifiers on the extracted embeddings from the ChemBERTa-2 model and evaluate their performance.

4.2.1 Senolytics

Table 4.1 shows the best mean F1 score and its standard deviation across the four stratified cross-validation folds of the best hyperparameters found during 30 hyperparameter optimisation trials for each classifier.

Model	Mean F1 Score	Standard Deviation
KNN	0.6623	0.0459
XGBoost	0.6540	0.0491
SVM	0.6434	0.0719
Random Forest	0.6244	0.0426

Table 4.1: Mean F1 score and standard deviation for the cross-validation folds of each classifier on the senolytics ChemBERTa-2 training embeddings

The mean F1 scores are all quite high and close with a range of 0.0379. The standard deviations of the cross-validation fold F1 values are low.

KNN was chosen as the best model as it had the highest mean F1 score. It also had the second-lowest standard deviation between folds. The hyperparameter configuration corresponding to the best mean F1 score from above is as follows:

- No. of Neighbours: 2
- Algorithm to compute nearest neighbours: auto
- Weight function: distance

The full results of the KNN model trained on a 70%-30% stratified training validation split of the training set are shown in Table 4.2. The final F1 score is almost 0.2 lower than the mean F1 of the cross-validation folds for KNN in Table 4.1. The corresponding confusion matrix and precision-recall curve are displayed in Table 4.3 and Figure 4.5 respectively. The area under the precision-recall curve is 0.32 as displayed in Figure 4.5.

Validation Accuracy	Validation Recall	Validation Precision	Validation F1
0.9762	0.4118	0.4667	0.4375

Table 4.2: Validation metrics for best classifier of ChemBERTa-2 senolytics embeddings, KNN

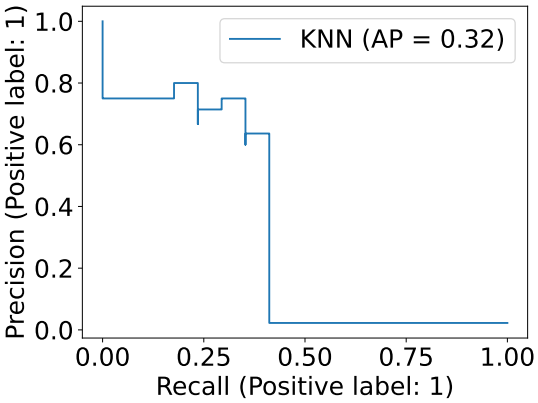

Figure 4.5: Precision-recall curve for KNN, the best ChemBERTa-2 senolytics embeddings classifier

		Predicted	
		0	1
True	0	732	8
	1	10	7

Table 4.3: Confusion matrix for best classifier KNN on ChemBERTa-2 embeddings of senolytics training set

4.2.2 Antibiotics

The best mean F1 score and its standard deviation across the four stratified cross-validation folds during hyperparameter optimisation are displayed in Table 4.4.

Model	Mean F1 Score	Standard Deviation
KNN	0.3102	0.0432
XGBoost	0.2809	0.0103
Random Forest	0.2841	0.0384

Table 4.4: Mean F1 Score and Standard Deviation for the cross-validation folds of each classifier on the ChemBERTa-2 antibiotics training embeddings

The mean F1 scores are quite low, particularly compared to those of the senolytics dataset. This is a further indication that the ChemBERTa-2 embeddings may not be sufficient for representing the antibiotic compounds, resulting in poor transfer of knowledge from ChemBERTa-2's original training.

The best model based on the mean F1 score, KNN, was trained on a 70% 30% stratified training validation split of the training set.

The best hyperparameters for KNN were:

- No. of Neighbours: 2
- Algorithm to compute nearest neighbours: ball tree
- Weight function: distance

The validation metrics can be seen in Table 4.5 and the respective confusion matrix in Table 4.6. Figure 4.6 shows the precision-recall curve for this classifier. There was a slight increase in the F1 score compared to the mean F1 score across folds, however it is still lower than KNN's F1 score for the senolytics feature extraction. The area under the precision-recall curve is lower for the antibiotics than for the senolytics at 0.25.

Validation Accuracy	**Validation Recall**	**Validation Precision**	**Validation F1**
0.9202	0.3333	0.3529	0.3429

Table 4.5: Validation metrics for best classifier of ChemBERTa-2 antibiotics embeddings

		Predicted	
		0	1
True	0	2074	85
	1	96	48

Table 4.6: Confusion matrix of best classifier for ChemBERTa-2 antibiotics training embeddings, KNN

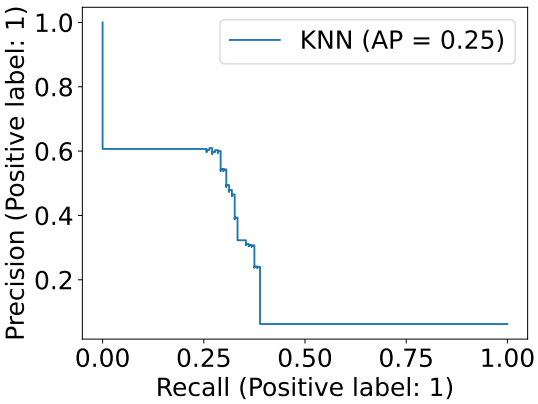

Figure 4.6: Precision-recall curve for KNN, the best ChemBERTa-2 antibiotics embeddings classifier

4.3 ChemBERTa-2 Fine-Tuning

We next carry out full fine-tuning of ChemBERTa-2 for antibiotic and senolytic prediction and outline the results.

4.3.1 Senolytics

Hyperparameter optimisation with 80 trials was carried out using Optuna and the best hyperparameters and their respective mean F1 score with standard deviation across the four stratified cross-validation folds are outlined in Table 4.7. The mean F1 score across folds is 0.1818 lower than the mean F1 score for the best classifier trained on the embeddings. The standard deviation is quite high between the four folds however since there are only approximately 13 positives per fold it is not necessarily indicative of poor performance.

Hyperparameter	**Value**
Learning rate	0.0006205038383410536
Weight decay	0.05753330901437465
Number of epochs	4
Performance Metric	**Value**
Mean F1 score	0.4805
Standard deviation	0.1474

Table 4.7: Best hyperparameters and performance metrics for senolytic fine-tuning hyperparameter optimisation of ChemBERTa-2

The best hyperparameter configuration found as stated in Table 4.7 were then trained on a 70% 30% stratified train validation split to obtain performance metrics. Hugging Face has an argument `load_best_model_at_end` which uses the best model seen across epochs based on a chosen metric rather than just using the final epoch only. The best F1 score was 0.5517 after all four epochs as shown in Table 4.8. We summarise the ChemBERTa-2 fine-tuning results in Table 4.10. The F1 score is 0.0712 higher than the mean F1 score, and 0.0845 higher than the F1 score for the best feature extraction classifier in Section 4.2.1.

Epoch	**Training Loss**	**Validation Loss**	**Validation Accuracy**	**Validation Recall**	**Validation Precision**	**Validation F1**
1	0.9291	1.8799	0.9789	0.0588	1.0000	0.1111
2	0.6651	1.2079	0.9802	0.3529	0.6000	0.4444
3	0.4013	1.3303	0.9841	0.4118	0.7778	0.5385
4	0.3158	1.2367	0.9828	0.4706	0.6667	0.5517

Table 4.8: Training and validation metrics across epochs of best hyperparameters found for senolytic fine-tuning of ChemBERTa-2

The training and validation loss curves in Figure 4.7 indicate overfitting is not occurring for this model.

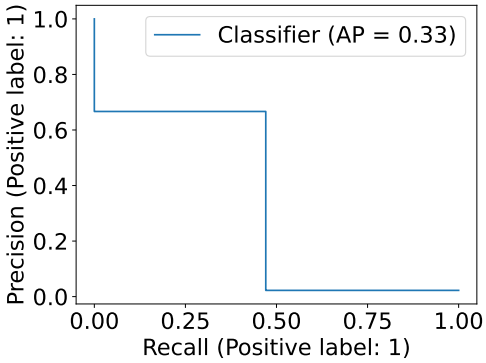

Figure 4.8: Precision-recall curve for fine-tuned ChemBERTa-2 model on senolytic training set using the best hyperparameters found in Table 4.7

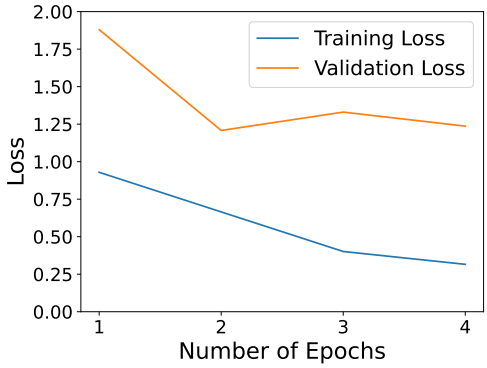

Figure 4.7: Training and validation loss curves the fine-tuned ChemBERTa-2 model using the best hyperparameters found in Table 4.7

The confusion matrix for the best model and the precision-recall curve are outlined in Figure 4.9 and Figure 4.8. The area under the precision-recall curve is comparable (0.01 higher) to that of the KNN model used on the ChemBERTa-2 embeddings.

