IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 6, 2021, accepted January 7, 2021, date of publication January 11, 2021, date of current version February 10, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3050838

Feature Extension of Gut Microbiome Data for Deep Neural Network-Based Colorectal Cancer Classification

MWENGE MULENGA[1,2], SAMEEM ABDUL KAREEM[2], AZNUL QALID MD SABRI[2], MANJEEVAN SEERA[3], SURESH GOVIND[4], CHANDRAMATHI SAMUDI[5], AND SAHARUDDIN BIN MOHAMAD[6]

[1] School of Science, Engineering and Technology, Mulungushi University, Kabwe 80415, Zambia
[2] Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur 50603, Malaysia
[3] Department of Econometrics and Business Statistics, School of Business, Monash University Malaysia, Subang Jaya 47500, Malaysia
[4] Department of Parasitology, Faculty of Medicine, University of Malaya, Kuala Lumpur 50603, Malaysia
[5] Department of Medical Microbiology, Faculty of Medicine, University of Malaya, Kuala Lumpur 50603, Malaysia
[6] Faculty of Science, Institute of Biological Sciences, University of Malaya, Kuala Lumpur 50603, Malaysia

Corresponding authors: Mwenge Mulenga (mwenge2008@yahoo.co.uk) and Aznul Qalid Md Sabri (aznulqalid@um.edu.my)

This work was supported by the Malaysia's Ministry of Higher Education via a research grant issued by the University of Malaya, through the Trans-Discipline Research Grant Scheme under Project TR001D-2018A.

ABSTRACT Colorectal cancer (CRC) is the third most deadly cancer worldwide. The use of gut microbiome in early detection of the disease has attracted much attention from the research community, mainly because of its noninvasive nature. Recent achievements in next generation sequencing technology have led to increased availability of sequence data and enabled an environment for the growth of gut microbiome research. The use of conventional machine learning algorithms for automatic detection of CRC based on the microbiome is limited by factors such as low accuracy and the need for manual selection of features. Despite their success in other fields, Deep Neural Network (DNN) algorithms have limitations in microbiome-based CRC classification. These limitations include high dimensionality of microbiome data and other characteristics associated with sequence data such as feature dominance. In this paper, we propose a feature augmentation approach that aggregates data normalization methods to extend existing features of a dataset. The proposed method combines feature extension with data augmentation to improve CRC classification performance of a DNN model. The proposed model obtained area under the curve (AUC) scores of 0.96 and 0.89 on two publicly available microbiome datasets.

INDEX TERMS Colorectal cancer, deep neural network, feature dominance, gut microbiome, normalization, feature extension.

I. INTRODUCTION

Colorectal cancer (CRC) is the third most prevalent cancer and is ranked as the fourth cause of cancer deaths worldwide [1]. Screening of CRC has proven to marginally reduce mortality rate of the disease [1]. Early detection of CRC can greatly improve the patient's survival chance [2]. Commonly used methods for early detection of CRC are Faecal Occult Blood Test (FOBT) and Colonoscopy. FOBT is limited by high false positive rates and false negative rates, whereas colonoscopy is expensive, an inconvenience to the patient, and may also cause perforation and bleeding [3]. Studies have shown strong associations between a type of microorganism called gut microbiome and the development of CRC [4]. The use of gut microbiome obtained from stool samples [5] can be a better noninvasive alternative for CRC detection. The advent of next generation sequencing technology has led to an increase in the number of sequence based datasets available such as microbiome data.

Automatic detection of CRC using machine learning (ML) algorithms has been of interest to many researchers. Several studies have used conventional ML methods to classify CRC based microbiome samples [6], [7]. These methods, however, have several limitations such as low accuracy and the need for

The associate editor coordinating the review of this manuscript and approving it for publication was Larbi Boubchir.

manual feature selection [8]. Feature engineering has been extensively used to improve the classification performance of ML methods on sequence data [9]. The work in [10] noted that the performance of deep neural network (DNN) algorithms in the classification of microbiome data is limited when the sample size is small. However, other works [11] [12], [13] have shown that using data preprocessing techniques in DNN models can improve the classification performance of CRC based on microbiome samples. In particular, data augmentation seeks to address the curse of dimensionality which is prevalent in sequence data [13]. Studies that combine feature engineering and data augmentation in order to improve CRC detection based on microbiome samples are lacking in the literature.

In this paper, we propose a method that combines feature extension and data augmentation to improve the CRC classification performance of a DNN model using microbiome samples. Our contributions are three-fold. For the first contribution, we propose a cube root (cbrt) based data normalization method that rescales data points in the dataset by first multiplying each of them by the standard deviation of the raw data, and then computing the cube root on the product in order to reduce the effect of dominant features. For the second contribution, the individual normalization methods are combined to extend features of a dataset to produce a better method for addressing feature dominance. For the third contribution, we combine data augmentation and feature extension to produce a method that is more robust as compared to using augmentation or feature extension.

Organization of the paper is as follows. Various literature is first reviewed in Section II. In Section III, details of the proposed method are given. Results are presented and discussed in Section IV. Concluding remarks are finally given in Section V.

II. LITERATURE REVIEW

Conventional ML algorithms have been widely used in the classification of microbiome data. For example, Random Forest (RF) was used in [14] to improve sensitivity of faecal immunochemical test for detecting colonic lesions. The method scored a comparatively good classification accuracy and was able to add interpretability to faecal immunochemical test-based analysis. It also had a very high sensitivity score which can help improve preventive screening of CRC. However, one drawback of the method is that it had a relatively low specificity. Another work in [15] proposed a hybrid ML method for phenotype classification that combines biomarker profiling with a k-mer frequency table. Although the method added interpretability to the model, the dataset used was very small which may have been the reason for the relatively low classification performance of the method.

To improve classification performance of ML algorithms on microbiome data, researchers have attempted a technique of combining samples from different studies. For instance, work in [16] used conventional ML algorithms to process shotgun metagenomic data, consisting of 2424 samples which were pooled from 8 different studies. The authors also used cross-disease validation, cross-study validation, and cross-validation in order to evaluate the accuracy of methods that were used to predict 6 disease classes. The pooling of samples and cross-validation across multiple studies improved classification accuracy. Despite the high prediction accuracies scored on other diseases, the accuracy on CRC was relatively low. Similarly, another study in [17] proposed a method for metagenomic data analysis based on faecal microbiome as a noninvasive biomarker in the detection of CRC. The study that included Chines, Danish, Austrian and French samples was able to identify novel species that were significantly enriched and 20 marker genes that had a significant association with CRC. The study, however, had a relatively low classification accuracy.

Authors in [18] carried out a multi-cohort analysis of CRC metagenome to identify altered bacteria in different populations. The study also sought to identify bacterial markers that were universal. They were able to identify seven bacteria that were consistently enriched across different groups regardless of technical and biological background of the samples. The drawback of this method was that it used traditional statistical methods to analyze data which were not very robust and may have led to relatively low classification accuracy. Similarly, the work in [7] performed a meta-analysis of 969 faecal metagenomic samples comprising publicly available datasets and two private cohorts. They identified microbial diagnostic signatures that were cross-cohort and had an association with the degradation of choline. The use of feature selection in addition to the pooling of heterogeneous datasets improved the prediction accuracy of the method. The method, however, was based on a traditional ML algorithm, which could limit the prediction accuracy.

The use of feature selection methods to enhance the performance of ML methods in classifying microbiome data has been of interest [19], [20]. While other studies have used methods proposed in [21], [22], a study in [23] used conventional ML techniques in order to accurately classify microbial communities associated with a disease called bacterial vaginosis (BV) using two datasets, based on amplicon sequencing. The research identified associations between microbiome and BV, which demonstrated the potential of their method. Besides, the method used feature selection, which produced highly predictive features which helped improve the classification accuracy. However, the classification results obtained across the two datasets were not comparable because the datasets were characterized by high variability owing to differences in the samples across studies and the heterogeneous nature of the preprocessing techniques that were used.

In contrast to conventional ML algorithms that depend on manual feature selection methods, DNN-based methods have an inbuilt mechanism for feature extraction [24]. DNN methods also have a better classification performance [25] as compared to traditional ML algorithms. Therefore, there has been a trend towards the use of DNN methods in order to improve the classification accuracy of microbiome data. The work in [26] proposed deep learning-based models for classifying the taxonomy of metagenomic data. The authors proposed a pipeline for classifying simulated metagenomic sequences into bacterial taxonomic classes using a deep belief network and a convolutional neural network (CNN). Although the method outperformed the reference classifier for bacteria identification called the RDP, it is based on the simulated data and was not meant for disease classification. Another work in [27] proposed a method that performs image transformation on nonimage data. The method attained very good results owing to the use of images, which enabled the method to utilize the full potential of CNNs, which included the algorithm's inbuilt feature extraction mechanism. However, the method was not tested on combined datasets for phenotype classification.

