Representation counts: the impact of embedding models on disease detection tasks from microbiome sequencing data

Mattia Strocchi*
Supervisors: Gabriele Corso†, Pietro Liò‡
EEMCS, Delft University of Technology, The Netherlands

June 19, 2022

A Dissertation Submitted to EEMCS faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering

*EEMCS, Delft University of Technology
†CSAIL, Massachusetts Institute of Technology
‡Department of Computer Science, University of Cambridge

Abstract

The human microbiome, the ensemble of microorganisms found in and on the human body, plays a key role in human health and disease. However, the current state of microbiome analysis represents a significant challenge for machine learning algorithms. Datasets of microbiome sequences are often characterized by a regime of large dimensionality and relatively few labels, making it difficult for a model to discriminate features from random noise and avoid overfitting. It is, therefore, paramount to reduce the dimensionality of the input data while preserving their structure and information for a model to properly learn from them. K-mer frequency vectors and learnable representations through encoders are some of the embedding methods that have been proposed in literature to reduce the dimensions of the input space for machine learning algorithms operating on biological sequences. This work aims to compare how various embedding techniques influence the performance of a downstream disease detection task from microbiome sequencing data. In particular, the research shows that k-mer frequency vectors lead to better classification metrics (AUC = 0.88) compared to NeuroSEED [1] embeddings (AUC = 0.76) on euclidean space. The work also presents how the classification problem formulation is critical to improving the overall disease detection performance.

1 Introduction

The human microbiome is the set of bacteria, archaea, fungi, protists, and viruses that inhabit our tissues and biofluids. An increasing number of scientific studies have shown how the microbiome plays a key role in human health and disease [2]. Its impact and mutation were studied across a plethora of diseases ranging from cancer to autism spectrum disorders (ASD) [3, 4, 5].

The exponentially decreasing cost of genome sequencing [6] and the rising number of large-scale open-source datasets [7, 8] is providing a solid ground for the application of machine learning (ML) in microbiome analysis, which has the potential to revolutionise the field of personalised medicine. However, microbiome data are characterised by large dimensionality and relatively few labels, making it complex for models to properly learn on it. This enhances the risk of overfitting as generalisable features are harder to discriminate from random noise when these features are extremely sparse in the input space. The *overfitting* problem leads to an overestimation of the accuracy of the model that performs poorly on unseen data samples.

A preliminary step common to most ML tasks in literature for microbiome data analysis is transforming the microbe DNA sequences into numerical vectors. In this research, the step is not regarded merely as a transformation from biological sequences to vectors: it also offers the opportunity to address the aforementioned dimensionality problem. *Embedding models* in literature have proposed various methods to construct embeddings from biological sequences that reduce the size of the input space for downstream machine learning tasks. Some of these use particularly meaningful (and interpretable) features, like the microbe abundance profiles [9]. Others use a simpler, alignment-free approach based on k-mer frequency vectors [10]. Finally, `NeuroSEED` uses encoders to learn low-dimensional geometric representations [1].

This project aims to evaluate the effectiveness of different *embedding models* through a binary classification (disease detection) task (Fig. 2). The dataset was picked based on the

number of samples, annotations available, and balance between cases and controls, without searching for a specific pathology. One of the largest, open-access datasets found is on Inflammatory Bowel Disease (IBD) [11], an umbrella term that describes disorders involving chronic inflammation of the gastrointestinal tract. Therefore, the task presented consists of distinguishing between patients with Crohn's (pathology of the IBD group) and controls. The performance of an *embedding model* is measured through the Area Under Curve (AUC) of the Receiver Operating Characteristic (ROC) of the classifier (Fig. 5). Each sample (patient) in the dataset is characterized by a variable number of sequences (microbes), with the cardinality spanning 10^2—10^6. A diagram of the percentage of samples against the cardinality is provided in Figure 1. The data-to-signal ratio is deemed particularly low as thousands of sequences spanning the order of 10^2 dimensions are associated to a single binary label. Considering that not all the embedding methods have the same overhead, a reflection on the performance metrics and the associated *embedding model* complexity is provided in the conclusions of this paper.