		Predicted	
		0	1
True	0	736	4
	1	9	8

Table 4.9: Confusion matrix for best fine-tuned ChemBERTa-2 model for senolytics using the best hyperparameters found in Table 4.7

Training Loss	Validation Loss	Validation Accuracy	Validation Recall	Validation Precision	Validation F1
0.3158	1.2367	0.9828	0.4706	0.6667	0.5517

Table 4.10: Training and validation metrics for fine-tuned ChemBERTa-2 for senolytics using the best hyperparameters found in Table 4.7

4.3.2 Antibiotics

The best hyperparameter configuration was found using Optuna on 50 trials. The number of trials was lower than for the senolytics dataset because we ran into issues where the code was killed before completion after approximately 60 trials due to RAM usage. The best hyperparameters found are outlined in Table 4.11 along with the mean F1 score across the stratified cross-validation folds. The standard deviation is low and the mean F1 score is slightly higher than the best mean F1 score of 0.3102 for the best embeddings classifier during hyperparameter optimisation. The mean F1 is lower than the F1 scores for the senolytics embeddings classifier and ChemBERTa-2 fine-tuning, however.

Hyperparameter	**Value**
Learning rate	0.00020709433087541914
Weight decay	5.140649443776566e-05
Number of epochs	4
Performance Metric	**Value**
Mean F1	0.3380816225769492
Standard deviation	0.009783221808823905

Table 4.11: Best hyperparameters and performance metrics for antibiotic fine-tuning hyperparameter optimisation of ChemBERTa-2

The best hyperparameters obtained from Optuna hyperparameter optimisation were then trained on a 70% 30% stratified training validation split of the training compounds to obtain performance metrics. The metrics across epochs are displayed in Table 4.12.

The `load_best_model_at_end` argument in Hugging Face was originally used to choose the best number of epochs based on the F1 score, however when using these results on the test set, the obtained probability distribution was undesirable with the probability prediction range being too high. In the case of drug discovery, we do not want predictions to be too high since this results in wasted costs for laboratory tests. Therefore the best number of epochs was selected based on the precision metric which occurred after three epochs as we can see in Table 4.12. This model led to a better probability distribution which we outline in Section 4.4.2.

We also carried out hyperparameter optimisation trials based on the best mean precision score rather than F1, however, the probability distribution was very poor and narrow,

with only a couple of compounds having probability scores above 50%, and the range of probabilities was much narrower than when optimised on F1 score and were deemed too high. The latter is unexpected for hyperparameter optimisation on the precision metric since precision favours fewer false positives so you would expect the probability predictions to be lower on average than when optimised on the F1 score yet this was not the case. These training issues an additional sign of the difficulties ChemBERTa-2 has with training on the antibiotic dataset and that it is not able to transfer its previously learned knowledge well to this task.

Epoch	Training Loss	Validation Loss	Validation Accuracy	Validation Recall	Validation Precision	Validation F1
1	0.7005	0.8603	0.9293	0.1736	0.3623	0.2347
2	0.6403	0.9251	0.9141	0.3264	0.3176	0.3219
3	0.6372	1.3034	0.9363	0.2083	0.4762	0.2899
4	0.6037	1.2741	0.9345	0.2153	0.4493	0.2911

Table 4.12: Training and validation metrics across epochs of fine-tuning ChemBERTa-2 for antibiotic prediction using best hyperparameters found in Table 4.11

The training and validation loss curves for the best hyperparameters are seen in Figure 4.9. These indicate the model may be starting to overfit from epoch 2 onwards.

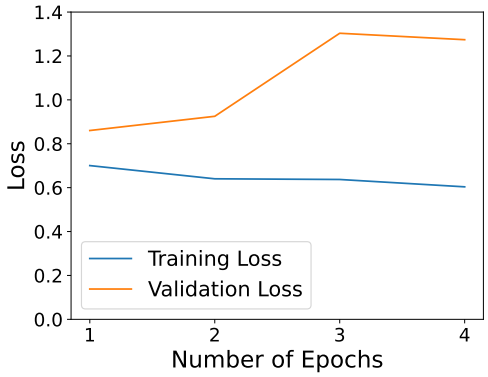

Figure 4.9: Training validation loss curves of fine-tuned ChemBERTa-2 for antibiotic prediction using best hyperparameters found in Table 4.11

		Predicted	
		0	1
True	0	2129	33
	1	114	30

Table 4.13: Confusion matrix for best fine-tuned ChemBERTa-2 model for antibiotics using the best hyperparameters found in Table 4.11

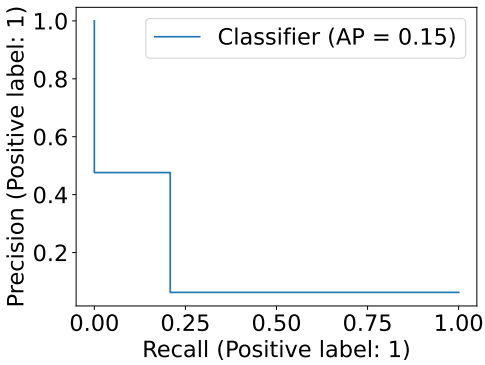

Figure 4.10: Precision-recall curve for fine-tuned ChemBERTa-2 model on antibiotic training set using the best hyperparameters found in Table 4.11

Table 4.13 shows the confusion matrix and Figure 4.10 depicts the precision-recall curve for the fine-tuned ChemBERTa-2 model for antibiotic prediction. The area under the precision-recall curve is 0.1 lower than that of the embeddings classifier for feature extraction.

4.4 Test Compounds Analysis

In this section, we perform a comparative study of the fine-tuned ChemBERTa-2 model for senolytic prediction and the XGBoost classifier from Smer-Barreto et al. (2023) [2] as well as the fine-tuned ChemBERTa-2 model for antibiotic prediction and the ensemble model used in Liu et al. (2023) [3].

4.4.1 Senolytics

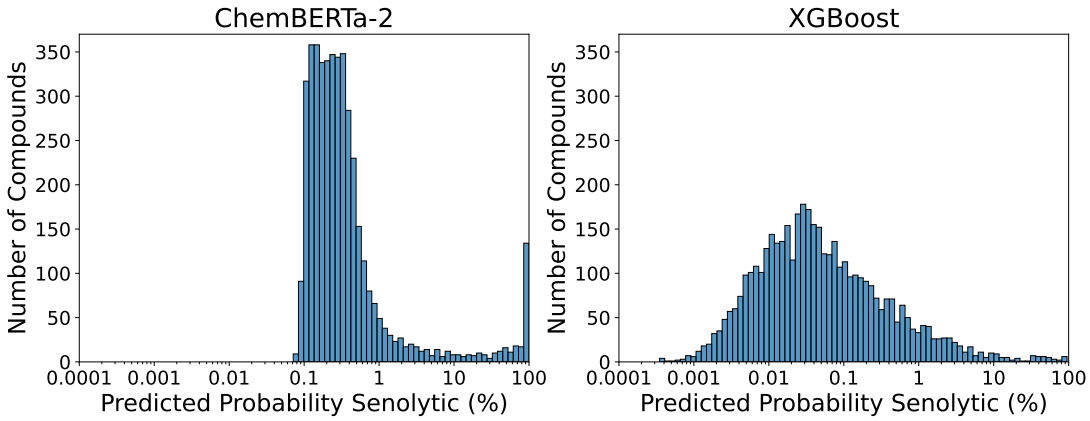

Figure 4.11: Log histogram of predicted probabilities for senolytic test compounds given by ChemBERTa-2 and XGBoost from Smer-Barreto et al. (2023) [2]

The comparison of the ChemBERTa-2 test compounds' probability distribution with the XGBoost test compounds' probability distribution can be seen in Figure 4.11. The

ChemBERTa-2 probability distribution has two distinct peaks, one at roughly 0.3% and the other close to 100%. The range of probability predictions is narrower for the ChemBERTa-2 model than for XGBoost, and the lower peak at 0.3% is much narrower than the main peak for XGBoost which is close to 0.03%. This indicates the ChemBERTa-2 model is more confident about its top predicted compounds but is less confident than the XGBoost model about its lower predicted compounds. The XGBoost model displays a more desirable probability score distribution since the model should ideally have lower probability scores and fewer false positives given the high cost of testing drug candidates in a lab.