Similarly, a general regression neural networks (GRNNs) method for detecting CRC was proposed in [12]. The proposed method used a nonlinear feature selection method to filter the most predictive microbial species. Although the researchers combined GRNNs with a nonlinear feature selection method which improved detection accuracy and interpretability, sensitivity and specificity scores of the model were too low to be considered for medical application. The work in [9] proposed a microbiota classification-based feature engineering method for disease status classification that makes use of phylogenetic hierarchy. The method performed taxonomic abstraction by capitalizing on the oligophyletic distribution of important features. Therefore, the method produced a reduced feature space that in turn improved classification accuracy and interoperability of the model. However, a reduced feature space is mostly used in traditional ML algorithms and rarely in DNN based algorithms, which perform feature reduction by default.

Data augmentation and normalization have been widely used to improve disease classification. For example, a method that uses neural networks to classify phenotype of a host based on metagenomic data was proposed in [28]. In order to address overfitting, the method used data augmentation and a dropout technique. The approach presents a comparatively high classification accuracy on both synthetic and the actual metagenomic data. However, since the study did not consider pooled datasets it did not address variability across CRC datasets. Recent work in [13] investigated computational practices and design principles related to variational autoencoders (VAE) in cancer data integration. The study showed that the application of VAE produces relevant representations of data that yield stable and accurate diagnosis. Whereas the methods used in the study reduced the dimensionality of input data without compromising its quality, the downstream analysis was only evaluated on traditional ML algorithms. The researchers in [29] investigated how classification performance of ML algorithms is impacted by data normalization. Normalization methods covered in [29] included Minimum-Maximum normalization (MMN), Z-score normalization (ZSN), Variable Stability Scaling (VSS) and Pareto Scaling normalization (PSN). The authors observed that no single data normalization method outperformed other methods on all datasets. They attributed this behavior to different characteristics possessed by the datasets which in turn affect normalization methods differently. Although the study covered a wide range of normalization methods, it did not investigate the aggregation of normalization methods.

Based on the gaps identified in the review of recent related works, we propose a method that aims to combine a feature extension technique (that is based on aggregating data normalization methods) with data augmentation in order to improve DNN based classification of CRC using gut microbiome in stool samples. The proposed feature extension method transforms features in the input dataset from vector to matrix form. Extending features increases the number of important features in the dataset and can be used to solve the problem of feature dominance [29]. Data augmentation is a technique used to increase the number of samples in a dataset by synthetically producing new samples and combining them with the original data [30]. The technique is commonly used to reduce variability. In terms of sequence data, variability refers to variations in the abundance of a specific gene across samples and high variability can reduce the sensitivity of a model [31]. Therefore, the proposed method attempts to reduce variability while improving feature importance across the dataset.

III. METHOD

The experiments in this study were implemented in Python using the Keras [32] framework with TensorFlow, which was processed on a GPU using Google Colab.

A. DATASETS

In this study, two publicly available CRC based microbiome datasets were used. Dataset 1 is based on curated samples of shotgun sequence data and is detailed in [33]. The number of samples and features in the dataset are 884 and 2031, respectively. The data is a combination of samples from six countries namely United States of America (USA), Canada, France, Germany, Austria and Italy. Table 1 shows the composition of both healthy controls and CRC samples according to the country of origin.

Two metadata attributes namely Biomass Index (BMI) and age were combined with microbiome data which increased the total number of features to 2033. Another attribute called ‘‘study condition’’ which is also part of the metadata, was used as the label during ML classification. The attribute represents the health status of a sample, which can be CRC or non-CRC. Further, the total number of samples was reduced to 796 after samples that did not have any entry in the disease field (in metadata) were removed from the dataset.

Dataset 2 comprises 16S rRNA sequence data and is based on a study in [14]. In this dataset, some of the samples were collected from Toronto, Canada and the rest were collected in Houston, Boston and Ann Arbor in the USA. The dataset

TABLE 1. CRC-based microbiome samples according to the country of origin.

Country	Non-CRC	CRC	Total
United States of America	87	77	164
Germany	10	76	86
Austria	108	46	154
France	206	106	312
Italy	79	61	140
Canada	26	2	28
Total	516	368	884

contains 336 features and a total of 490 samples. The respective number of cancer, normal and adenoma samples in the dataset are 120, 172, and 198. In order to use this dataset for binary classification adenoma samples were grouped with health samples under a new label called non-CRC.

B. PROPOSED METHOD

This study proposes a method that combines data augmentation and feature extension in order to improve CRC classification performance of a DNN model using gut microbiome in stool samples. Whereas data augmentation uses a VAE to generate new data samples, the feature extension method transforms input data into a dataset that has a higher dimension. Features of the resultant dataset are obtained by combining each of the corresponding features in the input dataset (as an input feature vector) with outputs of two normalization methods that transform the same input feature vector into a matrix. Our presentation of the method starts with a description of the data normalization methods used in this study. Next we outline how the normalization methods are combined to extend existing features. This is then followed by a presentation of how the proposed method is constructed. Whereas standard deviation is used to evaluate how the normalization methods affect data variability, L2 regularization is used to determine how the methods influence feature importance [34].

1) COMPUTATION OF NORMALIZATION METHODS

The performance of ML algorithms on microbiome data is limited by feature dominance and the presence of outliers, among other things. In this experiment three normalization methods, namely PSN, ZSN, and VSS tailored to datasets that are characterized by outliers and dominant features, are compared with the proposed cbrt normalization method. We also added MMN in this study as it is a commonly used normalization method in the literature [35].

Consider dataset A that has n rows and m columns which can be represented as $A = (a_{ij}) \in R^{nx\, m}$, where the parameter a is an entry in A, i is the *i-th* row and j is the *j-th* column. Rows and columns represent instances and features respectively. In the following computations we will use a'_{ij} and a_{ij} to denote the result of a computation and a data point in the dataset respectively. The computation of MMN, a value-based normalization method, used to preserve the relationships among the original input data (orig) is shown in (1).

$$a'_{ij} = \frac{a_{ij} - min\left(a_j\right)}{\max\left(a_j\right) - \min\left(a_j\right)} (iMax - iMin) + iMin \qquad (1)$$

where max and min denote the highest and least value in the j-th column. Also, *iMax* and *iMin* denote respective highest and least values in the i-th row of the dataset. The limitation of this normalization method is that it is sensitive to outliers. Several normalization methods are suitable for reducing the effect of outliers. These normalization methods are mainly computed using the mean and standard deviation of the raw data. For example, the computation of ZSN which introduces zero means and unit variance across data features is shown in (2),

$$a'_{i,j} = \frac{a_{ij} - \mu_j}{\sigma_j} \qquad (2)$$

where μ and σ denote the standard deviation and mean respectively. A slightly different method, PSN, which uses the square root of variance as the scaling factor is computed as shown in (3).

$$a'_{ij} = \frac{a_{ij} - \mu_j}{\sqrt{\sigma_j}} \qquad (3)$$

PSN is based on a concept called Pareto optimality which attempts to make an entity better off without compromising the quality of another entity within the same group [36]. The method attempts to improve the representation of lower value features while minimizing the effect of noise in the data. Similarly, the VSS method attempts to improve the ZSN approach by introducing a scaling factor called coefficient of variation as shown in (4).

$$a'_{i,j} = \frac{a_{ij} - \mu_j}{\sigma_j} * \frac{\mu_j}{\sigma_j} \qquad (4)$$

The method scales data by giving more relevance to features with lower standard deviation and less relevance to those with higher standard deviation. Although the normalization methods mentioned above reduce the effect of outliers, they are not robust against dominant features [29]. Therefore, in this work we propose the cbrt method that multiplies each data point in the dataset with the standard deviation of the raw data and then computes the cube root of the product to rescale the data as shown in (5).

$$a'_{i,j} = 3\sqrt{a_{ij} * \sigma} \qquad (5)$$

By multiplying each data point by the standard deviation of the raw dataset the method attempts to shift the feature space with respect to the standard deviation. Computation of the cube root on the product is meant to marginally reduce the effect of extremely large values while slightly downscaling smaller values. Unlike the methods described earlier that use the standard deviation of individual features, cbrt scaling uses standard deviation of the entire dataset. This in enforces uniformity in rescaling data points and further reduces variability or does not worsen it.

2) COMPUTATION OF FEATURE EXTENSION

This study proposes a method for combining the above-mentioned data normalization techniques for use in a DNN classification task. The proposed method transforms two-dimensional (2D) input data to a three-dimensional (3D) dataset using a novel feature extension technique. The feature extension method transforms input data by combining each of its feature vector and two normalized copies of the same feature vector to produce a feature matrix. The conversion of features from vectors to matrices transforms the original dataset from 2D to 3D. Since we are combining normalization methods in order to perform feature extension, we will use the terms feature extension and combined normalization interchangeably in the remaining part of this text.

