RESEARCH

Using CNN to determine the aetiology from simulated and boosted microbiological data

Awais Qureshi[1], Abdul Wahid[1,2*], Shams Qazi[1], Muhammad Moazam Fraz[1] and Hasan Ali Khattak[1]

*Correspondence:
a.wahid@bham.ac.uk
[1]National University of Science and Technology, School of Electrical Engineering and Computer Science (NUST-SEECS), Sector H-12, Islamabad, Pakistan
[2]University of Birmingham, School of Computer Science
Full list of author information is available at the end of the article

Abstract

Background: Metagenomics is an emerging domain. A pivotal feature of metagenomics is the microbiome. This is a biological commodity that provides insight into the diversity of the human gut microbiome, reveals novel genes and microbial pathways, and determines functional dysbiosis. Gut microbiomes encode 100 times more genes than genomes. Microorganisms live in the gastrointestinal tract in numbers ranging from 30 trillion to 400 trillion. When studying the microbiome at such a scale, several challenges can impede its effectiveness during sample collection, storage, and sequencing. Microbiome data analysis has been complicated by unbalanced and incomplete data as well as by dimensionality, or a limited number of samples in a dataset with many dimensions and features. Bioinformaticians and other healthcare professionals might be able to analyze and understand such enormous amounts of data using artificial intelligence (AI) techniques like Machine Learning and Deep Learning. For this reason, we used and evaluated a Convolutional Neural Network (CNN) on the microbiome image data extracted from the synthetic dataset.

Results: Our results on synthetic and augmented samples revealed acceptable and comparable performances on balanced samples compared to their machine-learning counterparts. Test confidence stalls between 60-90% on balanced datasets, but with imbalanced datasets, test confidence can be as high as 85%, but with incorrect results. As a result, Imbalanced data samples would affect the overall classification accuracy and generate incorrect predictions. Furthermore, training accuracy increases as the number of samples increases until a certain threshold value is reached, after which it declines. But on the other hand, testing confidence decreases as the number of samples and features increases until eventually it increases and becomes stable at [80-90]%.

Conclusions: We examined how dropout and augmentation techniques could affect synthetic datasets with overfitting. As compared with machine-learning samples, synthetic and augmented samples delivered comparable and acceptable results. A CNN's performance is impacted by data augmentation and dropout. This study does not focus on improving classification performance on datasets that are imbalanced by oversampling, undersampling, and ensemble learning.

Keywords: Deep Learning; Convolutional Neural Networks; Metagenomics; Disease Predictions; Data Augmentation; Machine Learning

Background

"Metagenomics" is the blend of the two words "meta" and "genomics". Genomics negotiates with the means of getting the DNA sequence, but the word 'meta' reveals that we are performing it on many organisms altogether. Metagenomics is applied when we are exploring microbial inhabitants where we cannot recognize one microbe

from another. This is like when there may be two bacteria that evolve at the same time, and so when you take the DNA sequence, you are getting the DNA sequence of two bacteria together[1].

The leading player in metagenomics is called a "Microbiome." Microbes are tiny organisms that we can hardly see under a microscope, like bacteria, fungi, and viruses, they are living the world over. They live in the soil and water, and they live inside and outside of our bodies as shown in figure 1. There is no habitat on the planet without microbes. In many ecosystems, microbes perform crucial functions. They live inside us as well, where they help us digest food and protect us from pathogens [1]. To study these complex communities, we can glimpse at microbial

Figure 1: **Human Microbiome Ecosystem.** The human microbiota is a complex ecosystem made up of bacteria, viruses, fungi, and protozoans. It is made up of a group of genes known as the MICROBIOME.

DNA. We accomplish this by sequencing it – a process that turns the information encoded in the DNA into strings of letters – A, C, G, and T for adenine, cytosine, guanine, and thymine. Each of these stretches is called a read as shown in figure 2.

Microbiome data analysis seeks to answer the: "Who is there?" (Micro-organism diversity & abundance), "what can they do?" (Functional capabilities), and "what phenotype are they associated with?" (e.g., symbiosis, pathogenesis, etc.). To address this question, numerous processes are required to go from raw sequence reads to a count table, namely, DNA extraction, sequence alignment, taxonomic profiling, and functional analysis [2] - a transformational pipeline as shown in the figure the 3. The two most used pipelines or microbiome data analysis packages for 16S

Figure 2: **Microbiome Transformation - From Sequence to Predictions.** Data analysis on microorganisms aims to answer the question "Who are they?" "What can they do?" (Microorganism Diversity and Abundance). This includes functional abilities and phenotype associations (such as symbiosis, pathogens, etc.). Raw sequence readings require multiple conversion steps.

sequencing data are QIIME2 [3] and mother [4]. Both are typically command-line tools and provide graphical user interfaces (GUI) as well.

The open-source MG-RAST web application server [5] is used for Whole-Genome Sequencing data, performing all processing stages on the uploaded data and displaying the resulting taxon and function count in different representations. Similar to MEGAN6 [6], which is a desktop application, processing steps in MEGAN6 can be managed through a graphical user interface. Last but not least, CloVR [7] can be utilized online or locally.

The microbiome transformational pipeline as shown in figure 3 covers every step from quality control to count table generation. The ability of a pipeline to run all steps at once without further user input simplifies read processing. These pipelines also make it easier to try out different programs and keep track of the programs and

their parameters as they progress through each step. Because parameter changes can significantly impact the count table and thus the results, it is critical to keep track of program versions and parameters. Using a pipeline makes this task easier.[8]

With the introduction of new molecular microbiota profiling tools, such as Next-Generation Sequencing (NGS) and metagenomics shotgun sequencing, we can now learn more about the microbiota's influence in both healthy and pathological settings. [9].

These techniques have accelerated the development of bioinformatics techniques since they generate vast amounts of data. As a result, the link between microbiota, illnesses, and the clinical relevance is currently unclear.

Therefore, in this scenario, advances in artificial intelligence (AI) techniques such as Machine Learning (ML) and Deep Learning (DL) can assist clinicians in processing and interpreting information extracted from these gigantic amounts of data[10].

As a result of these developments, clinicians can make better sense of the information extracted from these data sets by using Machine Learning (ML) and Deep Learning (DL) [10]. After passing through different pipelines, the resultant micro-

Figure 3: **Metagenomics With ML - High-Level View.** Following the generation of Operational Taxonomic Units (OTUs) using OTU generation pipelines such as QIIME2 and Mothur, the next step is the conversion of OTU tables to feature vectors, followed by the feature extraction and model selection phases of machine learning until the final predictions are made.

biome data are often organized into Operational Taxonomic Units (OTUs) as shown in the figure 4.

After the OTU has been generated, then the next step is a taxonomic assignment - a process in which taxonomy is employed to characterize the relationship between the microbes and each OTU.

In an OTU table as shown in the figure 5, the number of reads per sample per OTU is represented by a matrix i.e., an organization of closely related species such as bacteria, fungi, and viruses into clusters based on sequence similarity [11].

This taxonomic correlation can be used in machine learning for the taxonomy-learned feature selection as shown in the figure 5, enabling the selected features to be used as an input for ML algorithms [10]. Data from Operational Taxonomic Units

Figure 4: **Operational Taxonomic Units.** Based on the OTU approach, similar reads are grouped together into one representative group that may contain more than one organism from the sample. A similarity threshold of 97% sequence identity is frequently used to generate clusters from sequencing data.

(OTU) is analyzed using samples and categories of attributes such as healthy or sick, and day or night. OTU counts and frequencies are used to describe observations, that is, columns in the OTU table. The column could be viewed as a feature vector in machine learning terminology.

There is a lot of research being done on Machine Learning and Deep Learning to analyze microbiomes. However, Deep Learning is catching more attention in metagenomics because of its automatic feature representation or learning abilities.

Figure 5: **OTUs with Predictions.** The data that comprise microbiomes are frequently grouped into Operational Taxonomic Units (OTUs), each of which represents an individual bacterium. Identifying the relationships between microbes within each OTU requires taxonomy. With this correlation, machine learning can perform taxonomy-learned feature selection.