Since we do not have labels for all 4,340 test compounds, we will zoom in on the top 21 predicted compounds by the XGBoost model from Smer-Barreto et al. (2023) [2] which were tested in the lab for senolytic activity, and compare the predicted probabilities and the overlap between the top 21 compounds from ChemBERTa-2.

The top 21 ChemBERTa-2 compounds have a much higher probability threshold at 99.8302% whereas the XGBoost compounds have a minimum probability of 44.1450%. Table 4.14 outlines the prediction probability given by ChemBERTa-2 and XGBoost for the top 21 XGBoost compounds tested in the lab. Only four of these compounds, periplocin, oleandrin, γ-mangostin and herbacetin are present in the top 21 compounds from ChemBERTa-2. Two of the three positive senolytics (highlighted in bold), oleandrin and periplocin, are present in the top 21 ChemBERTa-2 compounds and are the third and fourth highest predicted compounds for the ChemBERTa-2 model. Both are assigned much higher prediction probabilities by ChemBERTa-2 than XGBoost. However, the third senolytic, ginkgetin, received a very low prediction probability of only 0.4330%, which XGBoost had predicted at 62.4231%.

Six of the compounds in the XGBoost top 21 have ChemBERTa-2 prediction scores over 90%, two of which are positive senolytics, then there is a significant drop in probability predictions by ChemBERTa-2 for the remaining 15 compounds, 10 of which have prediction probabilities of less than 3%. Only one of these compounds is a senolytic. These are quite different to the predictions given by XGBoost, which had assigned probabilities greater than 44% to each of the top 21 compounds.

Therefore, we can observe that ChemBERTa-2 mostly chooses different compounds for its top predicted compounds. It is showing success, with two of the three positive senolytics demonstrating very high scores and most of the negative compounds having low probability predictions.

Figure 4.12 shows the cumulative scree plot for PCA applied separately to the top 21 compounds for XGBoost and ChemBERTa-2. The first two components for XGBoost explain more of the variance than those for ChemBERTa-2. This is some indication the ChemBERTa-2 compounds may display more diversity.

PCA is applied to the 21 ChemBERTa-2 compounds, XGBoost compounds from Smer-Barreto et al. (2023) [2] and the 58 positive training examples together. The first two principal components, which together explain 35.60% of the variance, are visualised in Figure 4.13 to compare the distribution of compounds in the PCA space. For the most part, both models choose compounds that are similar to positive training examples. The

Name	**Prediction Probability ChemBERTa-2 (%)**	**Prediction Probability XGBoost (%)**
Oleandrin	**99.9231**	**44.3171**
Periplocin	**99.9199**	**95.1690**
γ-Mangostin	99.8847	88.0722
Herbacetin	99.8815	58.0074
Morin	92.5118	80.0142
Velpatasvir	92.1892	56.6387
Everolimus	46.4085	54.5637
Rapamycin	38.0331	49.2538
Verteporfin	12.9436	71.7421
Paritaprevir (ABT-450)	10.4902	81.7664
γ-Oryzanol	9.9164	47.9905
Vinblastine sulfate	2.6376	88.3727
Zotarolimus	2.4789	44.1500
BMS 599626 2HCl (873837-23-1(HCl))	1.6536	92.9105
Scutellarein	1.3782	61.0926
BMS986142	0.4411	81.2854
Ginkgetin	**0.4330**	**62.4231**
Ridaforolimus	0.3536	59.1284
Ellagic acid	0.3225	44.8007
Gossypol	0.2496	50.4777
Taurocholic acid sodium salt hydrate	0.1132	55.8706

Table 4.14: Prediction probability for senolytic wet lab compounds given by ChemBERTa-2 and XGBoost from Smer-Barreto et al. (2023). The positive senolytics are written in bold text. [2]

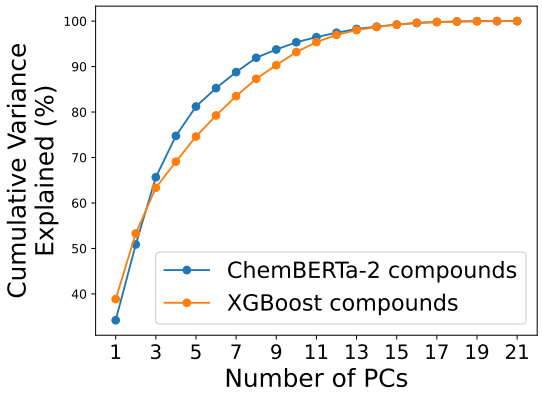

Figure 4.12: Cumulative scree plot of PCA applied to the top 21 ChemBERTa-2 compounds and the top XGBoost compounds from Smer-Barreto et al. (2023) [2] showing how much of the variance is explained cumulatively for each principal component (PC) up to 21 PCs

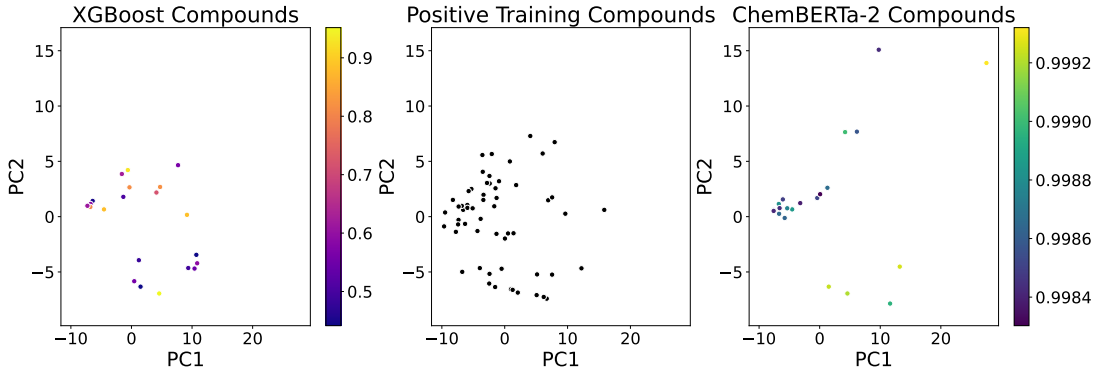

Figure 4.13: Plot for first two principal components (PCs) of PCA for the positive training compounds, top 21 ChemBERTa-2 compounds and top 21 XGBoost compounds from Smer-Barreto et al. (2023) [2]

ChemBERTa- 2 compounds are clustered in two areas on the periphery of the region the positive training examples inhabit, with two clear outliers. The XGBoost compounds are roughly located in same two regions as the ChemBERTa-2 compounds yet are more spread out, and no outliers are present. This indicates that the ChemBERTa-2 compounds are more similar to each other but show more diversity with respect to the positive traininge examples than the XGBoost compounds given the presence of outliers. The compounds for ChemBERTa-2 show a trend with higher probability compounds falling below the diagonal and lower probability compounds above the diagonal. No trend is observed for the XGBoost compound probabilities.