Each instance of the model uses two different normalization methods to produce a unique dataset. The mathematical representation of the feature extension method used in this experiment denoted by the parameter TT is shown in (6),

$$\mathrm{T}\left(\bar{a}\right) = \bar{a}*\bar{x}|\bar{x} = [1,F_i,F_j],F_i,\, F_j\in F \text{ and } i \neq j \qquad (6)$$

where $\bar{a}$ denotes a column vector in the raw dataset, F denotes the set of normalization methods that comprises cbrt, PSN, ZSN and VSS. Also, the subscripts i and j represent the i-th and j-th entry in F. The expression shows that every feature vector is multiplied by a 1-by-3 matrix to produce an n-by-3 matrix where n is the number of items in the feature vector. Just as the approach used in this method of increasing the feature space may increase dimensionality, it can also improve feature importance of less important features across a dataset. This is particularly important in microbiome data, which is usually affected by the problem of feature dominance. The increase in the number of features should not be a source of concern, if the quality of the new features is good, because a DNN model performs its own feature selection.

The exact computation used to extend features can be broken down into two parts. The first part iterates over all the normalization methods and arranges them in paired permutations. The second part creates transformed versions of the original dataset according to the paired permutations of normalization methods obtained in the previous step. The use of paired permutations is meant to ensure that each new dataset is obtained using a unique combination of the available normalization methods. Algorithm 1 shows the computation of paired permutations of the normalization methods.

Algorithm 1 shows that the *methodPermutation* function takes a list of normalization methods as input and returns a list containing paired permutations of the methods. Local variables of the function are declared from line 2 to 6. The outer *while* statement, which starts from line 7 to line 24 loops over the normalization methods contained in the *methodsArray* parameter. The inner *while* statement, which starts from line 11 to 18 pairs the current normalization method that is supplied by the outer loop with each of the subsequent normalization methods in the list. The paired normalization methods are stored in a variable called *innerArray*, which is subsequently stored in the *permutationsArray* variable in the inner loop as shown in lines 15 and 16. When the outer loop terminates execution, the paired normalization methods are returned as *permutationsArray* by the function. Once normalization methods are paired, each pair and the input dataset are passed to the next algorithm for further processing. Algorithm 2 outlines how a dataset is transformed using paired normalization methods.

Algorithm 1 Make Permutations of All Normalization Methods

```
    Input: methodsArray
    Output: permutationsArray
 1 : Function methodPermutation
 2 : permutationsArray ← [ ]
 3 : size ← length of methodsArray
 4 : mainCounter ← 0
 5 : outerCounter ← 0
 6 : permutationCount ← 0
 7 : while outerCounter < size do
 8 :   currentMethod ← methodsArray[outerCounter]
 9 :   innerCounter ← outer Counter + 1
10:    innerArray ← [ ]
11:    while innerCounter <= size do
12:         innerArray [0] ← methodsArray
             [outerCounter]
13:         innerArray [1] ← methodsArray
             [innerCounter]
14:          innerCounter ← innerCounter + 1
15:          permutationsArray [permutationCount]
             ← 16 innerArray
17:         permutationCount ← permutationCount +1
18:    end while
19:    outerCounter ← outerCounter +1
20:         permutationCount ← permutationCount +1
21:    end while
22:    outerCounter ← outerCounter +1
23:    permutationIndex ← permutationIndex + 1
24: end while
25: return permutationsArray
26: End Function
```

Algorithm 2 takes a 2D dataset and two other parameters (each corresponding to a method in a pair of the normalization methods) as input. It then returns a three-dimensional dataset that has the same number of samples as the input dataset. The primary purpose of the local variable *transformedDataset* defined in line 2 is to store the new dataset generated in the subsequent lines of code. The outer while statement starting from line 6 to 20 loops over the samples in the input dataset. The inner while statement starting from line 10 to 18 extends the original feature with two additional features generated according to each of the paired normalization methods as shown in lines 12 and 14. By assigning a 1-by-3 matrix to a single value as shown in line 16, we transformed a feature vector into a 2D feature matrix. Consequently, the original dataset was transformed from 2D to 3D.

Algorithm 2 Feature Extension

```
   Input: rawDataset,method1,method2
   Output: transformedDataset
Function transformDataset
transformedDataset ← [ ]
rowCount ← number of rows in rawDataset
columnCount ← number of columns in rawDataset
i ← 0
 while i < rowCount do:
    newFeature ← [ ]
    j ← 0
    while j <= columnCount do
        newFeature [0] ← rawDataset [i] [j]
        newFeature [1] ← normalize rawDataset
         [i][j]
        according to method1
        newFeature [2] ← normalize rawDataset
         [i][j]
        according to method2
         transformedDataset [i][j] ← newFeature
        j ← j+ 1
    end while
   i ← i +1
  end while
return transformedDataset
End Function
```

3) CONSTRUCTING THE MODEL

This study uses a 6 layer DNN architecture with 8 nodes in each of the four hidden layers and a sigmoid activation function in the output layer. The activation function used in the hidden layers is the rectified linear unit (ReLu). The architecture of the proposed DNN model has two variations for the classification of both extended and non-nextended data. For non-extended data, input neurons in the DNN model are in a linear form that takes in data in the form of a vector, and its size is specified by a scalar value (2033 for dataset 1 and 336 for dataset 2). To accommodate the extended samples which are in matrix form, the neurons in the input layer of the DNN are arranged in 2D accordingly (2033 × 3 for dataset 1 and 336 × 3 for dataset 2). In this case, where a matrix of neurons is used in the input layer, an additional layer that is designed to convert incoming data into a vector (flatten layer) is placed immediately after the input layer. The optimization algorithm used in the DNN is Root Mean Square propagation (RMSprop) [37] with the learning rate of 0.0008. RMSprop was used because it converges fast and is easy to fine-tune [38]. Fig. 1 shows a typical structure of a basic DNN model.

In Fig. 1, parameter Z_{ij} denotes the ReLu activation function used in each node in the hidden layers. The subscripts i and j denote the i-th layer and j-th neuron respectively. The parameter X used as input in the activation functions denotes a vector of data items from the previous layer. The parameter F denotes the sigmoid activation function used in the output layer. From Fig. 1, it can also be observed that unprocessed data is fed into the DNN model. While such a model is easy to implement and uses less computational resources, it produces very poor results because of overfitting and irregularities related to data distribution.

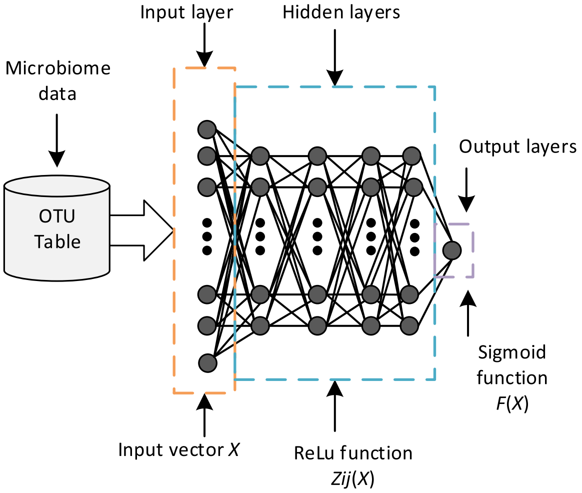

FIGURE 1. Presentation of a basic DNN model that can be used to classify microbiome data.

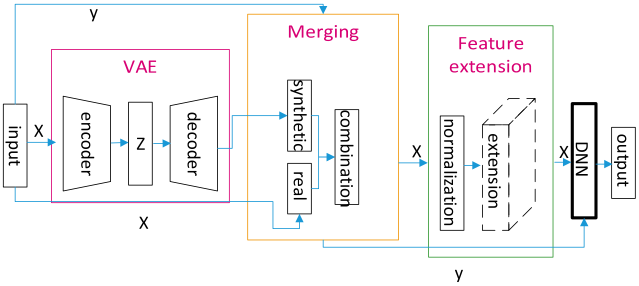

FIGURE 2. Combining data normalization, feature extension and augmentation in a DNN model. Whereas X represents features, y represents the labels in a dataset.

In this study, we propose a combination of feature extension and data augmentation to improve CRC classification accuracy of a DNN model using gut microbiome data. Data augmentation is performed first, followed by the feature extension technique before the DNN model is invoked as shown in Fig. 2.

From Fig. 2 it can be observed that data is first augmented, merged, normalized, transformed (features are extended), then passed to the DNN classifier. VAEs are increasingly being used as a data augmentation technique in ML classification tasks [13]. Whereas an autoencoder is a type of artificial neural network (ANN) that efficiently performs data encodings in an unsupervised manner, a VAE is a type of generative autoencoder that uses the probability distribution of input data to generate new samples [39]. Although there are variants of VAEs, this research uses a basic VAE to generate additional samples in both datasets (1 and 2).