Therefore, in this context, we are looking to explore CNN (Convolutional Neural Networks) in this study because previous studies have not evaluated the performance of CNNs on image data [10]. The data is available as OTU tables but should be converted into a 2D matrix to be compatible with CNNs. In addition to this, a full-featured synthetic dataset is needed to evaluate CNN performance. For that purpose, we have the following contributions to this paper:

1 Evaluate the performance estimation of CNNs in terms of classification on the full-featured metagenomic synthetic dataset.
2 Use Data Augmentation and Dropout techniques to overcome the consequences of overfitting on synthetic data.

The rest of the paper is organized as follows. Section Literature Review: ML and DL in Metagenomics will give a summary of ML and DL in Metagenomics. The information about the dataset and the methodology adopted are discussed in sections Methods. Experiments and Results insights are in Results sections. Finally, the Discussion and Conclusion sections will end up the paper.

Literature Review: ML and DL in Metagenomics

Machine Learning in Metagenomics

The most basic definition of machine learning is the process of training software - .i.e. a model, to make useful data predictions. It deals with creating and evaluating algorithms to identify, classify, and forecast data patterns. It has three flavours as shown in figure 6 to work with: Traditionally, supervised methods are used in

Figure 6: **Machine Learning Types**

metagenomics for the prediction of host traits. That is, we can identify features associated with a trait (labels) by training a model. Among the major challenges in the problem is that there are a large number of features (m rows) in the OTU table and the sample size n may be small. Therefore, complex models may over-fit the data. Trying to understand how the microbiota is related to diseases, in conjunction with all the clinical implications, is even more challenging.

Several distinct ML algorithms are capable of being used to understand this correlation alone or in a hybrid approach [10] such as the following:

1 Random Forest

2 SVM
3 LASSO
4 K-Nearest Neighbors
5 Decision Tree
6 Gradient Boosting Decision Tree
7 EXtreme Gradient Boosting
8 Naïve Bayes
9 Multinomial Naïve Bayes
10 Logistic Regression

The most often applied ML algorithms[10] are Random Forest and Support Vector Machines as shown in figure 7. However, ML algorithms have faced challenges concerning microbiome data [12]. There are a variety of challenges associated with OTU tables. The first problem with OTU tables is that they are sparse, with many zero counts. OTUs that occur in too few samples have often been deleted or collapsed down to the genus level, a form of feature engineering. Furthermore, it's often unclear how taxonomy affects prediction - small sequences are frequently correlated between samples, a property that can be assessed directly without knowledge of taxonomy. A third issue is that, like many omics technologies, library sizes (the sum of the columns in the OTU table) can vary greatly. This variation must be taken into account when normalizing data. Conventional ML techniques as outlined in

Figure 7: **Machine Learning in Metagenomics**

the different review studies [10] can be constrained by the representation capability of the microbiome data, and hence cannot learn complex patterns from the data. With the advent of machine learning comes the so-called "Curse of Dimensionality". The number of dimensions in the input data increases the complexity of ML algorithms.[10].

To resolve the challenges, Deep Learning settled into a rescue. Deep learning is a subset of machine learning which uses multiple layers of artificial neurons called neural networks to learn complex patterns from data as shown in figure 8.

Figure 8: **Deep Learning**

Deep Learning in Metagenomics

Deep Learning is gaining momentum in metagenomics to discover microbiome and disease patterns to answer "Who is there" and "What are they doing" questions. Several techniques have been tested such as CNN with different architectures, Recurrent Neural Networks (RNN), non-linear dimensionality reductions like autoencoders, Graph Convolutional Networks, and so on.

The most recent work in metagenomics w.r.t deep learning consists of the following deep learning methods.

1 Multi-layer Perceptron Network (MLP) [13–17]

2 Convolutional Neural Networks (CNN) [14, 18–22]
3 Recurrent Neural Networks (RNN) [23–25]
4 Gated Recurrent Units (GRU) [23]
5 Natural Language Processing (NLP) [24]
6 Bi-directional Long Short-Term Memory (LSTM)[26]
7 Graph Convolution Network (GCN) [27, 28]
8 Deep Neural Networks (DNN) [29]
9 Ontology-aware neural network (ONN) [30]

In this paper, we are more interested in Convolutional Neural Networks (CNN) as a widely adopted Deep Learning method in Computer Vision and Image Processing domains. As microbiome data is not in image format normally, so we have to convert OTU tables into the amendable form first. Let's review some of the most recent work.

Meta-Signer [20] exploited the phylogenetic structures in microbial taxa for the host phenotype prediction. The proposed technique receives input in a 2D matrix format – which is a phylogenetic tree filled with relative abundances of microbial taxa present in a metagenomic sample in a way such that rows of the matrix represent an evolutionary relationship between species and the column of the matrix stands for taxonomic information. This transformation enables CNNs (Convolutional Neural networks) to investigate the spatial relationship of the taxonomic annotations on the tree and its quantitative characteristics, thus proving outstanding performance for multi-class classification problems.

This approach has been benchmarked with Random Forest, Support Vector Machine, LASSO, MLP-NN, and 1D-CNN models on three different publicly available datasets [5] generated using whole metagenome shotgun sequencing (WMS). The area under the receiver operating characteristic curve (AUC-ROC), the area under the precision-recall curve (AUC-ROC), Matthews's correlation coefficient (MCC), and F1-Score were reported. They concluded that the feature extraction method of the proposed approach can find more informative features than hand-crafted feature selection methods in a shallow learning paradigm.

A different CNN-based framework PopPhy-CNN [21] has evolved by the same group as [20] and dealt with the very same host phenotype prediction from the metagenomic samples. Their approach used nine publicly accessible data sets [31], [32], [33], and [34] that comprise real and synthetic data samples generated using Metagenomic Shotgun (MGS) sequencing.

Results show that PopPhy-CNN [21] models are competitive in comparison to RF, SVM, LASSO, 1D-CNN, MLPNN, and Ph-CNN models in a comparative analysis for binary classification datasets. The limitations are the information loss in the form of feature extraction methods adopted within the approach to the like kernel map activation at the first convolutional layer for mere interpretation.

The microbiome data is highly dimensional and requires an efficient data visualization method to conduct exploratory data analysis, hence, predicting phenotypes. Therefore, a novel approach [35] based on Self-Organizing Maps, is an unsupervised deep learning technique proposed to visualize metagenomic data but also influence progress to improve disease predictions. It has been assessed on six metagenomic datasets using abundance profiles of species and proved substantial improvements in the performance but also enables visualization of biomedical signatures.

Another proposed novel scheme [27] which deals with the multiclass disease problem, has adopted the GCN (Graph Convolutional Networks)- Graph Convolutional Networks to reveal the phylogenetic structure of the microbiomes, and equated the performance of the proposed scheme with the standard feed-forward Deep Neural Networks (DNN) and conventional ML methods like Random Forest. The dataset [36] was curated with 5643 annotated whole-community metagenomes to categorize 18 different diseases. They have shown that their proposed scheme out-performed DNN in terms of accuracy which is 75% on the test set and achieved a 92.1% ROC-AUC curve, and the precision-recall is 50% on average. However, there are limitations in this study such as that various samples have multiple labels, missing labels, or are labeled as "Control." This class imbalance has affected the overall accuracy of the model.

Similarly, another method proposed by [18] tackles that same challenge as [37] and illustrates the supervised learning approach for the visualization of metagenomic data built upon the Linear Discriminant Analysis (LDA) technique. In this instance, synthetic images have been created from metagenomic data read counts with four different techniques such as using bins of synthetic images, making image colormaps, using the Fill-Up method to convert species abundances to images, and finally using supervised dimensionality reduction algorithms. Then run three different Convolutional Neural Network (CNN) deep learning architectures such as 1D, 2D, and VGG-like CNN to predict the diseases from supplied data. The scheme was evaluated on the datasets[31],[38], and [36]. Results revealed that the performance of CNN architectures decreases as the number of layers increases and the shallow architectures of CNN such as one and two convolutional layers performed better as compared to VGG-like.