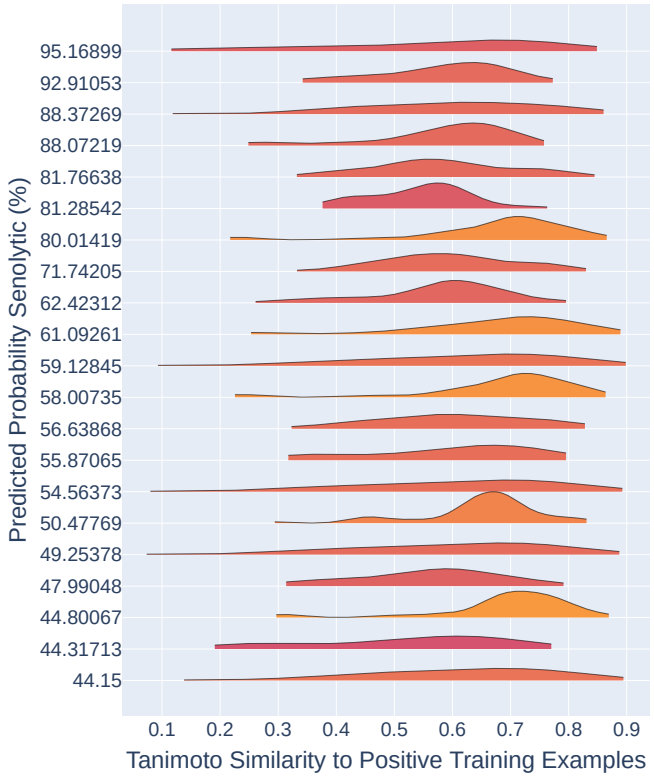

Figure 4.14: Ridge plot for the Tanimoto similarities between the top 21 XGBoost senolytics compounds from Smer-Barreto et al. (2023) [2] and the positive training examples. The colour is calculated using the min-max normalised weighted mean of every density [91]

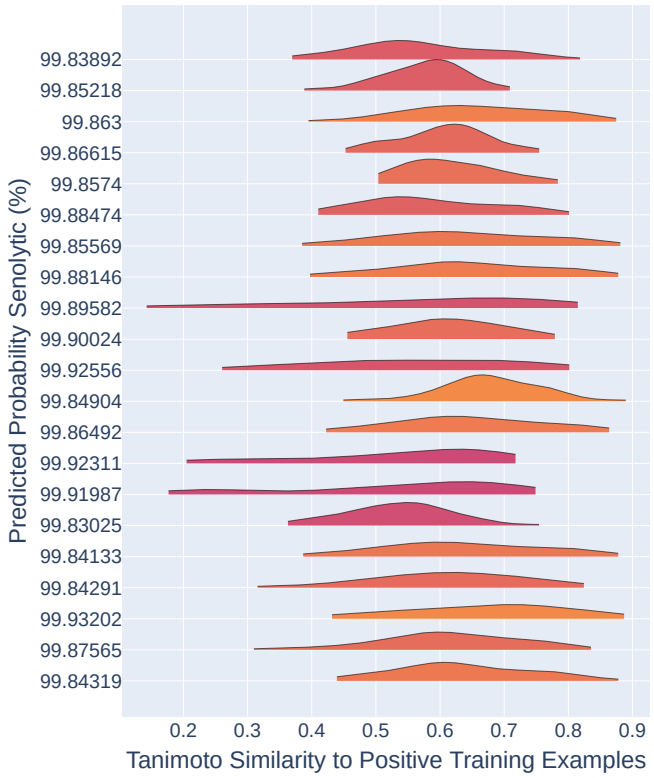

Figure 4.15: Ridge plot for the Tanimoto similarities between the top 21 ChemBERTa-2 senolytic compounds and the positive training examples. The colour is calculated using the min-max normalised weighted mean of every density [91]

The Tanimoto similarities between the top 21 XGBoost and ChemBERTa-2 compounds are calculated and the results are displayed using ridge plots in Figures 4.14 and 4.15 respectively. Both sets have peaks at similar levels but the XGBoost compound similarities are mostly concentrated in a narrow range with some outliers at the peripheries, yet the ChemBERTa-2 compounds have a more uniform spread of similarity values. From these plots, we can see that the ChemBERTa-2 compounds are more diverse since they exhibit a wider range of similarity values.

4.4.2 Antibiotics

We compare the distribution of the ChemBERTa-2 test compounds probability predictions with that of the ensemble model from Liu et al. (2023) [3] in Figure 4.16. Similar to when trained for senolytic prediction, the ChemBERTa-2 probability distribution has two peaks, however, the lower peak is at a higher value of around 2% (versus 0.3% for the senolytic dataset). This could be explained by the dataset imbalance - 2.30% of the senolytic training set compounds are positive whereas 6.38% of the antibiotic training compounds are positive, meaning higher prediction scores for antibiotic activity are expected compared to senolytic activity. ChemBERTa-2's second peak is also close to 100% for antibiotic prediction. The range of probability predictions is wider for the ChemBERTa-2 model than for the ensemble model. The main peak for the ChemBERTa-2 model is at roughly 2% whereas the ensemble model peaks around 5%. This indicates the ChemBERTa-2 model is more confident about its top predicted compounds and also its lower predicted compounds than the ensemble model. The ChemBERTa-2 predicted probability distribution is more ideal, despite the peak at close to 100%. The probability peak occurs at a lower value for ChemBERTa-2 than for the ensemble model and the probability prediction range is wider, which is more ideal when we consider the high cost of false positives in drug discovery.

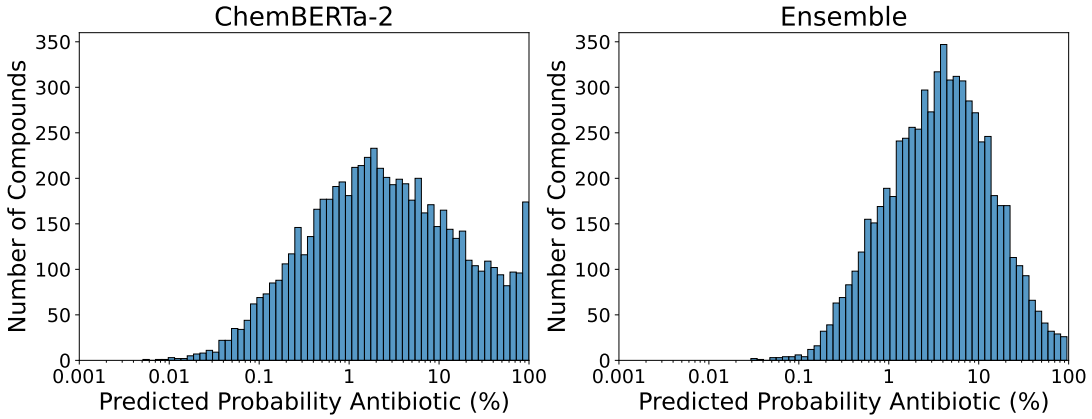

Figure 4.16: Log histogram of predicted probabilities for antibiotic test compounds given by ChemBERTa-2 and Ensemble model from Liu et al. (2023) [3]

We will narrow our focus to the top 240 compounds predicted by ChemBERTa-2 and compare them to the wet lab compounds from Liu et al. (2023) [3]. Similar to the senolytic dataset, the top 240 compounds for ChemBERTa-2 have a higher minimum

predicted probability at 76.75% versus 32.60% for the wet lab compounds in Liu et al. (2023) [3]. 80 of the 240 wet lab compounds feature in the 240 top compounds for ChemBERTa-2. 22 of these are positive for antibiotic activity, which is 55% of the total number of antibiotics found from the wet lab set.

We can see from Figure 4.17 that the majority of the wet lab compounds from Liu et al. (2023) [3] receive a high probability prediction from ChemBERTa-2, however, some receive a prediction of less than 10%.

These results indicate that ChemBERTa-2 has some overlap with the wet lab compounds and shows some similar probability scores yet differences are evident.

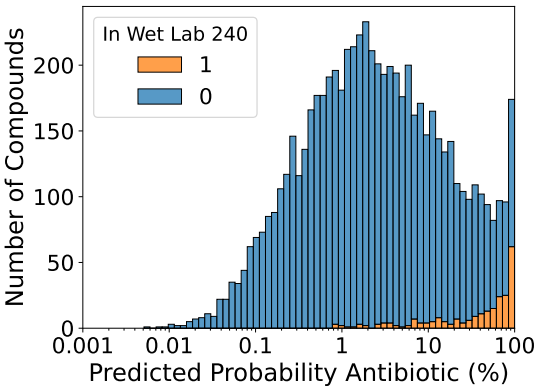

Figure 4.17: Predicted probability for antibiotic activity given by ChemBERTa-2. The wet lab samples from Liu et al. (2023) [3] are coloured in orange

In Figure 4.18, the cumulative scree plot for PCA applied to the top 240 ChemBERTa-2 compounds explains more of the variance for the same number of principal components than for the wet lab compounds. This may indicate that the compounds chosen by ChemBERTa-2 are less diverse.