Fig. 2 also shows that the VAE architecture has three major components, namely the encoder, the latent space denoted by the letter Z, and the decoder [13], [39]. The encoder converts input data to a reduced feature space which is called the latent space. The decoder then uses the data distribution represented in the latent space to try and generate new samples. A VAE does not generate new data items for individual items in a dataset but computes a distribution in which the input data lies. Therefore, based on the user specified value, a VAE can generate an arbitrary number of synthetic samples for a given dataset. Since binary classification was used in the experiment, input data was separated into the two subgroups consisting of CRC and non-CRC samples in order to perform augmentation on each subgroup separately. Otherwise, there would be no way to tell apart labels of the newly generated samples. Further, data without labels is fed into the VAE and then the output is merged with the appropriate label. Subsequently, the labelled raw data, depending on whether it is CRC or non-CRC, is combined with the newly generated samples appropriately. Although it is not shown in the illustration, CRC and non-CRC samples are also recombined before passing them into the feature extension module. In this experiment 250 CRC and another collection of 250 non-CRC samples were generated in both dataset 1 and 2. Therefore, a total of 500 new samples were generated in each of the two datasets.

Data normalization, the basis of the proposed method, is a preprocessing technique that either transforms or rescales data in order to produce equal relevance to each feature [29] which improves the performance of a ML model. In order to reduce the impact of data distribution problems, raw data is usually normalized before it is fed into a DNN model. In data normalization, features are transformed, whereas the label is not as shown in Fig. 2. The feature extension technique used in the proposed method transforms a feature vector into a matrix in order to improve prediction accuracy. The feature extension module which combines each feature vector and two normalized copies of the same feature vector to produce a feature matrix, transforms the input data from 2D to 3D. The transformed dataset has more features than the original dataset leading to an increase in significant features. However, increasing the number of features does not mean an automatic increase in the number of important features or the transformation of a dataset into one with better qualities. It is for this reason that the normalization methods used in the data transformation technique are carefully selected.

Whereas augmentation reduces variability and normalization improves the relevance of less dominant features, combining the two methods should produce a method that is more powerful than either of the two approaches used in isolation. In this experiment we also determine how the constituent components of the model and their combinations affect underlying properties of the data, which in turn alter the overall performance of model. Therefore, the model is constructed and evaluated starting from a basic DNN, DNN with single normalization method, DNN with combined normalization methods, DNN with augmentation, DNN with augmentation and single normalization method, and the proposed DNN with augmentation and combined normalization methods.

IV. RESULTS AND DISCUSSIONS

In the previous section, we described the methods used in this study. First, the cbrt method was described and compared with existing single normalization method techniques outlined in section III. Then *cbrt* and the rest of the single normalization methods were combined, using permutations described in section III, to perform feature extension. In order to transform input data, the feature extension method created two additional features for every single feature in a dataset using paired normalization methods. In this section we used feature importance scores and variability observed in datasets to analyze how the outlined methods affect the datasets. L2 regularization was used to compute feature importance scores and standard deviation was used to measure variability in the dataset associated with each of the methods described in section III. Computation of feature importance plays a central role in feature selection tasks, of which L2 regularization is one of the methods used [34]. In this study, we used L2 regularization to compute feature importance from all the datasets before and after augmentation, and normalization. The feature importance score associated with each method was used as the basis for evaluating the performance of the methods. Further, standard performance metrics were used to evaluate how these methods affect the DNN model.

A. PERFORMANCE METRICS

In order to evaluate the performance of the proposed model with the baseline methods, three performance measures, namely sensitivity, specificity, and area under the curve (AUC) were used. A confusion matrix was used to compute these performance measures and it has two rows and two columns. The first column in the first row represents the True Positive (TP) value, which is a number of records in which CRC is correctly classified; The second column in the first row represents the False Positive (FP) value, which is the number of records in which CRC samples are misclassified as health samples; The first column in the second row represents the False Negative (FN) value, which is a number of records in which health samples are misclassified as CRC samples; The second column in the second row represents the True Negative (TN) value, which is the number of records in which the health samples are correctly classified.

Sensitivity, which is also called the True Positive Rate (TPR), is the proportion of the actual CRC cases that are correctly classified and is computed as shown in (7). It is desirable for a model to have high sensitivity.

$$\text{Sensitivity} = \text{TPR} = \frac{TP}{TP + FN} \tag{7}$$

Specificity or the True Negative Rate (TNR) measures the proportion of health samples that are correctly classified and is computed as shown in (8). A good model is also expected to have high specificity.

$$\text{Specificity} = \text{TNR} = \frac{TN}{TN + FP} \tag{8}$$

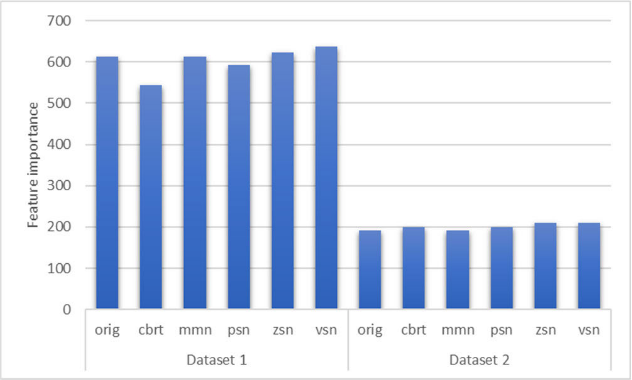

FIGURE 3. Comparison of feature importance scores across single normalization methods on raw forms of datasets 1 and 2.

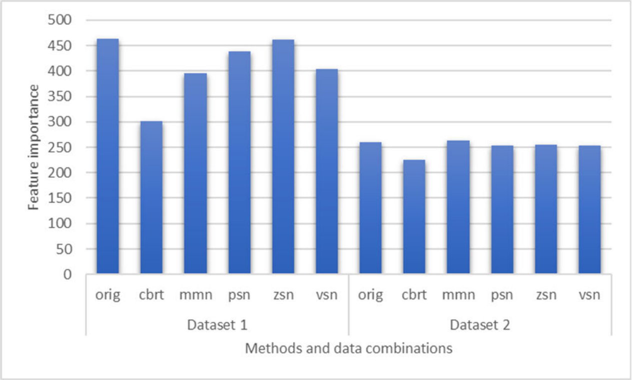

FIGURE 4. Comparison of feature importance of single normalization methods on datasets 1 and 2 after augmentation.

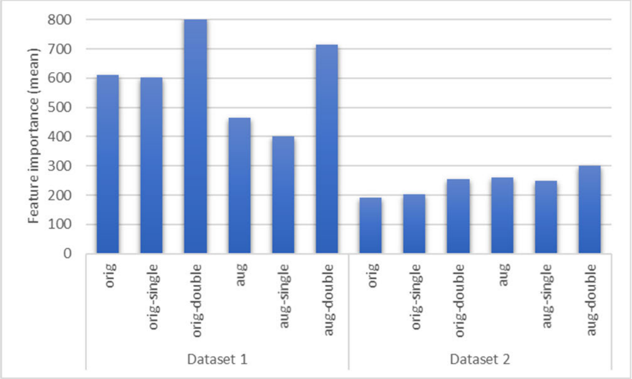

FIGURE 5. Comparison of mean group feature importance scores of single and combined normalization methods on augmented and nonaugmented versions of datasets 1 and 2.

The receiver operating characteristic curve (ROC) is described as a curve drawn on a 2D plane, where the abscissa represents the FPR and the ordinate represents the TPR. The AUC is defined as the area under the ROC curve. A high AUC score is desirable in a model.

B. PERFORMANCE ANALYSIS

This subsection analyzes the model's performance based on the single and combined normalization methods of datasets 1 and 2 with and without augmentation. We also discuss the model's performance by considering the single and combined normalization methods on augmented and nonaugmented data.

1) EFFECTS OF THE METHODS AND THEIR COMBINATIONS ON THE DATASETS

In this section, we used feature importance and variability to evaluate the effect of the methods described in section III on the two datasets. Feature importance scores on the nonaugmented datasets (orig) that have been preprocessed by single normalization methods are shown in Fig. 3.

Fig. 3 shows that single normalization methods produce more important features as compared to raw data. However, the cbrt method seems to produce fewer important features than non-normalized data in dataset 1, which is unexpected. Normalization methods produce more important features because appropriate normalization can shift the feature space by computations such as shrinking dominant features while improving the less dominant ones. Nevertheless, this is not always the case as can be observed in dataset 1. Fig. 4 shows a comparison between feature importance scores of augmented non-normalized data (denoted by vae) and single normalization methods applied to the same data.

In Fig. 4, it can be observed that single normalization methods did not significantly affect the feature importance of the augmented data and, in some cases, led to fewer important features. This poor effect of normalization methods can be attributed to the fact that the augmentation may have skewed the data in a direction that is not favorable to the normalization methods. Furthermore, the mean feature importance scores of single and combined normalization methods were used to evaluate the collective performances of the respective groups on augmented (aug) and nonaugmented versions of datasets 1 and 2 as shown in Fig. 5.

Fig. 5 shows that as opposed to single normalization methods, combined methods produce more important features on nonaugmented datasets. This is expected as the combined normalization methods can increase the feature space in a dataset, which may consequently increase the number of important features. Whereas augmentation produces varying effects with respect to feature importance across the two datasets, it can also be observed that combined normalization methods generally produce more important features on both nonaugmented and augmented data. Therefore, using combined normalization methods in addition to data augmentation, can improve feature importance in a dataset.