Another work MetaNN [14] has dealt with the same problems of the high-dimensional nature of the metagenomic data and the low number of samples with new data augmentation techniques and overcome the effects of overfitting in the classification of body sites, subjects, and disease states. MetaNN used eight separate real metagenomic datasets [39], [40], [41] mainly consisting of 16s rRNA sequencing data plus synthetic datasets generated by using Naïve Bayes distribution. In this way, fitted Naïve Bayes distribution to generate augmented samples for each class of interest. Using synthetic data augmentation and the adopted dropout technique drastically reduces overfitting impacts.

However, there are limitations in the approach as data augmentation and dropout techniques haven't been applied during model training and CNN shows inferior performance in the case of synthetic datasets because of 1D convolutional and pooling layers applied directly on OTU tables in 2D matrix format rather than converted them into images.

Earlier approaches worked well on two levels of classification and considered a single, small dataset for training and test predictions and were unable to classify large datasets with thousands of features and samples. Most of the earlier studies rely on statistical or machine learning-based approaches hence being a significant limitation to the usefulness of these models.

To minimize these limitations, MegaD [29] proposed, which is based on deep neural networks for the prediction of unknown metagenomic samples and can handle large

metagenomic datasets holding thousands of features and samples with high precision and accuracy. The approach has been compared with [31],[42], [21] on three different datasets [43],[44], and the Purina dataset from the Purina group and held samples from 3096 individuals. Results showed that the MegaD performed well in terms of cross-validation accuracy which is 70% in the case of T2D, but 83% for the Cirrhosis dataset which is considered low as compared to other compared approaches.

While for the Purina dataset, its accuracy is 98.7% which is quite a good indicator to predict the disease status of unknown samples using trained models.

Previous techniques especially CNN considered topological information of phylogenetic trees in the form of matrices, though not useful for conventional machine learning and other deep learning models. It means that when topological phylogenetic data is added to models, the shape of the models is transferred from vector to matrix. These need to manage dynamically so any type of model whether Machine Learning or Deep Learning requires vector-like inputs and can extract meaningful topological information from the phylogenetic trees without changing into matrix format.

It is rare for current methods to consider abundance profiles from both known and unknown microbial organisms or communities. Known microbial organisms are mined using reference-based methods and unknown microbial organisms are extracted with the help of de-novo assembly methods.

Most of the current methods are based on abundance profiles of known microbial organisms and result in the potential loss of useful information since much of the reads cannot be effectively mapped to a particular reference database while de novo assembly-based methods do not need any reference genomes or marker sequences. Sequence assembly in de novo assembly-based methods is computationally expensive and takes a lot of time and memory. Moreover, high-dimensional data with a low sample size results in overfitting of the data. Then black-box models prevent biological explanations.

Overcoming above mentioned issues, a completely different framework is proposed which is the MetaDR [45]. It is a hybrid model that combines CNNs in a form of ensembles i.e., an ensemble of phylogenetic convolution neural networks (EPCNN) that are based on several microbial features and taxonomic representations and hence helps to reduce overfitting. Together with CNNs, a weighted Random Forest model is used to describe a concept of machine learning interpretability. Shotgun sequenced metagenomic datasets [46],[44],[47],[33] are used and evaluated against five state-of-the-art methods [31], [48], [16], [14], [49]. Findings indicated that the efficacy in terms of the AUC of our MetaDR is best on three out of four datasets. Execution time with different baseline predictors is also quite promising. However, the absence of a truly independent validation set is the major issue, and running time complexity is high.

Hence from the above review, we can conclude that most of the techniques in metagenomics are answering questions of Who is there i.e., Phenotype Predictions, and What are they doing i.e., Host microbiome interactions and proposed feature extraction, feature selection, data augmentation techniques to answer the questions as shown in Table 1.

Table 1: Brief overview on Deep Learning In Metagenomics

Paper	Reference	Proposed Method	Problem Addressed	Traits	Dataset	Technique
Meta-Signer	[20]	ML (RF, SVM, LASSO), DL (MLPNN, PopPhy-CNN)	Host Microbiome Interactions	Cirrhosis, Obesity, Type 2 Diabetes (T2D), Inflammatory Bowel Disease (IBD), Colorectal Cancer (CRC)	05 Different Datasets Metagenomic Shotgun Sequencing (MGS)	Feature Extraction
PopPhy-CNN	[21]	Convolutional Neural Networks (CNN)	Host Phenotype Prediction	Cirrhosis, Type 2 Diabetes (T2D), and Obesity, Inflammatory Bowel Disease (IBD): Crohne's disease (CD), ileal Crohne's disease (iCD), ulcerative cholitis (UC)	09 Metagenomic Datasets Metagenomic Shotgun (MGS) Sequencing	Data Augmentation
MicroPheno	[13]	Machine Learning (RF, Linear SVM) Deep Learning (MLP)	Environment Prediction Host Phenotype Prediction	Body-site identification, Crohn's Disease Prediction Environment Predictions	03 Datasets 16S rRNA Sequencing Data	Feature Selection
TaxoNN	[22]	CNN With Ensemble Learning	Disease State Predictions	Cirrhosis, Type 2 diabetes (T2D)	Gut Microbiome Simulated Datasets Metagenomic Shotgun Sequencing (MGS)	Feature Extraction
DeepEn-Phy	[15]	Phylogeny-Driven DNN (Vanilla MLP) with Ensemble Learning	Host Clinical Outcome Prediction	Smoking Status	16s rRNA marker gene sequencing data Guangdong Gut Microbiome Project GGMP Data	Feature Extraction
MetaNN	[14]	Multi-layer Perceptron (MLP) Convolutional Neural Networks (CNN)	Host Phenotype Prediction	Classification of body sites, subjects, disease states	08 Datasets 16S rRNA Sequencing Data	Data Augmentation

Methods

Acquisition and preprocessing of metagenomic data

The dataset used in this study is a curatedMetagenomicData [36] data package that provides the processed human microbiome data containing viral, taxonomic, bacterial, and archaeal abundances plus metabolic functional profiles with an associated meta-data of participants. The most significant benefit is the accessibility of bioinformatic resources with nominal familiarity and its integration with R/Bioconductor environments stipulates the flexibility for biologists, clinicians, epidemiologists, or statisticians to accomplish novel analyses and methodological development.

According to the paper[36], these resources have been generated by raw sequencing data, their processing via MetaPhlAn2 [50] and HUMAnN2 [51] pipelines, then by hand curated samples and studies information. This repository of curated metagenomic data is well-documented and easily usable by broad scientific communities. The current release of curatedMetagenomicData (0.4.0) now incorporates 20,533 samples from ninety studies covering fifty-one diseases and twenty-eight countries. A total of 251 samples added since Bioconductor 3.14. One can view the download stats of this dataset from its inception to the current date by visiting (http://bioconductor.org/packages/stats/data-experiment/curatedMetagenomicData/) as shown in figure 9. By following this dataset, we consider the binary classification in this study and classify 06 diseases.