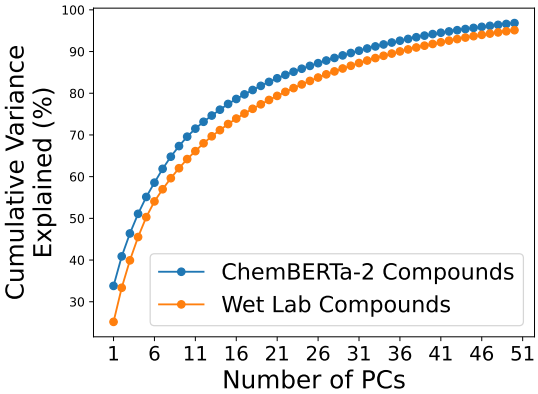

Figure 4.18: Cumulative scree plot of PCA applied to the top 240 ChemBERTa-2 compounds and the wet lab compounds from Liu et al. (2023) [3] showing how much of the variance is explained cumulatively for each principal component (PC) up to 21 PCs

Figure 4.19 however tells a different story. PCA is applied to 240 ChemBERTa-2

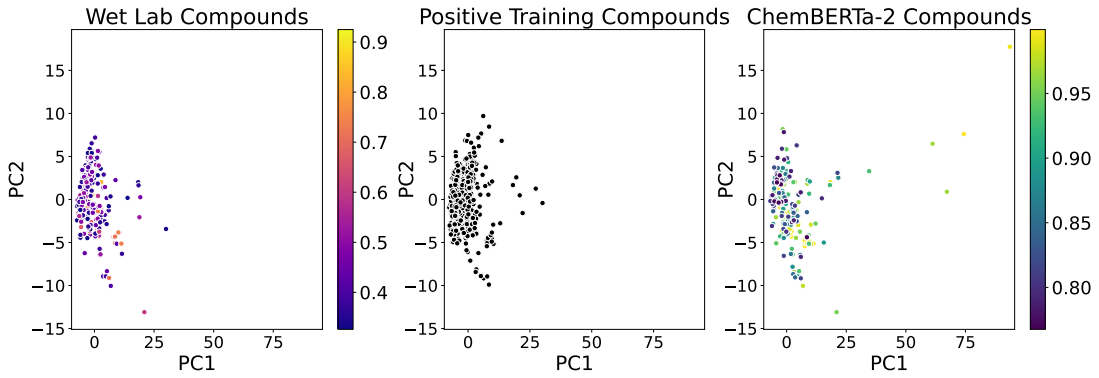

Figure 4.19: Plot of the two first principal components (PCs) of PCA for positive training compounds, top 240 ChemBERTa-2 compounds and top 21 XGBoost compounds from Liu et al. (2023) [3]

compounds, wet lab compounds and positive training examples together. We plot the compounds according to their first and second principal components, which explain 32.99% of the variance. The ChemBERTa-2 compounds and the wet lab compounds are mostly found in the same region as the positive training examples however five of the ChemBERTa-2 compounds are outliers, with four of them located very far from the positive training examples, whereas only one of the wet lab compounds is a clear outlier. The four clear outliers for ChemBERTa-2 far from the positive training examples are all on the higher end of the probability prediction scale with predictions over 90%. These plots imply the ChemBERTa-2 compounds are more diverse than those chosen by the ensemble model.

We computed the Tanimoto distance between each of the positive antibiotic training compounds and the top 240 ChemBERTa-2 compounds. We also performed the same calculations for the Tanimoto similarity between the wet lab compounds and the positive training examples. We display the ridge plot of the distribution of the Tanimoto similarity for the top 21 ChemBERTa-2 compounds in Figure 4.21 since it is infeasible to show all 240 compounds. Figure 4.20 displays the ridge plot for the top 21 wet lab compounds as predicted by the ensemble model. Most of the ChemBERTa-2 top compounds are less similar than the top 21 wet lab compounds to the positive training examples. The similarity peaks for the ChemBERTa-2 compounds mostly lie below 0.5 whereas many of the wet lab compounds have similarity peaks above 0.5. There is one exception for ChemBERTa-2, the 6th highest predicted compound, whose probability distribution is mostly above 0.6. The similarity values for ChemBERTa-2 are also lower for the antibiotic top test compounds than for the senolytic ones.

From the Tanimoto similarity and PCA space figures we can see that the ChemBERTa-2 compounds are less similar on average to the training examples, indicating more diversity in these compounds than the wet lab compounds.

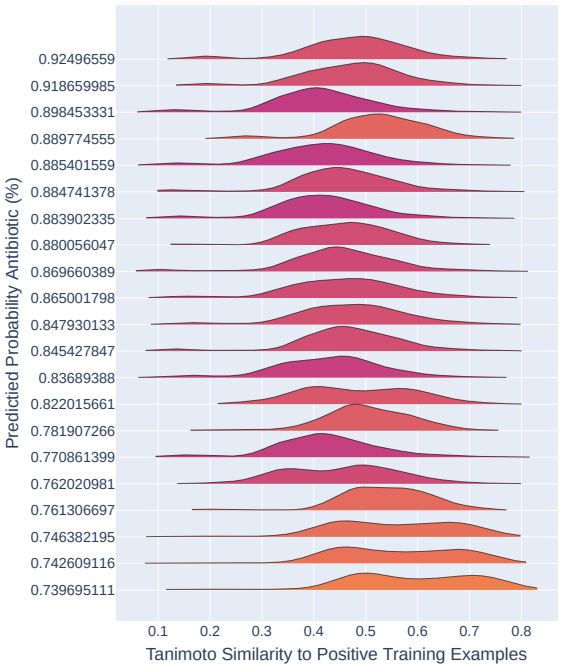

Figure 4.20: Ridge plot for the Tanimoto similarities between the top 21 wet lab antibiotic compounds from Liu et al. (2023) [3] and the positive training examples. The colour is calculated using the min-max normalised weighted mean of every density [91]

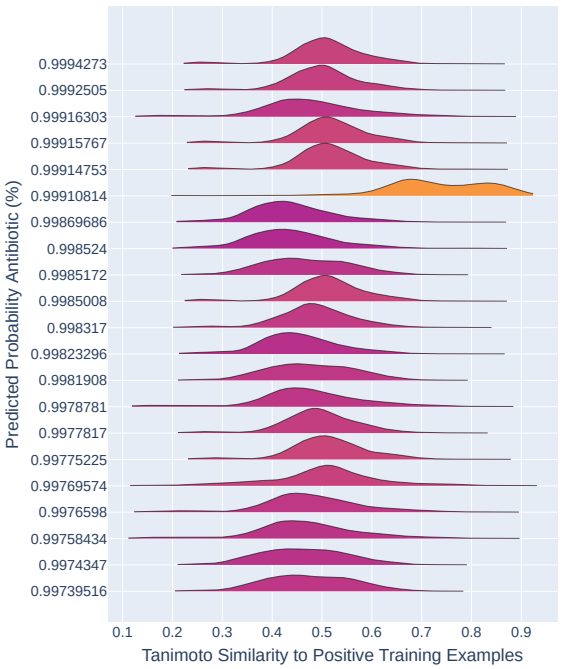

Figure 4.21: Ridge plot for the Tanimoto similarities between the top 21 ChemBERTa-2 antibiotic compounds and the positive training examples. The colour is calculated using the min-max normalised weighted mean of every density[91].

Chapter 5

Conclusions

In this chapter, we summarise our findings and make conclusions from them. We demonstrate that we have addressed the original research questions outlined. We also criticise aspects of this work and highlight future research areas that could improve the results of ChemBERTa-2 and enable us to gain further insights into how ChemBERTa-2 makes predictions.

5.1 Main Findings

This dissertation utilised the ChemBERTa-2 Large Language Model for senolytic and antibiotic prediction and benchmarked it against machine learning methods applied to the same datasets in Smer-Barreto et al. (2023) [2] and Liu et al. (2023) [3] to compare performance and compounds.

The ChemBERTa-2 embeddings are a good representation of the senolytics training data, and ChemBERTa-2 fine-tuning and feature extraction methods displayed good performance. The prediction probability distribution on the test set differs from Smer-Barreto et al. (2023) [2], and is not quite ideal for the task of drug discovery, however when investigating the outputted predictions for the 21 compounds tested in the lab, ChemBERTa-2 has very high probability prediction for 2 of the 3 detected senolytics, and most negative compounds have a low probability prediction score. The overlap between the ChemBERTa-2 compounds and the XGBoost compounds showed that ChemBERTa-2 mostly chose different compounds for senolytic detection. Plots and calculations made using PCA and Tanimoto similarity indicate the compounds chosen by ChemBERTa-2 exhibited more diversity than those selected by the XGBoost model.