Similarly, the effect of single and combined normalization methods on the variability of augmented and nonaugmented versions of datasets 1 and 2 was evaluated using standard deviation. Fig. 6 shows the variability scores of single normalization methods on both datasets before augmentation.

From Fig. 6, it can be observed that of the two non-normalized datasets, dataset 2 has the highest variability. Fig. 6 also shows that PSN has the second highest variability which indicates that the method could be the worst among

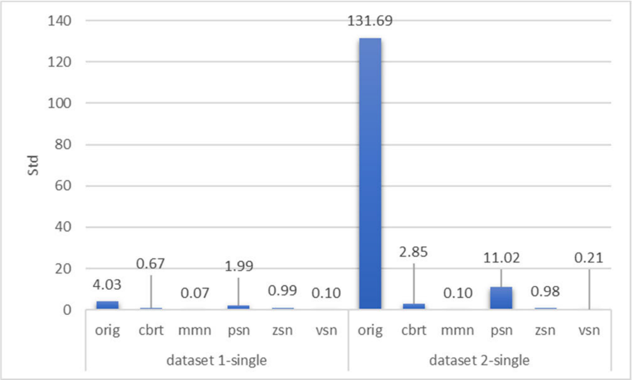

FIGURE 6. Comparison of variability scores of single normalization methods on dataset 1 and 2 before augmentation.

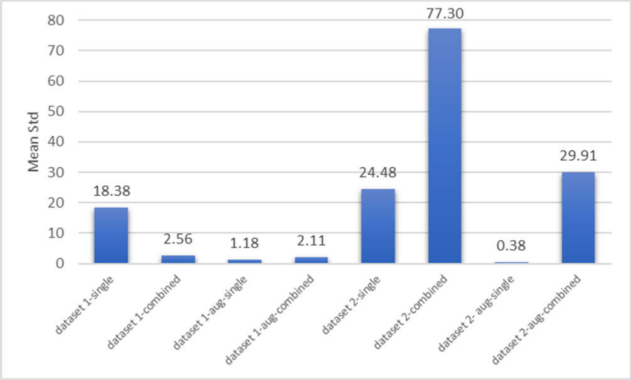

FIGURE 8. Comparison of mean group variability scores of single and combined normalization methods on augmented and nonaugmented versions of datasets 1 and 2.

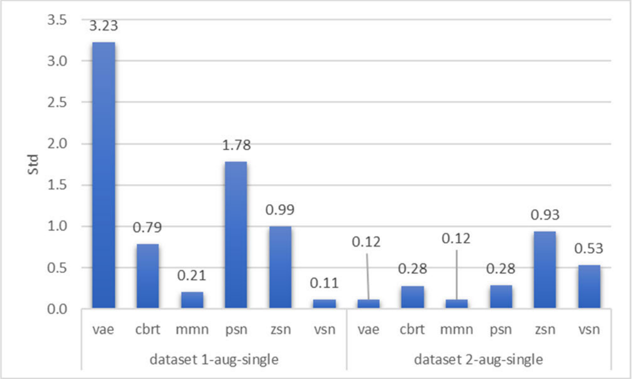

FIGURE 7. Comparison of Variability scores of single normalization methods on datasets 1 and 2 after augmentation.

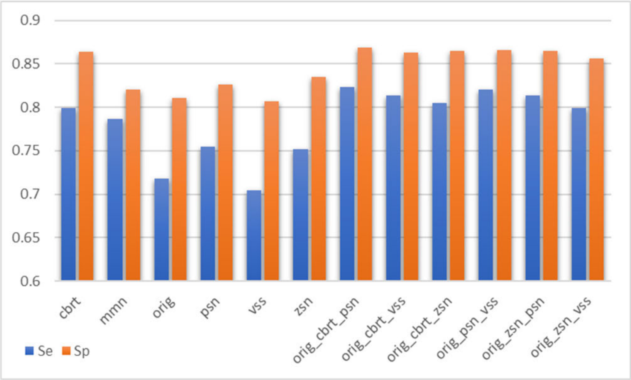

FIGURE 9. Comparison of sensitivity and specificity scores of normalization methods on nonaugmented dataset 1.

the normalization methods used in the experiments on this data. Fig. 7 shows variability scores of single normalization methods on augmented dataset 1 and 2.

Based on the non-normalized dataset in Fig. 6 and 7, it can be observed that augmentation reduces variability. Even though all normalization methods further reduce variability scores on augmented dataset 1, none of the same methods outperformed non-normalized augmented dataset 2. Here also we can assume that augmentation skewed dataset 1 in the direction which favors the normalization methods, whereas the skew on dataset 2 did not favor the normalization methods. Hence, the observed difference. Mean variability scores of single and combined normalization methods were used to evaluate the collective performances of the respective groups on augmented and nonaugmented versions of datasets 1 and 2 as shown in Fig. 8.

Fig. 8 shows that although normalization methods do not produce identical behavior across the two datasets, the combined normalization methods when applied to augmented data, can have better variability than the corresponding methods on nonaugmented data. However, it can also be observed that in the case of augmented data, combined methods produce higher variability than single normalization methods. This is because a the combined normalization methods used in this experiment are meant to produce a dataset that has a higher dimension than the input data. Finally, it also shows that combined normalization methods on nonaugmented dataset 2 have higher variability than the corresponding methods in dataset 1.

2) EFFECTS OF THE METHODS AND THEIR COMBINATIONS ON DNN PERFORMANCE

Next we show how the application of the single and combined normalization methods on both augmented and nonaugmented data affects the performance of a DNN model. The AUC, sensitivity (Se) and specificity (Sp) scores were used to evaluate the performance of the model with respect to the methods used in the experiment. A comparison of mean sensitivity and specificity scores of single and combined methods on nonaugmented dataset 1 are shown in Fig. 9.

Fig. 9 shows that combined normalization methods generally have better sensitivity and specificity on nonaugmented data than single methods. The methods have better performance because they can increase the number of important features and reduce variability, as observed in Fig. 5 and 8. High variability can greatly reduce the statistical power of ML algorithms and can lead to an increase in false positives [40]. The combination of the orig, cbrt and psn methods had

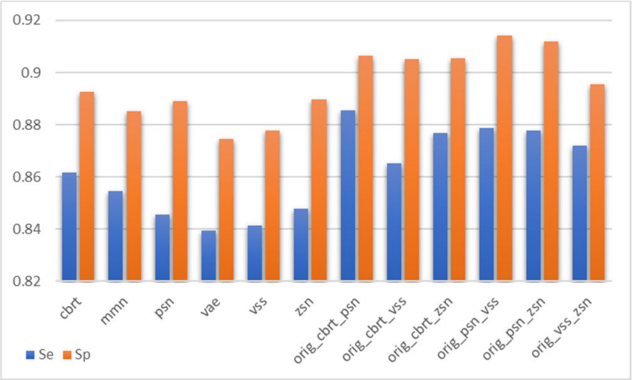

FIGURE 10. **Comparison of sensitivity and specificity scores of normalization methods on augmented dataset 1.**

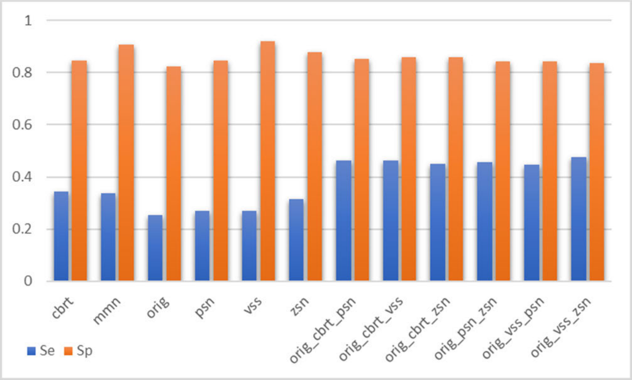

FIGURE 11. **Comparison of sensitivity and specificity scores of normalization methods on nonaugmented dataset 2.**

the highest specificity of 86.9% which was followed by that of orig, psn and vss methods that had a specificity of 86.6%. Following in third place, was a single normalization method cbrt which had a specificity of 86.4% that is higher than other single normalization methods. Though sensitivity was generally low across all the methods on this data, specificity had a similar pattern. The combination of the orig, cbrt and psn; orig, psn and vss methods; and a single normalization method cbrt obtained the top three sensitivity scores of 82.4%, 82%, and 79.9%, respectively. The mean sensitivity and specificity scores of single and combined methods on augmented dataset 1 are shown in Fig. 10.

From Fig. 10, it can be observed that sensitivity and specificity scores of combined normalization methods on augmented dataset 1 are comparatively higher than those of single methods. The difference in performance is because of the combined normalization methods producing a higher number of important features. This is despite the methods having a slightly higher variability than single normalization methods. The combined normalization methods also have better performance on augmented data than corresponding methods on nonaugmented data, because they have higher feature importance and lower variability. The same can be said about single normalization methods on augmented, in contrast to those on the nonaugment datasets. The top three specificity scores of 91.4%, 91.2% and 90.6% were obtained by combining orig, psn and vss; orig, psn and zsn; and orig, cbrt and psn methods respectively. The best three sensitivity scores of 88.6%, 87.9% and 87.8% were obtained by combining the orig cbrt and psn; orig, psn and vss; and orig, psn and zsn methods respectively. Similarly, mean sensitivity and specificity scores of single and combined methods on nonaugmented dataset 2 are shown in Fig. 11.