Figure 9: **curated Metagenomic Dataset Stats**

We assessed our method on six different datasets including data estimated at the species level from different diseases, namely: liver cirrhosis (CIR), colorectal cancer (COL), obesity (OBE), inflammatory bowel disease (IBD), and Type 2 Diabetes (T2D) [38]. Furthermore, one dataset, such as WT2, which holds 96 European women with n=53 T2D patients and n=43 healthy individuals was also examined [31] [38]. The data is downloaded from curatedMetagenomicData [36] in R. The summary of the dataset is given in the Table 2

Table 2: Overview of the datasets evaluated in the experimentation

Dataset name	Disease	Disease Samples	Control Samples	Total Samples	Ratio of Patients	Ratio of Controls	Features in Dataset
Colorectal	*Colorectal Cancer*	48	73	121	0.40	0.60	503
Cirrhosis	*Liver Cirrhosis*	118	114	232	0.51	0.49	542
IBD	*Inflammatory Bowel Disease*	25	85	110	0.23	0.77	443
Obesity	*Obesity*	164	89	253	0.65	0.35	465
T2D	*Type 2 Diabetes*	170	174	344	0.49	0.51	572
WT2D	*Women Type 2 Diabetes*	53	43	96	0.552	0.448	381

Image Classification And Convolution Neural Networks

Images are categorized according to their features. An image's edges, pixel intensity, or change in pixel values are examples of features. How the Deep Learning model classifies photos is the real mystery, though. Each item in a 2D matrix used to store digital images corresponds to the brightness or intensity of a single pixel. Black is made up of smaller integers (closer to zero), and white is made up of larger numbers (closer to 255). The same is true with black-and-white pictures. Red, green, and blue are the three primary colors from which all other hues in a colorful image are generated. Red, Green, and Blue are the three matrices (or channels) that make up a colored image. The color intensity of each pixel in each matrix ranges from 0-255. If 450 is the height, 428 is the breadth, and 3 is the number of color channels, the ideal image shape might be 450x428x3. When we say 450 x 428, we are referring to the 192,600 pixels in the data, each of which has an RGB value and three distinct color channels.

It can be difficult to identify between backgrounds, color scales, and other elements because they vary from image to image. Convolutional Neural Networks, or CNNs, can be helpful in this situation. Using the weights and biases associated with the input photos, it is a class of deep learning algorithms that can take photographs as input, assign priority to various elements or objects, and distinguish one from the other. As a result, it operates by extracting features from the images. Any CNN consists of the following:

1 The input layer is a gray-scale/color image.
2 The Output layer generates a binary or multi-class label.
3 The convolution layers, ReLU (rectified linear unit) layers, the pooling layers, and a fully connected Neural Network forms Hidden layers.

As artificial neural networks (ANNs) consist of multiple neurons, they cannot extract features from images. This is performed with convolution and pooling layers. The convolution and pooling layers cannot classify images. To classify the objects into distinct categories, we need a dense layer at the end.

Metagenomics Images Generation

OTU (Operational Taxonomic Units) analysis is based on observations and categories such as healthy/sick, and day/night. Observations are described by the OTU counts or frequencies in a sample, which is columns within the OTU table. The column can be viewed as a vector in machine learning terminology called a feature vector (OTU feature).

After the OTU is generated, there should be some mechanism in place to convert OTUs (Operational Taxonomic Units) into images for image classification by CNNs into healthy and sick classes.

In this paper, we have reclaimed the technique of Met2Img [38]. Using Fill-up and t-SNE, Met2Img visualizes features into images. For the sake of more clarity, we have used phylogenetic sorting using the Fill-up technique. According to the paper, [38], the Fill-Up approach outperformed other methods of generating images. In this study, we have used only Gray Scale and Color images of metagenomes via the Fill-Up approach as shown in figure 10. The complete details of converting OTU abundance tables into images are given here (https://pypi.org/project/deepmg/).

(a) **Cirrhosis Gray Images** (b) **Cirrhosis Rainbow Images**

Figure 10: **Cirrhosis Gray and Color Images Example.** as generated by Met2Img and by using the techniques of Fill-up and t-SNE. The generated images have dimensions 24 by 24.

Evaluation Metrics

Model quality is measured by metrics. It is necessary to evaluate a machine learning model or algorithm. A variety of metrics are used to evaluate testing models. These include the following:

1 Accuracy
2 Confusion Matrix
3 Precision and Recall
4 F1-score
5 AU-ROC

Accuracy

A classifier's accuracy is measured by how often it makes a prediction. The accuracy is calculated by dividing the number of correct predictions by the total number of predictions. If a model gives a 99% accuracy rate, we might think that the model works very well. However, this is not always true and can be misleading in some situations. Then we have to look for other performance metrics. It is defined mathematically as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Confusion Matrix

The output of a classification model can be categorical in several ways. We can't find individual error cases in our model, since most error measures calculate the total error. A standard measure of accuracy cannot indicate whether the model misclassifies some categories more than others. This is where the confusion matrix helps us as shown in figure 11 is useful.

Figure 11: **Confusion Matrix**

- **True Positive**: A positive value is predicted, and it is correct.
- **False Positive**: A negative value is predicted, and it is actually positive.
- **True Negative**: A negative value is predicted, and it is actually negative.
- **False Negative**: A negative value predicted, and it is actually positive.

Precision

Precision is the ability of the model to correctly classify positive values. These are true positives divided by the total number of positive values predicted.

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

Recall
It is used to determine if the model is able to predict positive results. The model predicts positive values quite often, but how often does it get it right? We calculate this by dividing the total number of positive values indicated by the total number of real positive values.

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

F1-Score
The F1-score metric is based on a combination of precision and recall. In particular, the F1 score is the harmonic average of both. It provides a good balance between accuracy and recall and gives good results on unbalanced classification problems.

$$F1Score = 2(\frac{Precision * Recall}{Precision + Recall})$$

AUC(Area Under ROC curve)
In classification problems, the AUC-ROC curve is used to assess performance at different threshold settings. AUC is a measure of separation between two points and ROC is a probability curve.

Results

During the training process of neural networks, data is passed through the model, and the predictions are then compared with the labels that were derived from the ground truth. As a result of this comparison, a loss function is used. The categorical crossentropy loss is the loss function of choice when solving multiclass classification problems.

Consequently, it assumes that your labels are one-hot encoded, but this is not always the case. In that case, sparse categorical cross-entropy loss as shown in the

Figure 12: **CNN Architecture Used.** The input to the architecture is a 24 by 24 image with 03 channels in the case of the color image. The first layer is the convolution layer, the second is the pooling layer and after that two more convolution and pooling layers just before the flattened layers. There is a total of 97698 trainable parameters

equation 1 can be a viable choice. In this loss function, categorical cross-entropy loss is performed on integer targets instead of one-hot encoded ones. Therefore, for this reason, we have used a SparseCategoricalCrossentropy loss with an Adaptive Moment Estimation(Adam) Optimizer. Truth labels are integer encoded in sparse categorical cross-entropy, for example, [1], [2] for a 2-class problem.

$$L_{CE} = -\sum_{i=1}^{n} t_i log(p_i), \tag{1}$$

where p_i is the softmax probability for the ith class and t_i is the truth label.

The adaptive learning rate method is used by Adam to calculate individual learning rates for each parameter. Adaptive moment estimation gives rise to the name. Adapting the learning rate is done by estimating the first and second moments of the gradient for each weight in the neural network as shown in the equations 2 and3.

$$\beta_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{2}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \tag{3}$$

The update rule in Adam optimizer is given below:

$$\Theta_i = \Theta_i - \frac{\epsilon}{\delta + \sqrt{v_t}} \beta_t \tag{4}$$

The architecture of the Convolutional Neural Network (CNN) used in this study consists of the input layer, then subsequent convolutional and pooling layers, finally the dense layer and the output layer as shown in figure 12 and the model summary is given in figure 13.

Figure 13: **Model Summary**

The input to the architecture is a 24 by 24 image. The experiments are conducted and the results are shown in the following way: Cirrhosis color images in 14a and 14b, Cirrhosis gray images in 15a and 15b, Colon Cancer gray images in 16a and 16b, Colon cancer color images in 17a and 17b, inflammatory bowel disease (ibd) gray images in 18a and 18b, inflammatory bowel disease (ibd) color images in 19a and 19b, Obesity gray images in 20a and 20b, Obesity color images in 21a and 21b, Type 2 diabetes gray images in 22a and 22b, Type 2 diabetes color images in 23a and 23b, wt2d gray images in 24a and 24b, wt2d color images in 25a and 25b.