The ChemBERTa-2 results for antibiotic discovery were less successful, however. The UMAP plot of the antibiotic embeddings from ChemBERTa-2 implied that the embeddings were failing to capture the compounds, indicating poor knowledge transfer from ChemBERTa-2's original training. Both feature extraction and fine-tuning of ChemBERTa-2 for antibiotic prediction had poorer performance metrics than for the senolytic dataset. The poor results of the embeddings classifier in particular may indicate that the ChemBERTa-2 embeddings are not adequate for representing the antibiotic

compounds, meaning the transfer of knowledge from ChemBERTa-2's training objective to the task of antibiotic discovery is less useful. That being said, the predicted probability distribution of the test set compounds is more favourable for drug discovery than that of the ensemble model from Liu et al. (2023) [3] and there is some indication that the top compounds chosen by ChemBERTa-2 are more diverse than those chosen by the ensemble model.

5.2 Research Question

This dissertation aimed to investigate whether LLMs can be used for the discovery of antibiotics and senolytics and how the results differed from machine learning methods previously used for these tasks. The experiments carried out demonstrate that ChemBERTa-2 can be used for senolytic and antibiotic discovery. The results for senolytic prediction proved more successful than for antibiotic prediction. We have observed that ChemBERTa-2 outputs differing probability prediction distributions for the test compounds than the machine learning classifiers, with two distinctive probability peaks rather than one. There is only a small overlap between the top predicted compounds by ChemBERTa-2 and those chosen by the respective models in Liu et al. (2023) [3] and in Smer-Barreto et al. (2023) [2] in particular.

There is also reason to believe that the compounds to which ChemBERTa-2 assigns high probability are more diverse than those chosen by the compared machine learning models from Smer-Barreto et al. (2023) [2] and Liu et al. (2023) [3]. ChemBERTa-2 had more outliers in the plots depicting the locations of the ChemBERTa-2 top compounds, machine learning methods top compounds and the positive training examples in the PCA space. The Tanimoto similarity ridge plots also indicated ChemBERTa-2's highest predicted compounds displayed more diversity than the machine learning models from Liu et al. (2023) [3] and Smer-Barreto et al. (2023) [2].

5.3 Future work

Although this dissertation addresses the research objectives outlined, there are improvements and further avenues of investigation that can be taken to enhance ChemBERTa-2's performance and to explore the predictions made in greater detail.

5.3.1 Fine-Tuning Improvements

The method used for fine-tuning ChemBERTa-2 for senolytic and antibiotic prediction could be adapted and refined. Implementing a narrow range of epoch numbers for fine-tuning was a blunt approach to overcoming overfitting, which we observed occurring after only two or three epochs during initial hyperparameter optimisation trials with a wider range of epochs. However, the training and validation loss curves for the hyperparameter configurations chosen by Optuna for the narrower range of one to four epochs suggest that further epochs could have proven beneficial to learning for the senolytic dataset. Literature and tutorials differed on whether including the number of

epochs as a hyperparameter for hyperparameter optimisation is a sensible approach. Early stopping could be explored as an alternative to preventing overfitting, this was not employed in this dissertation due to the difficulty of smoothly incorporating it into hyperparameter optimisation using cross-validation. Other hyperparameters and evaluation metrics may also enable performance gains. It would be especially beneficial to improve the probability prediction distribution outputted by ChemBERTa-2; many compounds are given a very high probability score of over 99% and ideally, the highest probability peak would occur at lower probability predictions similar to the XGBoost model from Smer-Barreto et al. (2023) [2] in Figure 4.11. Freezing some of the layers of ChemBERTa-2 may prove useful since the classifiers trained on the ChemBERTa-2 embeddings had higher mean F1 scores than the fine-tuned ChemBERTa-2 model. This indicates a good representation of the ChemBERTa-2 embeddings for the senolytic compounds and that ChemBERTa-2's knowledge has transferred reasonably well to the task. This would not be a wise approach for the antibiotics data however given the low F1 scores exhibited by the embeddings classifiers.

5.3.2 Task-Adaptive Pre-Training

The performance of ChemBERTa-2 for senolytic and antibiotic prediction could have been inhibited by ChemBERTa-2's original training data. Unfortunately, its full training set has not been released publicly however the published subset of 10 million features very few of the antibiotic and senolytic data compounds, especially the positive examples. Few compounds appearing in the ChemBERTa-2 training set would mean that the knowledge it has learned in its original training would not transfer well to antibiotic and senolytic prediction with the datasets used in this dissertation. Our findings indicate that this could be the case for the antibiotic dataset given the feature extraction results were quite poor, the UMAP plot did not exhibit the compound diversity, and the difficulties faced with fine-tuning ChemBERTa-2 for antibiotic detection.
One method that could be implemented to overcome this challenge is to use task-adaptive pre-training (TAPT). TAPT adds an additional layer of pre-training to an LLM prior to fine-tuning for tasks whose dataset is a small, task-specific subset of the general model domain [92]. This approach has demonstrated improved performance across various disciplines for the RoBERTa model [92]. This approach could be of particular benefit to antibiotic prediction. Recently, a new chemical compound model was released called ChemGPT, a Generative Pretrained Transformer-3 model based on GPT-Neo [12, 93]. ChemGPT was trained on the 10 million subset of ChemBERTa-2's training data. Preliminary experiments carried out with ChemGPT during this dissertation for senolytic detection demonstrated very poor performance, which is expected due to the lack of senolytic training examples featured in ChemGPT's training data. By leveraging TAPT, ChemGPT's prediction results could be improved and compared to those of ChemBERTa-2 to gain an insight into the influence of different LLM architectures on senolytic and antibiotic performance.

5.3.3 Chemical Compound Class Investigation

The training data of ChemBERTa-2 could also influence prediction scores based on the distribution of the classes of compounds featured in it. In the case of the senolytic prediction, ChemBERTa-2 highly favours two of the three senolytics detected in Smer-Barreto et al. (2023) [2], giving prediction scores of over 99% to both periplocin and oleandrin, however, the third senolytic, ginkgetin, received a very low probability score of only 0.4330%. This result may be impacted by the distribution of compound classes in the training set of ChemBERTa-2; periplocin and oleandrin are both examples of cardiac glycosides, whereas ginkgetin is a flavonoid [2]. We also observe clusters in the PCA plot of the top 21 ChemBERTa-2 compounds for senolytic prediction, which may indicate compound classes. Further investigation is necessary to determine whether ChemBERTa-2 favours some compound classes over others, and if so, does this result from the distribution of compounds in its training data as opposed to the model picking up on relevant features for senolytic and antibiotic discovery.

Bibliography

[1] W. Ahmad, E. Simon, S. Chithrananda, G. Grand, and B. Ramsundar, "Chemberta-2: Towards chemical foundation models," *arXiv preprint arXiv:2209.01712*, 2022.

[2] V. Smer-Barreto, A. Quintanilla, R. J. Elliott, J. C. Dawson, J. Sun, V. M. Campa, Á. Lorente-Macías, A. Unciti-Broceta, N. O. Carragher, J. C. Acosta, *et al.*, "Discovery of senolytics using machine learning," *Nature Communications*, vol. 14, no. 1, p. 3445, 2023.

[3] G. Liu, D. B. Catacutan, K. Rathod, K. Swanson, W. Jin, J. C. Mohammed, A. Chiappino-Pepe, S. A. Syed, M. Fragis, K. Rachwalski, *et al.*, "Deep learning-guided discovery of an antibiotic targeting acinetobacter baumannii," *Nature Chemical Biology*, pp. 1–9, 2023.

[4] J. P. Hughes, S. Rees, S. B. Kalindjian, and K. L. Philpott, "Principles of early drug discovery," *British journal of pharmacology*, vol. 162, no. 6, pp. 1239–1249, 2011.

[5] D. Sun, W. Gao, H. Hu, and S. Zhou, "Why 90% of clinical drug development fails and how to improve it?," *Acta Pharmaceutica Sinica B*, vol. 12, no. 7, pp. 3049–3062, 2022.

[6] A. Mullard, "The high, and redundant, cost of failure in cancer drug development.," *Nature reviews. Drug Discovery*, 2023.

[7] J. Vamathevan, D. Clark, P. Czodrowski, I. Dunham, E. Ferran, G. Lee, B. Li, A. Madabhushi, P. Shah, M. Spitzer, *et al.*, "Applications of machine learning in drug discovery and development," *Nature reviews Drug discovery*, vol. 18, no. 6, pp. 463–477, 2019.