Fig. 11 shows that single normalization methods have relatively higher specificity than the combined approach on the nonaugmented data. This is expected as the combined methods have a higher mean variability score on this data. Also, to note is the extremely low sensitivity across all methods, which can be attributed to the disproportionately lower number of CRC than non-CRC samples in the dataset.

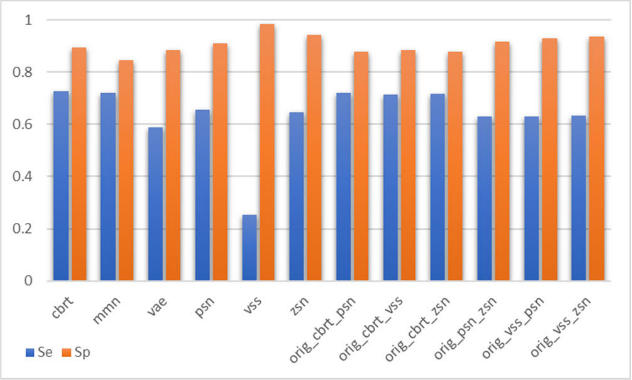

FIGURE 12. **Comparison of sensitivity and specificity scores of normalization methods on augmented dataset 2.**

However, combined methods seem to have comparatively higher sensitivity scores than single methods on this data. This improvement can be attributed to the higher number of important features associated with the combined normalization methods. Improving feature importance of less dominant features may have also slightly improved the influence of features that represent CRC. The top three specificity scores of 92.1%, 90.6% and 87.9%, were obtained by the single normalization methods, namely vss, mmn and zsn respectively. However, the best three sensitivity scores obtained by the combination of orig, vss and zsn; orig, cbrt and psn; and orig, cbrt, and zsn methods were 47.5%, 46.4% and 46.4%, respectively. The performance in terms of sensitivity and specificity of single and combined methods on augmented dataset 2 are shown in Fig. 12.

Fig. 12 shows that single normalization methods on augmented dataset 2 have the highest specificity and sensitivity than combined methods. These results are expected because combined methods in contrast to single normalization methods have a slightly fewer number of important features and a high variability. The highest three specificity scores of 98.5%, 94.1% and 93.4%, on this data, were obtained by the single normalization methods vss, zsn and a combination of orig, vss and zsn, respectively. The best three sensitivity scores of 72.7%, 72.2% and 72%, were obtained by cbrt, mmn and

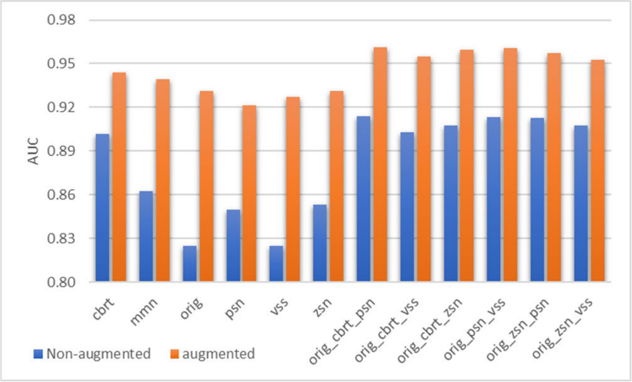

FIGURE 13. Comparison of mean AUC scores of normalization methods on dataset 1.

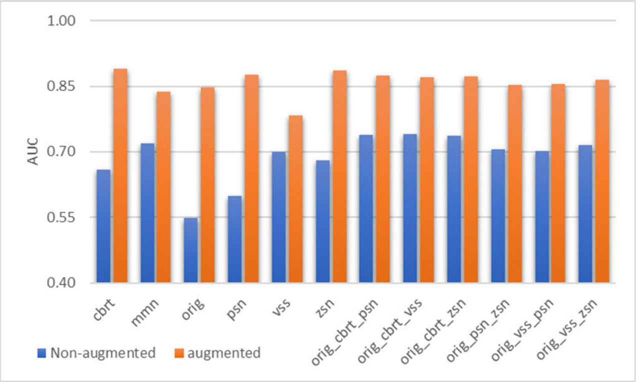

FIGURE 14. Comparison of mean AUC scores of normalization methods on dataset 2.

a combination of orig, cbrt and psn methods, respectively. Though the top methods in terms of specificity and sensitivity are single normalization methods, it can be observed that combined methods have a better average performance in both metrics.

The performance of the methods on both the augmented and nonaugmented datasets was also compared using AUC scores. AUC values of both single and combined normalization methods on augmented and nonaugmented dataset 1 are shown in Fig. 13.

From Fig. 13 it can be observed that the augmented dataset generally has a higher performance. The results also show that using combined normalization methods, in addition to augmentation, produces better results on this data. The highest mean AUC scores for nonaugmented single, nonaugmented combined, augmented single and augmented combined normalization methods on dataset 1 are 0.90, 0.91, 0.94 and 0.96, respectively. It is also noteworthy that of all single normalization methods, the performance of the cbrt method is among the highest in augmented and has performed relatively well on nonaugmented data. Similarly, AUC scores of both single and combined normalization methods on augmented and nonaugmented dataset 2 are shown in Fig. 14.

Fig. 14 shows a similar pattern to that of Fig. 13, where all normalization methods perform better when applied on augmented than on nonaugmented data. It also confirms that combined normalization methods are more robust when applied together with augmentation. The best mean AUC scores for nonaugmented single, nonaugmented combined, augmented single and augmented combined normalization methods on dataset 2 are 0.70, 0.74, 0.89 and 0.89, respectively. Although the cbrt method does not outperform other single normalization methods in nonaugmented data, its performance is superior in augmented data. The method has approximately the same performance as the best combined normalization method in augmented data.

TABLE 2. Post hoc significance test results of the method combinations using dataset 1.

	Comparison of method combinations	p-value
1	Shotgun_augmentation_combined vs Shotgun_augmentation_single	< 0.001
2	Shotgun_augmentation_combined vs Shotgun_basic_combined	< 0.001
3	Shotgun_augmentation_single vs Shotgun_basic_combined	< 0.001
4	Shotgun_augmentation_combined vs Shotgun_basic_single	< 0.001
5	Shotgun_augmentation_single vs Shotgun_basic_single	< 0.001
6	Shotgun_basic_combined vs Shotgun_basic_single	< 0.001

3) SIGNIFICANCE TESTING

The difference in performance of the DNN model across the method combinations used in the experiment was tested for significance using AUC scores obtained by the methods. Although all samples had 14 items, not all of them had a normal distribution. Owing to this discrepancy in sample distribution and their small size, a non-parametric test is the appropriate method to use for significance testing. In this significant test, method combinations and AUC scores relate to categorical target and continuous predictor variables, respectively. Therefore, Kruskal Wallis was chosen as the suitable method for use in group-wise significance testing, and Dunny's test was used for post-hoc testing. The tests were two-tailed, and 0.05 was used as the confidence interval. In order to reduce family-wise error, significance testing was done separately on the methods on datasets 1 and 2, respectively. The group-wise p-values on both dataset 1 and 2 were less than 0.001. The results for pair-wise significance testing for the method combinations using dataset 1 are shown in Table 2.

From Table 2, it can be observed that the difference in performance between method combinations is significant. Table 3 shows the results for pair-wise significance testing of the method combinations using dataset 2.

From Table 2, it can be observed that there is no significant difference in performance between the combined and single normalization methods on augmented dataset 2. However, the rest of the method combinations show a substantial difference

TABLE 3. Post hoc significance test results of the method combinations using dataset 2.

	Comparison of method combinations	p-value
1	16srrna_augmentation_combined vs 16srrna_augmentation_single	1.00
2	16srrna_augmentation_combined vs 16srrna_basic_combined	< 0.001
3	16srrna_augmentation_single vs 16srrna_basic_ combined	< 0.001
4	16srrna_augmentation_double vs 16srrna_basic_single	< 0.001
5	16srrna_augmentation_single vs 16srrna_basic_single	< 0.001
6	16srrna_basic_double vs 16srrna_basic_single	< 0.001

TABLE 4. Summary of AUC performance comparisons of the methods.

	AUC	
Method	**Dataset 1**	**Dataset 2**
nonaugmented	0.83	0.55
nonaugmented single	0.90	0.70
nonaugmented combined	0.91	0.74
augmented	0.93	0.85
augmented single	0.94	0.89
augmented combined	0.96	0.89

in their performance. The results show that combined normalization methods when used in collaboration with data augmentation produce a model that has a performance which is significantly higher than a model that only depends on data augmentation or normalization.