As we can see easily in the plots 14a, 15a, 16a, 17a, 18a, 19a, 20a, 21a, 22a, 23a, 24a, 25a and as shown in the table 3, the validation accuracy remains 60 % to 85 % during the training process, while training accuracy is 1. There is a noticeable difference in accuracy between training and validation - which indicates overfitting.

In the absence of many training examples, the model may learn from noises or unwanted details, affecting the model's performance on actual examples. Overfitting is the result of this phenomenon. This means that the model will be difficult to generalize to new datasets. This is the reason why our model fitted the training data so closely that it does not generalize well to the unseen data as we can judge from the training and validation graphs before data augmentation figures 14a, 15a, 17a, 16a, 18a, 19a, 20a, 21a, 22a, 23a, 24a, 25a

In the training process, overfitting can be combated in a variety of ways. For this study, we have added dropouts and used data augmentation. A data augmentation

(a) **Pre Augmentation and Dropout Regularization.** Before Data Augmentation and Dropout, the Training Accuracy is perfectly 100% and the Validation Accuracy is 82% as shown in the left figure. Also, training and validation losses are shown in the right figure.

(b) **Post Augmentation and Dropout Regularization.** After Data Augmentation and Dropout, the Training Accuracy is 91% and the Validation Accuracy is 80%. Also, training and validation losses are shown in the right figure.

Figure 14: **Cirrhosis classification using image augmentation and dropout on Color images.** The (Cirrhosis) dataset contains 232 samples i.e. the ratio of the patients is 51% and the controls are 49%. There is a total of 542 features in the dataset. 186 samples are used for Training and the validation split contains 46 samples.

(a) **Pre Augmentation and Dropout Regularization.** Before Data Augmentation and Dropout, the Training Accuracy is perfectly 100% and the Validation Accuracy is 84% as shown in the left figure. Also, training and validation losses are shown in the right figure.

(b) **Post Augmentation and Dropout Regularization.** After Data Augmentation and Dropout, the Training Accuracy is 85% and the Validation Accuracy is 76% as shown in the left figure. Also, training and validation losses are shown in the right figure.

Figure 15: **Cirrhosis classification using image augmentation and dropout on Gray images.** The dataset (Cirrhosis) contains 232 samples i.e. the ratio of the patients is 51% and the controls are 49%. There is a total of 542 features in the dataset. 186 samples are used for Training and the validation split contains 46 samples.

technique is used to enrich existing examples by adding random transformations that generate realistic-looking images from existing examples. As a result, the model is exposed to a wider range of data aspects and can be generalized better. With

(a) **Pre Augmentation and Dropout Regularization.** Before Data Augmentation and Dropout, the Training Accuracy is perfectly 100% and Validation Accuracy is 87%.

(b) **Post Augmentation and Dropout Regularization.** After Augmentation and Dropout, the Training Accuracy is 77% and Validation Accuracy is 70%.

Figure 16: **Colon Cancer classification using image augmentation and dropout on Gray images.** The dataset (Colon Cancer) contains 121 samples i.e. the ratio of the patients is 40% and the controls are 60%. There is a total of 503 features in the dataset. 97 samples are used for Training and the validation split contains 24 samples.

dropout regularization, a fixed number of network units are randomly selected for a gradient step. As more units are dropped, regularization increases. In essence, it is the process of teaching a network to emulate an exponentially large ensemble of smaller networks. As a result, during training, dropout randomly drops out several output units from a layer (by setting the activation to zero). Input values for Dropout are fractions such as 0.1, 0.2, 0.4, etc. The output units from the applied layer will be randomly dropped out by 10%, 20%, or 40%.

(a) **Pre Augmentation and Dropout Regularization.** Before Data Augmentation and Dropout, the training accuracy is perfectly 100% and validation accuracy is 70%.

(b) **Post Augmentation and Dropout Regularization.** After Data Augmentation and Dropout, the training accuracy is 78% and validation accuracy is 70%.

Figure 17: **Colon Cancer classification using image augmentation and dropout on Color images.** The dataset contains 121 samples i.e. the ratio of the patients is 40% and the controls are 60%. There is a total of 503 features in the dataset. 97 samples are used for Training and the validation split contains 24 samples.

Preferably after the data augmentation in the form of Random Flip, Random Rotation, and Random Zoom and dropouts, the overfitting goes off and is adjusted to an acceptable level as shown in figures 14b, 15b, 16b, 17b, 18b, 19b, 20b, 21b, 22b, 23b, 24b, 25b. The following table 3 summarizes the overall results.

Prior to data augmentation, the data shows overfitting as shown in figures 26 and 27. Still, after it is augmented, the accuracy of training and validation is acceptable as shown in figures 28 and 29.

The test confidence is also measured on all images whether Grayscale or Colored as shown in table 3. However, we noticed that the Inflammatory Bowel Disease (IBD)

(a) **Pre Augmentation and Dropout Regularization.** Before Augmentation and dropout, the training accuracy is perfectly 100% and the validation accuracy is 66%.

(b) **Post Augmentation and Dropout Regularization.** After Augmentation and dropout, the training accuracy is 81% and the validation accuracy is 63%.

Figure 18: **inflammatory bowel disease classification using image augmentation and dropout on Gray images.** The dataset contains 110 samples i.e. ratio of the patients is 23% and the controls are 77%. There is a total of 443 features in the dataset. 88 samples are used for Training and the validation split contains 22 samples.

(a) **Pre Augmentation and Dropout Regularization.** Before Augmentation and dropout, the training accuracy is perfectly 100% and the validation accuracy is 72%.

(b) **Post Augmentation and Dropout Regularization.** After Augmentation and dropout, the training accuracy is 88% and the validation accuracy is 61%.

Figure 19: **inflammatory bowel disease classification using image augmentation and dropout on Color images.** The dataset contains 110 samples, i.e., the patients are 23% and the controls are 77%. There is a total of 443 features in the dataset. 88 samples are used for training and the validation split contains 22 samples as shown in the left figure. Also, training and validation losses are shown in the right figure.

(a) **Pre Augmentation and Dropout Regularization.** Before Augmentation and dropout, the training accuracy is perfectly 100% while the validation accuracy is 60%.

(b) **Post Augmentation and Dropout Regularization.** After Augmentation and dropout, the training accuracy is 72% while the validation accuracy is 62%.

Figure 20: **Obesity disease classification using image augmentation and dropout on Gray images.** The dataset contains 253 samples i.e. the ratio of the patients is 65% and the controls are 35%. There is a total of 465 features in the dataset.203 samples are used for Training and the validation split contains 50 samples as shown in the left figure. Also, training and validation losses are shown in the right figure.

(a) **Pre Augmentation and Dropout Regularization.** Before Augmentation and dropout, the training accuracy is perfectly 100% while the validation accuracy is 66%.

(b) **Post Augmentation and Dropout Regularization.** After Augmentation and dropout, the training accuracy is 62% while the validation accuracy is 72%.

Figure 21: **Obesity disease classification using image augmentation and dropout on Color images.** The dataset contains 253 samples i.e. the ratio of the patients is 65% and the controls are 35%. There is a total of 465 features in the dataset.203 samples are used for Training and the validation split contains 50 samples as shown in the left figure. Also, training and validation losses are shown in the right figure.

(a) **Pre Augmentation and Dropout Regularization.** Before Image Augmentation and dropout, the training accuracy is perfectly 100% while the validation accuracy is 63%.

(b) **Post Augmentation and Dropout Regularization.** After Image Augmentation and dropout, the training accuracy is 65% while the validation accuracy is 54%.

Figure 22: **type 2 diabetes classification using image augmentation and dropout on Gray images.** The dataset contains 342 samples i.e. ratio of the patients is 49% and the controls are 51%. There is a total of 572 features in the dataset. 275 samples are used for Training and the validation split contains 68 samples as shown in the left figure. Also, training and validation losses are shown in the right figure.

(a) **Pre Augmentation and Dropout Regularization.** Before Image Augmentation and dropout, the training accuracy is perfectly 100% while the validation accuracy is 63%.

(b) **Post Augmentation and Dropout Regularization.** After Image Augmentation and dropout, the training accuracy is 62% while the validation accuracy is 55%.