[8] L. Gao, L. Zhang, C. Liu, and S. Wu, "Handling imbalanced medical image data: A deep-learning-based one-class classification approach," *Artificial intelligence in medicine*, vol. 108, p. 101935, 2020.

[9] T. Shaikhina and N. A. Khovanova, "Handling limited datasets with neural networks in medical applications: A small-data approach," *Artificial intelligence in medicine*, vol. 75, pp. 51–63, 2017.

[10] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler, *et al.*, "Emergent abilities of large language models," *arXiv preprint arXiv:2206.07682*, 2022.

[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[12] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[13] Y. Zang, W. Li, J. Han, K. Zhou, and C. C. Loy, "Contextual object detection with multimodal large language models," *arXiv preprint arXiv:2305.18279*, 2023.

[14] M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. d. O. Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman, *et al.*, "Evaluating large language models trained on code," *arXiv preprint arXiv:2107.03374*, 2021.

[15] A. Madani, B. Krause, E. R. Greene, S. Subramanian, B. P. Mohr, J. M. Holton, J. L. Olmos Jr, C. Xiong, Z. Z. Sun, R. Socher, *et al.*, "Large language models generate functional protein sequences across diverse families," *Nature Biotechnology*, pp. 1–8, 2023.

[16] C. Qiu, "Modeling of odor prediction from chemical structures." `https://cs230.stanford.edu/projects_fall_2020/reports/55792225.pdf`, 2020.

[17] A. S. Lang, "Fine-tuning ChemBERTa-2 for aqueous solubility prediction," *Annals of Chemical Science Research*, vol. 4, May 2023.

[18] W. Ahmad, E. Simon, S. Chithrananda, G. Grand, and B. Ramsundar, "Chemberta-2: Towards chemical foundation models," 2022.

[19] S. Nowakowska, "ChemBERTa-2: Fine-tuning for molecule's HIV replication inhibition prediction," Sept. 2023.

[20] S. Liu, J. Wang, Y. Yang, C. Wang, L. Liu, H. Guo, and C. Xiao, "Chatgpt-powered conversational drug editing using retrieval and domain feedback," *arXiv preprint arXiv:2305.18090*, 2023.

[21] F. Wong, S. Omori, N. M. Donghia, E. J. Zheng, and J. J. Collins, "Discovering small-molecule senolytics with deep neural networks," *Nature Aging*, pp. 1–17, 2023.

[22] W. H. Organization *et al.*, "2021 antibacterial agents in clinical and preclinical development: an overview and analysis," 2022.

[23] D. Weininger, "Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules," *Journal of Chemical Information and Computer Sciences*, vol. 28, no. 1, pp. 31–36, 1988.

[24] N. M. O'Boyle, "Towards a universal SMILES representation - a standard method to generate canonical SMILES based on the InChI," *J. Cheminform.*, vol. 4, p. 22, Sept. 2012.

[25] D. C. Elton, Z. Boukouvalas, M. D. Fuge, and P. W. Chung, "Deep learning for

molecular design—a review of the state of the art," *Mol. Syst. Des. Eng.*, vol. 4, pp. 828–849, 2019.

[26] B. G. Childs, M. Gluscevic, D. J. Baker, R.-M. Laberge, D. Marquess, J. Dananberg, and J. M. Van Deursen, "Senescent cells: an emerging target for diseases of ageing," *Nature reviews Drug discovery*, vol. 16, no. 10, pp. 718–735, 2017.

[27] Y. Li, J. Fan, and D. Ju, "Neurotoxicity concern about the brain targeting delivery systems," in *Brain targeted drug delivery system*, pp. 377–408, Elsevier, 2019.

[28] N. Herranz, J. Gil, *et al.*, "Mechanisms and functions of cellular senescence," *The Journal of clinical investigation*, vol. 128, no. 4, pp. 1238–1246, 2018.

[29] M. C. Velarde and R. Menon, "Positive and negative effects of cellular senescence during female reproductive aging and pregnancy.," *The Journal of Endocrinology*, vol. 230, no. 2, pp. R59–76, 2016.

[30] A. M. Andrade, M. Sun, N. S. Gasek, G. R. Hargis, R. Sharafieh, and M. Xu, "Role of senescent cells in cutaneous wound healing," *Biology*, vol. 11, no. 12, p. 1731, 2022.

[31] E. O. Wissler Gerdes, Y. Zhu, T. Tchkonia, and J. L. Kirkland, "Discovery, development, and future application of senolytics: theories and predictions," *The FEBS journal*, vol. 287, no. 12, pp. 2418–2427, 2020.

[32] C. A. Schmitt, T. Tchkonia, L. J. Niedernhofer, P. D. Robbins, J. L. Kirkland, and S. Lee, "Covid-19 and cellular senescence," *Nature Reviews Immunology*, vol. 23, no. 4, pp. 251–263, 2023.

[33] D. McHugh and J. Gil, "Senescence and aging: Causes, consequences, and therapeutic avenues," *Journal of Cell Biology*, vol. 217, no. 1, pp. 65–77, 2018.

[34] N. I. of Aging, "Does cellular senescence hold secrets for healthier aging? — nia.nih.gov." `https://www.nia.nih.gov/news/does-cellular-senescence-hold-secrets-healthier-aging`, 2021.

[35] M. A. Kohanski, D. J. Dwyer, and J. J. Collins, "How antibiotics kill bacteria: from targets to networks," *Nature Reviews Microbiology*, vol. 8, no. 6, pp. 423–435, 2010.

[36] A. Howard, M. O'Donoghue, A. Feeney, and R. D. Sleator, "Acinetobacter baumannii: an emerging opportunistic pathogen," *Virulence*, vol. 3, no. 3, pp. 243–250, 2012.

[37] F. C. Morris, C. Dexter, X. Kostoulias, M. I. Uddin, and A. Y. Peleg, "The mechanisms of disease caused by acinetobacter baumannii," *Frontiers in microbiology*, vol. 10, p. 448380, 2019.

[38] D. Wong, T. B. Nielsen, R. A. Bonomo, P. Pantapalangkoor, B. Luna, and B. Spellberg, "Clinical and pathophysiological overview of acinetobacter infections: a century of challenges," *Clinical microbiology reviews*, vol. 30, no. 1, pp. 409–447, 2017.

[39] J. F. Turton, M. E. Kaufmann, M. J. Gill, R. Pike, P. T. Scott, J. Fishbain, D. Craft, G. Deye, S. Riddell, L. E. Lindler, *et al.*, "Comparison of acinetobacter baumannii isolates from the united kingdom and the united states that were associated with repatriated casualties of the iraq conflict," *Journal of clinical microbiology*, vol. 44, no. 7, pp. 2630–2634, 2006.

[40] C. J. Murray, K. S. Ikuta, F. Sharara, L. Swetschinski, G. R. Aguilar, A. Gray, C. Han, C. Bisignano, P. Rao, E. Wool, *et al.*, "Global burden of bacterial antimicrobial resistance in 2019: a systematic analysis," *The Lancet*, vol. 399, no. 10325, pp. 629–655, 2022.

[41] J. O'Neill, "Tackling drug-resistant infections globally: final report and recommendations," 2016.

[42] K. S. Ikuta, L. R. Swetschinski, G. R. Aguilar, F. Sharara, T. Mestrovic, A. P. Gray, N. D. Weaver, E. E. Wool, C. Han, A. G. Hayoon, *et al.*, "Global mortality associated with 33 bacterial pathogens in 2019: a systematic analysis for the global burden of disease study 2019," *The Lancet*, vol. 400, no. 10369, pp. 2221–2248, 2022.

[43] M. E. Falagas and P. I. Rafailidis, "Attributable mortality of acinetobacter baumannii: no longer a controversial issue," *Critical care*, vol. 11, pp. 1–3, 2007.

[44] C. L. Ventola, "The antibiotic resistance crisis: part 1: causes and threats," *Pharmacy and therapeutics*, vol. 40, no. 4, p. 277, 2015.

[45] D. Bajusz, A. Rácz, and K. Héberger, "Why is tanimoto index an appropriate choice for fingerprint-based similarity calculations?," *Journal of cheminformatics*, vol. 7, pp. 1–13, 2015.

[46] G. Landrum, "Rdkit: Open-source cheminformatics." http://www.rdkit.org.

[47] P. Willett, "The calculation of molecular structural similarity: principles and practice," *Molecular informatics*, vol. 33, no. 6-7, pp. 403–413, 2014.