4) COMPARING THE PROPOSED METHOD WITH BASELINE METHODS

In section III the proposed model was described and its main components highlighted. In order to create a basis upon which to benchmark the proposed model alternative combinations of the components were also highlighted. AUC performance scores of the alternative component combinations and the proposed model (using data augmentation and combined normalization methods) are shown in Table 4.

Table 4 shows that whereas single normalization outperforms non-normalized data, combined normalization performs better than single normalization in both augmented and nonaugmented data. It can also be observed that whereas the DNN model has the lowest performance on nonaugmented data across the two datasets, the model has the highest performance when augmentation is used together with the combined normalization methods.

Analysis of the normalization methods has shown that even though the proposed cbrt method does not outperform other single normalization methods all the time, the normalization method produces outstanding results when combined with augmentation. The results have also shown that while the performance of a normalization method may depend on the number of relevant features produced and how it modifies variability in a dataset, it may also be influenced by other factors such as the quality of relevant features. This research has further shown that aggregation of normalization methods can improve the performance of a DNN model. The results have also proven that using data augmentation in addition to combined normalization methods can further improve the performance of a model. This combination of methods produces outstanding results by capitalizing on the reduced data variability due to augmentation, and improved feature importance due to the feature extension mechanism that combines normalization methods.

V. CONCLUSION

We have empirically demonstrated that aggregation of data normalization methods can significantly improve classification performance of a DNN model. The proposed method leveraged the strength of combined normalization methods in order to augment features, which in turn improved the feature relevance of less dominant features in the dataset. This outcome of this study indicates that combining the proposed feature extension method with data augmentation produced a more robust model than solely using data augmentation or normalization. The use of mixed normalization methods and data augmentation obtained AUC scores of about 0.96 and 0.89 in datasets 1 and 2. The combination of normalization in contrast to individual methods also improves model sensitivity even when data is highly imbalanced with more negative than positive cases.

In future works, we intend to investigate how the proposed method would perform in the context of DNN ensemble methods. Automatic aggregation of normalization methods based on input data properties would also be an exciting topic for future consideration.

REFERENCES

[1] P. Favoriti, G. Carbone, M. Greco, F. Pirozzi, R. E. M. Pirozzi, and F. Corcione, "Worldwide burden of colorectal cancer: A review," *Updates Surg.*, vol. 68, no. 1, pp. 7–11, Mar. 2016.

[2] J. P. Zackular, M. A. M. Rogers, M. T. Ruffin, and P. D. Schloss, "The human gut microbiome as a screening tool for colorectal cancer," *Cancer Prevention Res.*, vol. 7, no. 11, pp. 1112–1121, Nov. 2014.

[3] A. García-Bilbao, R. Armañanzas, Z. Ispizua, B. Calvo, A. Alonso-Varona, I. Inza, P. Larrañaga, G. López-Vivanco, B. Suárez-Merino, and M. Betanzos, "Identification of a biomarker panel for colorectal cancer diagnosis," *BMC Cancer*, vol. 12, no. 1, p. 43, Dec. 2012.

[4] J. L. Drewes, J. R. White, C. M. Dejea, P. Fathi, T. Iyadorai, J. Vadivelu, A. C. Roslani, E. C. Wick, E. F. Mongodin, M. F. Loke, K. Thulasi, H. M. Gan, K. L. Goh, H. Y. Chong, S. Kumar, J. W. Wanyiri, and C. L. Sears, "Author correction: High-resolution bacterial 16S rRNA gene profile meta-analysis and biofilm status reveal common colorectal cancer consortia," *NPJ Biofilms Microbiomes*, vol. 5, no. 1, p. 34, Dec. 2019.

[5] S. Tarallo, G. Ferrero, G. Gallo, A. Francavilla, G. Clerico, A. Realis Luc, P. Manghi, A. M. Thomas, P. Vineis, N. Segata, B. Pardini, A. Naccarati, and F. Cordero, "Altered fecal small RNA profiles in colorectal cancer reflect gut microbiome composition in stool samples," *mSystems*, vol. 4, no. 5, pp. 1–16, Sep. 2019.

[6] J. Wirbel *et al.*, "Meta-analysis of fecal metagenomes reveals global microbial signatures that are specific for colorectal cancer," *Nature Med.*, vol. 25, no. 4, pp. 679–689, Apr. 2019.

[7] A. M. Thomas *et al.*, "Metagenomic analysis of colorectal cancer datasets identifies cross-cohort microbial diagnostic signatures and a link with choline degradation," *Nature Med.*, vol. 25, no. 4, pp. 667–678, Apr. 2019.

[8] S. M. Anwar, M. Majid, A. Qayyum, M. Awais, M. Alnowami, and M. K. Khan, "Medical image analysis using convolutional neural networks: A review," *J. Med. Syst.*, vol. 42, no. 11, pp. 1–13, Nov. 2018.

[9] M. Oudah and A. Henschel, "Taxonomy-aware feature engineering for microbiome classification," *BMC Bioinf.*, vol. 19, no. 1, pp. 1–3, Dec. 2018.

[10] B. D. Topçuoğlu, N. A. Lesniak, M. T. Ruffin, J. Wiens, and P. D. Schloss, "A framework for effective application of machine learning to microbiome-based classification problems," *MBio*, vol. 11, no. 3, pp. 1–13, 2020.

[11] Q. Zhu *et al.*, "An ensemble feature selection method based on deep forest for microbiome-wide association studies," in *Proc. IEEE Int. Conf. Bioinform. Biomed. (BIBM)*, 2019, pp. 248–253.

[12] A. Arabameri, D. Asemani, and P. Teymourpour, "Detection of colorectal carcinoma based on microbiota analysis using generalized regression neural networks and nonlinear feature selection," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, p. 1, 2018.

[13] N. Simidjievski *et al.*, "Variational autoencoders for cancer data integration: Design principles and computational practice," *Frontiers Genet.*, vol. 10, pp. 1–14, Dec. 2019.

[14] N. T. Baxter, M. T. Ruffin, M. A. M. Rogers, and P. D. Schloss, "Microbiota-based model improves the sensitivity of fecal immunochemical test for detecting colonic lesions," *Genome Med.*, vol. 8, no. 1, pp. 1–10, 2016.

[15] A. Kishk *et al.*, "A hybrid machine learning approach for the phenotypic classification of metagenomic colon cancer reads based on kmer frequency and biomarker profiling," in *Proc. 9th Cairo Int. Biomed. Eng. Conf. (CIBEC)*, 2019, pp. 118–121.

[16] E. Pasolli, D. T. Truong, F. Malik, L. Waldron, and N. Segata, "Machine learning meta-analysis of large metagenomic datasets: Tools and biological insights," *PLoS Comput. Biol.*, vol. 12, no. 7, pp. 1–26, 2016.

[17] J. Yu *et al.*, "Metagenomic analysis of faecal microbiome as a tool towards targeted non-invasive biomarkers for colorectal cancer," *Gut*, vol. 66, no. 1, pp. 70–78, 2017.

[18] Z. Dai *et al.*, "Multi-cohort analysis of colorectal cancer metagenome identified altered bacteria across populations and universal bacterial markers," *Microbiome*, vol. 6, no. 1, p. 70, 2018.

[19] B. Flemer *et al.*, "The oral microbiota in colorectal cancer is distinctive and predictive," *Gut*, vol. 67, no. 8, pp. 1454–1463, 2018.

[20] A. Eetemadi, N. Rai, B. M. P. Pereira, M. Kim, H. Schmitz, and I. Tagkopoulos, "The computational diet: A review of computational methods across diet, microbiome, and health," *Frontiers Microbiol.*, vol. 11, pp. 1–22, Apr. 2020.

[21] C. L. Chowdhary, G. V. K. Sai, and D. P. Acharjya, "Decreasing false assumption for improved breast cancer detection," *J. Sci. Arts Year*, vol. 16, no. 2, pp. 157–176, 2016.

[22] C. L. Chowdhary and D. P. Acharjya, "Segmentation and feature extraction in medical imaging: A systematic review," *Procedia Comput. Sci.*, vol. 167, no. 2019, pp. 26–36, 2020.

[23] D. Beck and J. A. Foster, "Machine learning techniques accurately classify microbial communities by bacterial vaginosis characteristics," *PLoS ONE*, vol. 9, no. 2, 2014.

[24] J. Chaki and N. Dey, "Pattern analysis of genetics and genomics: A survey of the state-of-art," *Multimedia Tools Appl.*, 2019.

[25] H. Sajedi, F. Mohammadipanah, and S. A. H. Rahimi, "Actinobacterial strains recognition by machine learning methods," *Multimedia Tools Appl.*, vol. 78, no. 14, pp. 20285–20307, 2019.

[26] G. Renda *et al.*, "Deep learning models for bacteria taxonomic classification of metagenomic data," *BMC Bioinform.*, vol. 19, no. Suppl 7, 2018.

[27] A. Sharma, E. Vans, D. Shigemizu, K. A. Boroevich, and T. Tsunoda, "DeepInsight: A methodology to transform a non-image data to an image for convolution neural network architecture," *Sci. Rep.*, vol. 9, no. 1, pp. 1–7, 2019.