Figure 23: **type 2 diabetes classification using image augmentation and dropout on Color images.** The dataset contains 342 samples i.e. ratio of the patients is 49% and the controls are 51%. There is a total of 572 features in the dataset. 275 samples are used for Training and the validation split contains 68 samples as shown in the left figure. Also, training and validation losses are shown in the right figure.

(a) **Pre Augmentation and Dropout Regularization.** Before image Augmentation and dropout, the training accuracy is perfectly 100% while the validation accuracy is 68%

(b) **Post Augmentation and Dropout Regularization.** After Augmentation, the training accuracy is 67% while the validation accuracy is 63%.

Figure 24: **women type 2 diabetes classification using image augmentation and dropout on Gray images.** The dataset contains 95 samples i.e. ratio of the patients is 55% and the controls are 44%. There is a total of 572 features in the dataset. 76 samples are used for Training and the validation split contains 19 samples as shown in the left figure. Also, training and validation losses are shown in the right figure.

(a) **Pre Augmentation and Dropout Regularization.** Before image Augmentation and dropout, the training accuracy is perfectly 100% while the validation accuracy is 78%

(b) **Post Augmentation and Dropout Regularization.** After image Augmentation and dropout, the training accuracy is 67% while the validation accuracy is 68%

Figure 25: **women type 2 diabetes classification using image augmentation and dropout on Color images.** The dataset contains 95 samples i.e. ratio of the patients is 55% and the controls are 44%. There is a total of 572 features in the dataset. 76 samples are used for Training and the validation split contains 19 samples as shown in the left figure. Also, training and validation losses are shown in the right figure.

Table 3: Results before and after image augmentation and dropout techniques applied on synthetic images. The test confidence for IBD (Inflammatory Bowel Disease) and Colon Cancer (Col) is in the range of [80 - 92] %, but the predictions from the test set are incorrect. The training accuracy and validation accuracy before and after Image Augmentation, as well as the dropouts, are also shown.

						Before Image Augmentation & Dropout		After Image Augmentation & Dropout			
S#	Image Type	Dataset	Total Samples	Train Samples	Validation Samples	Training Accuracy	Validation Accuracy	Training Accuracy	Validation Accuracy	Test Confidence	Prediction
1	Color	ibd	110	88	22	1	0.7273	**0.8864**	**0.619**	**85.53%**	**Wrong**
2	Gray	ibd	110	88	22	1	0.6667	**0.8182**	**0.6364**	**80.36%**	**Wrong**
3	Gray	cir	232	186	46	1	0.8478	**0.8541**	**0.7609**	74.26%	Correct
4	Color	cir	232	186	46	1	0.8261	**0.9135**	**0.8043**	90.08%	Correct
5	Gray	col	121	97	24	1	0.875	**0.7732**	**0.7083**	**80.43%**	**Wrong**
6	Color	col	121	97	24	1	0.7083	**0.7812**	**0.7083**	**91.72%**	**Wrong**
7	Gray	obe	253	203	50	1	0.6	**0.6305**	**0.72**	62.63%	Correct
8	Color	obe	253	203	50	1	0.66	**0.6287**	**0.72**	62.08%	Correct
9	Gray	t2d	344	275	68	1	0.6324	**0.6509**	**0.5441**	87.78%	Correct
10	Color	t2d	344	275	68	1	0.6324	**0.6255**	**0.5588**	77.70%	Correct
11	Gray	wt2d	95	76	19	1	0.6842	**0.6711**	**0.6316**	92%	Correct
12	Color	wt2d	95	76	19	1	0.7895	**0.6711**	**0.6842**	87.08%	Correct

and Colon Cancer (Col) datasets in [36] data package have high test confidence, but their test predictions are not accurate. This is because of the imbalanced datasets. In an imbalanced classifier dataset, the number of observations varies between classes.

Figure 26: **Evaluation of training accuracy before Image Augmentation and Dropout Regularization Method.** The training accuracy is exactly 100%

Figure 27: **Evaluation of validation accuracy before Image Augmentation and Dropout Regularization Method.** Although the training accuracy is exactly 100%, the validation accuracy is significantly higher approximately [75-85]% with fewer samples. However, as the number of samples increased, the validation accuracy dropped by approximately 64%. As a result, the data shows evidence of overfitting.

This imbalance can lead to inaccurate results. The dataset is skewed towards a particular type. It is likely that algorithms trained on biased datasets will also be biased toward that same class. Other techniques, such as oversampling, undersampling, ensemble learning, etc., are available to solve this problem. This study does not cover these techniques.

Comparison with the Benchmark

We have compared our approach with Machine learning meta-analysis of large metagenomic datasets: tools and biological insights [31]. Figures 30a and 30b show the test confidence with the increasing number of samples and features while the figure 31 illustrates the comparisons and the table 4 summarizes the comparison results on the same datasets. Figure 30a shows the x-axis representing the samples and the y-axis representing the accuracies and figure 30b shows the x-axis representing the features and the y-axis representing the accuracies. Depending on the number of samples, CNN with Data Augmentation and dropout achieves classification accuracy of 65% to 80%.

Table 4: As a microbiome feature, species abundance was used. Metrics comparing the accuracy of predictions for various diseases and healthy controls. We give the best value for each dataset in bold. While italics and bold represent values that are roughly the same.

			Overall Accuracy		
Dataset name	**#samples**	**#species**	**Random Forest (RF)**	**Support Vector Machine (SVM)**	**Convolutional Neural Networks (CNN)**
Colorectal	121	503	**0.805**	**0.743**	**0.7083**
Cirrhosis	232	542	**0.877**	**0.834**	**0.8043**
IBD	110	443	**0.809**	**0.809**	0.6364
OBE	253	465	**0.644**	**0.636**	**0.6305**
T2D	344	572	**0.664**	0.613	0.5588
WT2D	96	381	0.703	0.596	**0.72**

Figure 28: **Training Accuracy After image Augmentation and dropout:** As we can see, the training accuracy falls in the range [68-95]% for the samples less than 250, but it continuously dropped below 65% for the samples greater than 250.

The accuracy of SVM or Random Forest falls with increasing sample size. The reason is that deep learning requires a lot of data. Big data sets and their complex relationships are major factors contributing to deep learning's success. A large number of training examples, or even billions, works best. Accuracy is also affected by an imbalanced sample, such as obesity and inflammatory bowel disease samples.

Discussion

Most of the techniques in metagenomics answer the question of Who is there, i.e., Phenotype Predictions and What they are doing, i.e., Interactions between host microbiomes. According to the previous research studies, feature extraction, feature selection, and data augmentation techniques are proposed to answer the questions.

However, overfitting is the main problem due to the limited number of samples and high dimensionality, known as the curse of dimensionality. The approaches like [18], [21], and [14] addressed this question by using Data Augmentation. Still, they have a few limitations such as they have not taken data augmentation and dropout techniques to address questions of overfitting on synthetic images.

Furthermore, the proposed approaches only explored 1D Convolutional and Pooling layers because each microbial count data is one-dimensional and CNN in these approaches is not able to manage sparse features since the convolution layer considers spatial relationships among features; this results in inferior performance for the synthetic datasets. Also, these approaches missed notable features and increased information loss.

Figure 29: **Validation Accuracy After Augmentation and dropout:** As we see, the validation accuracy falls in the range [65-78]% for the samples less than 250, but it continuously dropped below 60% for the samples greater than 250.

(a) **Test Confidence w.r.t samples** As we can see, test confidence continuously decreases when the number of samples increases. But after that, it continually increases.

(b) **Test Confidence w.r.t features** As we can see, test confidence continuously decreases when the number of features increases. But after that, it continually increases and becomes stable at the range of [80-85]%.