[48] "Jaccard index - Wikipedia — en.wikipedia.org." `https://en.wikipedia.org/wiki/Jaccard_index#Tanimoto_similarity_and_distance`.

[49] G. Liu and J. M. Stokes, "A brief guide to machine learning for antibiotic discovery," *Current Opinion in Microbiology*, vol. 69, p. 102190, 2022.

[50] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5–32, 2001.

[51] M. Fratello and R. Tagliaferri, "Decision trees and random forests," in *Encyclopedia of Bioinformatics and Computational Biology* (S. Ranganathan, M. Gribskov, K. Nakai, and C. Schönbach, eds.), pp. 374–383, Oxford: Academic Press, 2019.

[52] V. F. Rodriguez-Galiano, B. Ghimire, J. Rogan, M. Chica-Olmo, and J. P. Rigol-Sanchez, "An assessment of the effectiveness of a random forest classifier for land-cover classification," *ISPRS journal of photogrammetry and remote sensing*, vol. 67, pp. 93–104, 2012.

[53] M. S. Brieuc, C. D. Waters, D. P. Drinan, and K. A. Naish, "A practical introduction to random forest for genetic association studies in ecology and evolution," *Molecular ecology resources*, vol. 18, no. 4, pp. 755–766, 2018.

[54] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794, 2016.

[55] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho, K. Chen, R. Mitchell, I. Cano, T. Zhou, *et al.*, "Xgboost: extreme gradient boosting," *R package version 0.4-2*, vol. 1, no. 4, pp. 1–4, 2015.

[56] A. Ogunleye and Q.-G. Wang, "Xgboost model for chronic kidney disease diagnosis," *IEEE/ACM transactions on computational biology and bioinformatics*, vol. 17, no. 6, pp. 2131–2140, 2019.

[57] S. Ramraj, N. Uzir, R. Sunil, and S. Banerjee, "Experimenting xgboost algorithm for prediction and classification of different datasets," *International Journal of Control Theory and Applications*, vol. 9, no. 40, pp. 651–662, 2016.

[58] S. K. Kiangala and Z. Wang, "An effective adaptive customization framework for small manufacturing plants using extreme gradient boosting-xgboost and random forest ensemble learning algorithms in an industry 4.0 environment," *Machine Learning with Applications*, vol. 4, p. 100024, 2021.

[59] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory*, pp. 144–152, 1992.

[60] V. K. Chauhan, K. Dahiya, and A. Sharma, "Problem formulations and solvers in linear svm: a review," *Artificial Intelligence Review*, vol. 52, no. 2, pp. 803–855, 2019.

[61] G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, "Knn model-based approach in classification," in *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003. Proceedings*, pp. 986–996, Springer, 2003.

[62] K. Taunk, S. De, S. Verma, and A. Swetapadma, "A brief review of nearest neighbor algorithm for learning and classification," in *2019 international conference on intelligent computing and control systems (ICCS)*, pp. 1255–1260, IEEE, 2019.

[63] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on knowledge and data engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[64] H. R. Sofaer, J. A. Hoeting, and C. S. Jarnevich, "The area under the precision-recall curve as a performance metric for rare binary events," *Methods in Ecology and Evolution*, vol. 10, no. 4, pp. 565–577, 2019.

[65] S. Ayesha, M. K. Hanif, and R. Talib, "Overview and comparative study of

dimensionality reduction techniques for high dimensional data," *Information Fusion*, vol. 59, pp. 44–58, 2020.

[66] G. T. Reddy, M. P. K. Reddy, K. Lakshmanna, R. Kaluri, D. S. Rajput, G. Srivastava, and T. Baker, "Analysis of dimensionality reduction techniques on big data," *Ieee Access*, vol. 8, pp. 54776–54788, 2020.

[67] M. Ringnér, "What is principal component analysis?," *Nature biotechnology*, vol. 26, no. 3, pp. 303–304, 2008.

[68] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley interdisciplinary reviews: computational statistics*, vol. 2, no. 4, pp. 433–459, 2010.

[69] G. H. Golub and C. Reinsch, "Singular value decomposition and least squares solutions," in *Handbook for Automatic Computation: Volume II: Linear Algebra*, pp. 134–151, Springer, 1971.

[70] L. Luo, S. Bao, and C. Tong, "Sparse robust principal component analysis with applications to fault detection and diagnosis," *Industrial & Engineering Chemistry Research*, vol. 58, no. 3, pp. 1300–1309, 2019.

[71] H. Yu, F. Khan, and V. Garaniya, "An alternative formulation of pca for process monitoring using distance correlation," *Industrial & Engineering Chemistry Research*, vol. 55, no. 3, pp. 656–669, 2016.

[72] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018.

[73] B. Min, H. Ross, E. Sulem, A. P. B. Veyseh, T. H. Nguyen, O. Sainz, E. Agirre, I. Heintz, and D. Roth, "Recent advances in natural language processing via large pre-trained language models: A survey," *ACM Computing Surveys*, vol. 56, no. 2, pp. 1–40, 2023.

[74] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler, "Aligning books and movies: Towards story-like visual explanations by watching movies and reading books," in *Proceedings of the IEEE international conference on computer vision*, pp. 19–27, 2015.

[75] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "Glue: A multi-task benchmark and analysis platform for natural language understanding," *arXiv preprint arXiv:1804.07461*, 2018.

[76] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "Squad: 100,000+ questions for machine comprehension of text," *arXiv preprint arXiv:1606.05250*, 2016.

[77] P. Rajpurkar, R. Jia, and P. Liang, "Know what you don't know: Unanswerable questions for squad," *arXiv preprint arXiv:1806.03822*, 2018.

[78] R. Zellers, Y. Bisk, R. Schwartz, and Y. Choi, "Swag: A large-scale adversarial dataset for grounded commonsense inference," *arXiv preprint arXiv:1808.05326*, 2018.

[79] C. Lothritz, K. Allix, L. Veiber, J. Klein, and T. F. D. A. Bissyande, "Evaluating pretrained transformer-based models on the task of fine-grained named entity recognition," in *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 3750–3760, 2020.

[80] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[81] G. Lai, Q. Xie, H. Liu, Y. Yang, and E. Hovy, "Race: Large-scale reading comprehension dataset from examinations," *arXiv preprint arXiv:1704.04683*, 2017.

[82] S. Chithrananda, G. Grand, and B. Ramsundar, "Chemberta: large-scale self-supervised pretraining for molecular property prediction," *arXiv preprint arXiv:2010.09885*, 2020.

[83] Z. Wu, B. Ramsundar, E. N. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing, and V. Pande, "Moleculenet: a benchmark for molecular machine learning," *Chemical science*, vol. 9, no. 2, pp. 513–530, 2018.

[84] E. Heid, K. P. Greenman, Y. Chung, S.-C. Li, D. E. Graff, F. H. Vermeire, H. Wu, W. H. Green, and C. J. McGill, "Chemprop: A machine learning package for chemical property prediction," *Journal of Chemical Information and Modeling*, vol. 64, no. 1, pp. 9–17, 2023.

[85] W. Ahmad, E. Simon, S. Chithrananda, G. Grand, and B. Ramsundar, "Paper page - ChemBERTa-2: Towards Chemical Foundation Models — huggingface.co." `https://huggingface.co/papers/2209.01712`, 2022.

[86] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[87] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 2623–2631, 2019.

[88] PyTorch, "CrossEntropyLoss &x2014; PyTorch 2.2 documentation — pytorch.org." `https://pytorch.org/docs/stable/generated/torch.nn.CrossEntropyLoss.html`.

[89] H. Face, "Fine-tune a pretrained model — huggingface.co." `https://huggingface.co/docs/transformers/en/training`.

[90] H. Face, "Hyperparameter Search using Trainer API — huggingface.co." `https://huggingface.co/docs/transformers/en/hpo_train`.

[91] ridgeplot, "ridgeplot.ridgeplot documentation." `https://ridgeplot.`

readthedocs.io/en/stable/api/public/ridgeplot.ridgeplot.html#ridgeplot.ridgeplot.

[92] S. Gururangan, A. Marasović, S. Swayamdipta, K. Lo, I. Beltagy, D. Downey, and N. A. Smith, "Don't stop pretraining: Adapt language models to domains and tasks," *arXiv preprint arXiv:2004.10964*, 2020.

[93] S. Black, L. Gao, P. Wang, C. Leahy, and S. Biderman, "Gpt-neo: Large scale autoregressive language modeling with mesh-tensorflow," 2021.