[28] C. Lo and R. Marculescu, "MetaNN: Accurate classification of host phenotypes from metagenomic data using neural networks," in *Proc. ACM Int. Conf. Bioinform., Comput. Biol. Health Inform. (ACM-BCB)*, vol. 20, no. Suppl 12, 2018, pp. 608–609.

[29] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Appl. Soft Comput. J.*, vol. 97, no. 105524, 2019.

[30] E. Sayyari, B. Kawas, and S. Mirarab, "TADA: Phylogenetic augmentation of microbiome samples enhances phenotype classification," *Bioinformatics*, vol. 35, no. 14, pp. i31–i40, 2019.

[31] V. Jonsson, T. Österlund, O. Nerman, and E. Kristiansson, "Variability in metagenomic count data and its influence on the identification of differentially abundant genes," *J. Comput. Biol.*, vol. 24, no. 4, pp. 311–326, 2017.

[32] F. Chollet. (2015). Keras. Github. Accessed: Aug. 13, 2019. [Online]. Available: https://github.com/fchollet/keras

[33] A. P. J. Mcmurdie, S. Holmes, G. Jordan, and S. Chamberlain, "Package 'phyloseq,'" 2019.

[34] J. Namkung, "Machine learning methods for microbiome studies," *J. Microbiol.*, vol. 58, no. 3, pp. 206–216, Mar. 2020.

[35] E. Ogasawara, L. C. Martinez, D. De Oliveira, G. Zimbrão, G. L. Pappa, and M. Mattoso, "Adaptive normalization: A novel data normalization approach for non-stationary time series," in *Proc. Int. Joint Conf. Neural Netw.*, 2010.

[36] I. Noda, "Scaling techniques to enhance two-dimensional correlation spectra," *J. Mol. Struct.*, vols. 883–884, nos. 1–3, pp. 216–227, 2008.

[37] F. Zou, L. Shen, Z. Jie, W. Zhang, and W. Liu, "A sufficient condition for convergences of Adam and RMSProp," no. 1, pp. 11127–11135, 2018.

[38] C. Garbin, X. Zhu, and O. Marques, "Dropout vs. batch normalization: An empirical study of their impact to deep learning," *Multimedia Tools Appl.*, 2020.

[39] R. Wei, C. Garcia, A. El-Sayed, V. Peterson, and A. Mahmood, "Variations in variational autoencoders—A comparative evaluation," *IEEE Access*, vol. 8, pp. 153651–153670, 2020.

[40] M. B. Pereira, M. Wallroth, V. Jonsson, and E. Kristiansson, "Comparison of normalization methods for the analysis of metagenomic gene abundance data," *BMC Genomics*, vol. 19, no. 1, pp. 1–17, 2018.

MWENGE MULENGA received the B.Sc. and M.Sc. degrees in computer systems engineering from Saint Petersburg Electrotechnical University, Saint Petersburg, Russia, in 2007 and 2009, respectively. He is currently pursuing the Ph.D. degree with the Department of Artificial Intelligence, University of Malaya, Malaysia. He was specialized in software engineering. He has been a Lecturer of Computer Science with the School of Science Engineering and Technology, Mulungushi University, Kabwe, Zambia, since August 2010, where he has also been the Assistant Dean, from 2011 to 2018. He is also working as a Research Assistant with the Department of Artificial Intelligence, University of Malaya. His research interest includes colorectal cancer classification using gut microbiome and images.

SAMEEM ABDUL KAREEM received the B.Sc. degree (Hons.) in mathematics from the University of Malaya, in 1986, the M.Sc. degree in computing from the University of Wales, Cardiff (currently known as the University of Cardiff), in 1992, and the Ph.D. degree in 2002. She started her career as a Lecturer with the Institute of Preparatory Studies, ITM (currently known as UiTM), in 1986. She subsequently embarked on career at the University of Malaya, since 1993. She is currently a Professor with the Department of Artificial Intelligence, University of Malaya, Kuala Lumpur, where she is also the Dean of the Faculty of Computer Science and Information Technology. She was the Deputy Dean (Undergraduate) of the Faculty of Computer Science and Information Technology, University of Malaya, from 2001 to 2008, where she is also the Deputy Dean (Postgraduate), from 2014 to 2016. Her current research interests include artificial intelligence in medicine, machine learning, deep learning, data analytics/mining, image processing, and biomedical informatics. She has successfully supervised a number of Master and Ph.D. candidates and acted as an Internal and External Examiners both in and outside of Malaysia.

AZNUL QALID MD SABRI received the Erasmus Mundus Masters degree in Vision and Robotics (ViBot) and the joint master's degree from three different universities (University of Burgundy, France; University of Girona, Spain; and Heriot-Watt University, Edinburgh, U.K.), for which he performed in a Research Internship Program at the Commonwealth Scientific Research Organization (CSIRO), Brisbane, Australia, focusing on medical imaging. He is currently a Senior Lecturer with the Department of Artificial Intelligence, Faculty of Computer Science and Information Technology (FCSIT), University of Malaya, Malaysia. The Ph.D. degree (très honorable) on the topic of "Human Action Recognition," under a program jointly offered by a well-known research institution in France, Mines de Douai (a research lab) and the University of Picardie Jules Verne, Amiens, France. He is an active Researcher in the field of Artificial Intelligence, having published in multiple international conferences as well as international journals. His main research interests include the field of computer vision, robotics, and machine learning.

MANJEEVAN SEERA received the Ph.D. degree in computational intelligence from Universiti Sains Malaysia. He has over 15 years of experience in both academia and industry. He is currently an Associate Professor of Business Analytics with the School of Business, Monash University Malaysia. His research specialization is on machine learning principles and applications in finance and engineering. His recent research focus includes the design and development of advanced machine learning models for Fintech applications, particularly the detection and prediction of fraudulent financial transactions.

SURESH GOVIND is currently a Professor and the former Head of the Department of Parasitology, Faculty of Medicine, University of Malaya. For the past 30 years, he has published more than 150 scientific articles, presented more than 270 conference papers, and written several chapters in publications by World Health Organization. He has supervised more than 100 elective, diploma, graduate, and post-graduate students, including at Ph.D. level. He was responsible for placing Blastocystis sp., for the first time in the fact list of the WHO publication on the drinking water guidelines. He was the winner of the National Young Scientist Award, the Malaysian Toray Grant Award, the Commonwealth Scholarship Award, the Gold Innovative Award at the national level as well as the winner of Gold Medal at the International ITEX Exhibition. He has also won the Malaysian Society of Parasitology and Tropical Medicine Silver Medal, the Prime Minister's Productivity Award, and the prestigious Malaysian Toray Science Award. He has also won the University Malaya Excellence Award on several occasions. He was conferred the Global Malayalee Award for research, in 2015, and the National Educators Award from the Association of Private Institutions. In 2016, he won the Parija Oration Award from the Indian Academy of Tropical Parasitology, for outstanding contribution to the field of Parasitology. He was a Fellow of the Malaysian Academy of Science, in 2015, and the prestigious Sandosham Gold Medal for outstanding contribution to the field of Parasitology and Tropical Medicine.

CHANDRAMATHI SAMUDI is currently a Senior Lecturer with the Department of Medical Microbiology, University of Malaya. She has been teaching and supervising students in the field of microbiology, virology, and immunology. Her research mainly focuses on the association of intestinal microorganisms (bacteria, viruses, and intestinal parasites) with colorectal cancer, CRC. Till date, she has successfully demonstrated that an intestinal protozoan parasite, Blastocystis sp., could exacerbate CRC progression. In addition, her novel ideas coupled with enthusiasm to unravel the immunopathogenesis of microorganisms allow her to explore into research related to antimicrobials and host immune response against dengue. In collaboration with experienced professors and clinicians, she has multiple publications revealing interaction between microorganisms and host response mainly in colorectal cancer and infectious diseases. She has received a number of grants at both local and international levels. Her dedication in research has yielded over 40 publications in peer reviewed journals and 50 conference papers in both national and international conferences.

SAHARUDDIN BIN MOHAMAD received the B.Eng. (Bioengineering), M.Eng., and D.Eng. degrees form Tokushima University, Japan. Since 2004, he has been a Lecturer with the Bioinformatics Program, Institute of Biological Sciences, Faculty of Science, University of Malaya, where he is currently an Associate Professor with the Institute of Biological Sciences. He is actively involved in teaching and development of courses for B.Sc. (Bioinformatics) and Master of Bioinformatics students at the University of Malaya. He has been serving as the Head of Institute of Biological Sciences, Faculty of Science, University of Malaya, since 2020, and the Head of the Centre of Research in Systems Biology, Structural, Bioinformatics and Human Digital Imaging (CRYSTAL), University of Malaya, since 2017. He has been an Advisory Board Member of MyBioInfoNet (Malaysia Bioinformatics Network), since 2019. He was elected as the Vice President of the Malaysian Society of Bioinformatics and Computational Biology (MaSBiC) session, from 2018 to 2020.