Figure 30: **Test confidence with samples and features correlation in CNNs.**

Figure 31: **Comparative Analysis with MetaML.** The samples along the x-axis and accuracy along the y-axis. In comparison to SVMs and CNNs, Random Forests are more effective at classifying data. The performance of CNN and SVM is almost equal when the samples are 242. With exactly 253 samples, CNN and Random Forest perform equally well.
In the presence of sufficient data samples for training, validation, and testing, CNN performance can be enhanced.

However, this study focused on 2D models and tested the impact of data augmentation and dropout on images of synthetic datasets. we analyzed the MetagenomicData data package [36] and created synthetic images to fit into CNNs based on it. Synthetic and augmented samples showed comparable and acceptable results compared to their machine-learning counterparts. Unbalanced datasets can have a test confidence range of 85%, but will result in incorrect results compared to balanced datasets with a range of 60-90%.

It is also pertinent to note that as the number of samples increases, the training accuracy increases until a certain threshold value is reached, after which the accuracy declines. However, as the number of samples and features increases, testing confidence decreases as well. Eventually, it will increase and becomes stable at about 80 to 90 percent as the number of samples increases.

In our experiments, we found that data augmentation and dropout have an impact on CNN image classifiers. In addition, it is still possible to improve classification performance with oversampling, undersampling, and ensemble learning when the datasets are imbalanced. However, the focus of this study is not on that. Furthermore, it would be necessary to explore other image classification architectures, such as AlexNet, ZFNet, ResNet, MobileNet, and ResNetXt on microbiome image data. However, this is totally dependent on the availability of a well-curated and balanced dataset.

Conclusion

Metagenomic information from the human microbiome provides a unique set of data that can be used for better diagnosis and prognosis of human diseases. Microbiome data are typically quantified based on their similarities to reference datasets, termed

Operational Taxonomic Units (OTUs). However, there are a number of challenges associated with OTU tables.

As we reviewed and discussed in this paper, traditional machine-learning techniques are constrained by the representation capability of microbiome data. This means that those techniques are unable to learn complex patterns from them. With the advent of machine learning comes the so-called "Curse of Dimensionality". The number of dimensions in the input data increases the complexity of ML algorithms.

A number of Deep Learning (DL) techniques are currently being used to resolve major metagenomics issues related to operational taxonomic units (OTUs), including Convolutional Neural Networks, Recurrent Neural Networks, Generative Adversarial Networks, Multilayer Perceptrons (MLP), Self-Organizing Maps (SOM), and Autoencoders.

This study's objectives were to examine Convolutional Neural Networks (CNN) and the consequences of data augmentation and dropout techniques on OTU images generated by synthetic datasets. Accuracy and performance metrics may be increased and adequately extended to unknown data if a dataset is well-balanced and has enough samples for training and validation.

Ethics approval and consent to participate
Not Applicable

Consent for publication
Not Applicable

Availability of data and materials
The datasets analyzed during the current study are available at the https://waldronlab.io/curatedMetagenomicData/

Competing interests
The authors declare that they have no competing interests.

Funding
Not Applicable

Author's contributions
A.Q. contributed to Conceptualization, Methodology, Investigation, Data Curation, and Writing - Original Draft. A.W. and S.Q. contributed to Supervision, Writing - Review, and Editing. M.M.F. and H.A.K. contributed to Formal analysis and Project/Research Administration. All authors contributed to the Analysis and Interpretation of data.

Acknowledgements
For his valuable suggestions, we thank Dr. Muhammad Adnan, Assistant Professor, (IoC), at (KUST).
Email: adnan@kust.edu.pk.

Author details
[1]National University of Science and Technology, School of Electrical Engineering and Computer Science (NUST-SEECS), Sector H-12, Islamabad, Pakistan. [2]University of Birmingham, School of Computer Science.

References

1. Julie Segre, P.D.: Glossary of Metagenomics. `https://www.genome.gov/genetics-glossary/Metagenomics`
2. Liu, Y.-X., Qin, Y., Chen, T., Lu, M., Qian, X., Guo, X., Bai, Y.: A practical guide to amplicon and metagenomic analysis of microbiome data. Protein & cell **12**(5), 315–330 (2021)
3. Bolyen, E., Rideout, J.R., Dillon, M.R., Bokulich, N.A., Abnet, C.C., Al-Ghalith, G.A., Alexander, H., Alm, E.J., Arumugam, M., Asnicar, F., *et al.*: Reproducible, interactive, scalable and extensible microbiome data science using qiime 2. Nature biotechnology **37**(8), 852–857 (2019)
4. Schloss, P.D., Westcott, S.L., Ryabin, T., Hall, J.R., Hartmann, M., Hollister, E.B., Lesniewski, R.A., Oakley, B.B., Parks, D.H., Robinson, C.J., *et al.*: Introducing mothur: open-source, platform-independent, community-supported software for describing and comparing microbial communities. Applied and environmental microbiology **75**(23), 7537–7541 (2009)
5. Keegan, K.P., Glass, E.M., Meyer, F.: Mg-rast, a metagenomics service for analysis of microbial community structure and function. In: Microbial Environmental Genomics (MEG), pp. 207–233. Springer, ??? (2016)
6. Bağcı, C., Beier, S., Górska, A., Huson, D.H.: Introduction to the analysis of environmental sequences: metagenomics with megan. In: Evolutionary Genomics, pp. 591–604. Springer, ??? (2019)

7. Angiuoli, S.V., Matalka, M., Gussman, A., Galens, K., Vangala, M., Riley, D.R., Arze, C., White, J.R., White, O., Fricke, W.F.: Clovr: a virtual machine for automated and portable sequence analysis from the desktop using cloud computing. BMC bioinformatics **12**(1), 1–15 (2011)
8. edx.org: Microbiome Course at edx. `https://learning.edx.org/course/course-v1:KULeuvenX+MICROx+1T2021/home`
9. Espinoza, J.L., Kotecha, R., Nakao, S.: Microbe-induced inflammatory signals triggering acquired bone marrow failure syndromes. Frontiers in immunology **8**, 186 (2017)
10. Iadanza, E., Fabbri, R., Bašić-ČiČak, D., Amedei, A., Telalovic, J.H.: Gut microbiota and artificial intelligence approaches: a scoping review. Health and Technology **10**(6), 1343–1358 (2020)
11. wikipedia: en.wikipedia.org/wiki/Bioinformatics. `https://en.wikipedia.org/wiki/Bioinformatics`
12. Zhou, Y.-H., Gallins, P.: A review and tutorial of machine learning methods for microbiome host trait prediction. Frontiers in genetics **10**, 579 (2019)
13. Asgari, E., Garakani, K., McHardy, A.C., Mofrad, M.R.: Micropheno: predicting environments and host phenotypes from 16s rrna gene sequencing using a k-mer based representation of shallow sub-samples. Bioinformatics **34**(13), 32–42 (2018)
14. Lo, C., Marculescu, R.: Metann: accurate classification of host phenotypes from metagenomic data using neural networks. In: Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics, pp. 608–609 (2018)
15. Ling, W., Qi, Y., Hua, X., Wu, M.C.: Deep ensemble learning over the microbial phylogenetic tree (deepen-phy). In: 2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 470–477 (2021). IEEE
16. Oh, M., Zhang, L.: Deepmicro: deep representation learning for disease prediction based on microbiome data. Scientific reports **10**(1), 1–9 (2020)
17. Khajeh, T., Reiman, D., Morley, R., Dai, Y.: Integrating microbiome and metabolome data for host disease prediction via deep neural networks. In: 2021 IEEE EMBS International Conference on Biomedical and Health Informatics (BHI), pp. 1–4 (2021). IEEE
18. Nguyen, T.H., Prifti, E., Sokolovska, N., Zucker, J.-D.: Disease prediction using synthetic image representations of metagenomic data and convolutional neural networks. In: 2019 IEEE-RIVF International Conference on Computing and Communication Technologies (RIVF), pp. 1–6 (2019). IEEE
19. Nguyen, H.T., Tran, T.B., Luong, H.H., Le, T.P., Tran, N.C.: Improving disease prediction using shallow convolutional neural networks on metagenomic data visualizations based on mean-shift clustering algorithm. International Journal of Advanced Computer Science and Applications **11**(6) (2020)
20. Reiman, D., Metwally, A., Sun, J., Dai, Y.: Meta-signer: Metagenomic signature identifier based on rank aggregation of features. F1000Research **10**(194), 194 (2021)
21. Reiman, D., Metwally, A.A., Sun, J., Dai, Y.: Popphy-cnn: a phylogenetic tree embedded architecture for convolutional neural networks to predict host phenotype from metagenomic data. IEEE journal of biomedical and health informatics **24**(10), 2993–3001 (2020)
22. Sharma, D., Paterson, A.D., Xu, W.: Taxonn: ensemble of neural networks on stratified microbiome data for disease prediction. Bioinformatics **36**(17), 4544–4550 (2020)
23. Chen, X., Liu, L., Zhang, W., Yang, J., Wong, K.-C.: Human host status inference from temporal microbiome changes via recurrent neural networks. Briefings in Bioinformatics **22**(6), 223 (2021)
24. Queyrel, M., Prifti, E., Templier, A., Zucker, J.-D.: Towards end-to-end disease prediction from raw metagenomic data. bioRxiv, 2020–10 (2021)
25. Zhao, Z., Woloszynek, S., Agbavor, F., Mell, J.C., Sokhansanj, B.A., Rosen, G.L.: Learning, visualizing and exploring 16s rrna structure using an attention-based deep neural network. PLoS computational biology **17**(9), 1009345 (2021)
26. Zeng, W., Gautam, A., Huson, D.: Deeptoa: An ensemble deep-learning approach to predicting the theater of activity of a microbiome. bioRxiv (2022)
27. Khan, S., Kelly, L.: Multiclass disease classification from microbial whole-community metagenomes. In: PACIFIC SYMPOSIUM ON BIOCOMPUTING 2020, pp. 55–66 (2019). World Scientific
28. Li, B., Zhong, D., Qiao, J., Jiang, X.: Gnpi: Graph normalization to integrate phylogenetic information for metagenomic host phenotype prediction. Methods (2022)
29. Mreyoud, Y., Song, M., Lim, J., Ahn, T.-H.: Megad: Deep learning for rapid and accurate disease status prediction of metagenomic samples. Life **12**(5), 669 (2022)
30. Zha, Y., Ning, K.: Ontology-aware neural network: a general framework for pattern mining from microbiome data. Briefings in bioinformatics **23**(2), 005 (2022)
31. Pasolli, E., Truong, D.T., Malik, F., Waldron, L., Segata, N.: Machine learning meta-analysis of large metagenomic datasets: tools and biological insights. PLoS computational biology **12**(7), 1004977 (2016)
32. Sokol, H., Leducq, V., Aschard, H., Pham, H.-P., Jegou, S., Landman, C., Cohen, D., Liguori, G., Bourrier, A., Nion-Larmurier, I., *et al.*: Fungal microbiota dysbiosis in ibd. Gut **66**(6), 1039–1048 (2017)
33. Zeller, G., Tap, J., Voigt, A.Y., Sunagawa, S., Kultima, J.R., Costea, P.I., Amiot, A., Böhm, J., Brunetti, F., Habermann, N., *et al.*: Potential of fecal microbiota for early-stage detection of colorectal cancer. Molecular systems biology **10**(11), 766 (2014)
34. Qin, J., Li, R., Raes, J., Arumugam, M., Burgdorf, K.S., Manichanh, C., Nielsen, T., Pons, N., Levenez, F., Yamada, T., *et al.*: A human gut microbial gene catalogue established by metagenomic sequencing. nature **464**(7285), 59–65 (2010)
35. Nguyen, T.H.: Metagenome-based disease classification with deep learning and visualizations based on self-organizing maps. In: International Conference on Future Data and Security Engineering, pp. 307–319 (2019). Springer
36. Pasolli, E., Schiffer, L., Manghi, P., Renson, A., Obenchain, V., Truong, D.T., Beghini, F., Malik, F., Ramos, M., Dowd, J.B., *et al.*: Accessible, curated metagenomic data through experimenthub. Nature methods **14**(11), 1023–1024 (2017)

37. Nguyen, T.H., Zucker, J.-D.: Enhancing metagenome-based disease prediction by unsupervised binning approaches. In: 2019 11th International Conference on Knowledge and Systems Engineering (KSE), pp. 1–5 (2019). IEEE
38. Nguyen, T.H., Prifti, E., Chevaleyre, Y., Sokolovska, N., Zucker, J.-D.: Disease classification in metagenomics with 2d embeddings and deep learning. arXiv preprint arXiv:1806.09046 (2018)
39. Knights, D., Costello, E.K., Knight, R.: Supervised classification of human microbiota. FEMS microbiology reviews **35**(2), 343–359 (2011)
40. Costello, E.K., Lauber, C.L., Hamady, M., Fierer, N., Gordon, J.I., Knight, R.: Bacterial community variation in human body habitats across space and time. science **326**(5960), 1694–1697 (2009)
41. Fierer, N., Lauber, C.L., Zhou, N., McDonald, D., Costello, E.K., Knight, R.: Forensic identification using skin bacterial communities. Proceedings of the National Academy of Sciences **107**(14), 6477–6481 (2010)
42. Dhungel, E., Mreyoud, Y., Gwak, H.-J., Rajeh, A., Rho, M., Ahn, T.-H.: Megar: an interactive r package for rapid sample classification and phenotype prediction using metagenome profiles and machine learning. BMC bioinformatics **22**(1), 1–12 (2021)
43. Bajaj, J.S., Betrapally, N.S., Gillevet, P.M.: Decompensated cirrhosis and microbiome interpretation. Nature **525**(7569), 1–2 (2015)
44. Karlsson, F.H., Tremaroli, V., Nookaew, I., Bergström, G., Behre, C.J., Fagerberg, B., Nielsen, J., Bäckhed, F.: Gut metagenome in european women with normal, impaired and diabetic glucose control. Nature **498**(7452), 99–103 (2013)
45. Chen, X., Zhu, Z., Zhang, W., Wang, Y., Wang, F., Yang, J., Wong, K.-C.: Human disease prediction from microbiome data by multiple feature fusion and deep learning. Iscience **25**(4), 104081 (2022)
46. Qin, J., Li, Y., Cai, Z., Li, S., Zhu, J., Zhang, F., Liang, S., Zhang, W., Guan, Y., Shen, D., *et al.*: A metagenome-wide association study of gut microbiota in type 2 diabetes. Nature **490**(7418), 55–60 (2012)
47. Qin, N., Yang, F., Li, A., Prifti, E., Chen, Y., Shao, L., Guo, J., Le Chatelier, E., Yao, J., Wu, L., *et al.*: Alterations of the human gut microbiome in liver cirrhosis. Nature **513**(7516), 59–64 (2014)
48. Zhu, Z., Ren, J., Michail, S., Sun, F.: Micropro: using metagenomic unmapped reads to provide insights into human microbiota and disease associations. Genome biology **20**(1), 1–13 (2019)
49. Zhu, X., Li, H.-D., Guo, L., Wu, F.-X., Wang, J.: Analysis of single-cell rna-seq data by clustering approaches. Current Bioinformatics **14**(4), 314–322 (2019)
50. Blanco-Miguez, A., Beghini, F., Cumbo, F., McIver, L.J., Thompson, K.N., Zolfo, M., Manghi, P., Dubois, L., Huang, K.D., Thomas, A.M., et al.: Extending and improving metagenomic taxonomic profiling with uncharacterized species with metaphlan 4. bioRxiv (2022)
51. Beghini, F., McIver, L.J., Blanco-Míguez, A., Dubois, L., Asnicar, F., Maharjan, S., Mailyan, A., Manghi, P., Scholz, M., Thomas, A.M., *et al.*: Integrating taxonomic, functional, and strain-level profiling of diverse microbial communities with biobakery 3. Elife **10**, 65088 (2021)