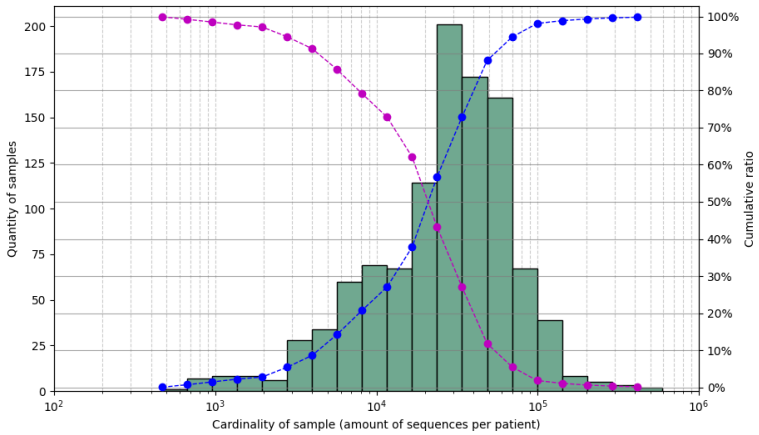

Figure 1: A histogram of the cardinality of the samples in the dataset. The plot is displayed on a logarithmic scale on the x-axis and linear for the y-axes. The blue line represents the cumulative ratio of samples and the purple line its complement. The graph shows that more than 70% of samples have at least 10^4 sequences and ∼99% have at least 10^3 sequences.

2 Methodology

The research was conducted on a microbiome dataset [11] from the Quiita database [7] of `1359` Inflammatory Bowel Disease (IBD) patients and controls. The sequences are raw reads of the 16S rRNA gene (specifically the V4 hypervariable region) of the microbes in the specimen. In this dataset, each of the samples is associated to one of the following classes: *inflammatory colitis*, *ulcerative colitis*, *Crohn's disease*, and *control*. This work focuses on distinguishing cases of Crohn's disease from controls, hence the samples associated with the other categories were dropped, leaving `1052` samples.

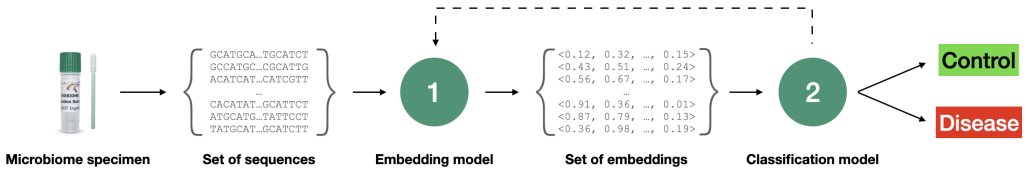

Figure 2: Diagram representing the data flow from the specimen to the output of the classification model. Each specimen corresponds to a sample taken from a patient. The DNA strings are referred to as "sequences" and the set of sequences as a dataset "sample". The set of sequences is produced from the microbiome specimen via targeted sequencing of the 16S gene (V4 region) of the microbes in the sample. The performance of the classification (2) are utilized to evaluate the various embedding models under test (1).

Data pipeline. The diagram in Fig. 2 provides an overview of the flow of data in this study. The sequencing machine takes a microbiome specimen and returns a set of sequences representing the raw reads of the 16S V4 region for the microbes in the sample. Each set of strings is then converted to a set of numerical vectors by using the embedding models under test. This process effectively creates a dataset for each embedding method. Afterwards, the binary classification model is trained on each embedding-dataset to output probabilities for the disease and control classes. Finally, the performance of the classification model, as measured by the AUC of the ROC curve, is utilized to establish which embedding technique performs best.

The following subsections will introduce the formulation of the classification and embedding problem presented in step `1` and `2` of Fig. 2.

2.1 Embedding problem formulation

In microbiome analysis literature, sequences are transformed into numerical vectors (step `1` of Fig. 2) prior to classification. Most of the techniques proposed to construct feature vectors involve either k-mer distributions or using encoder models. This study will review and compare a technique taken from each of the forementioned classes:

K-mer based embeddings. K-mers frequencies are one of the most widespread ways to encode DNA sequences into numerical vector representations. The process involves counting the occurrences of each unique k-length sub-string in the sequence and then normalizing the vector. The frequencies are computed by sliding a window of size k over the sequence, as Fig. 3 shows. Each DNA sequence is made up of four symbols (namely the DNA bases: `A`, `C`, `T`, `G`), thus the length of the k-mer frequency vector is 4^k. K-mers are fast to compute as they only take $\mathcal{O}(n)$ time, with n corresponding to the length of the sequence. Because of the limited RAM resources of the machine available, this work presents the results only for 3-mers and 4-mers frequency vectors.

Learning-based embeddings. Recently, Corso et al. introduced `NeuroSEED`, a learning-based framework to encode biological sequences into a low-dimension geometrical space (Fig. 3). The dataset of sequences used in the experiments referred in Fig. 4, 5 as 'NeuroSEED'

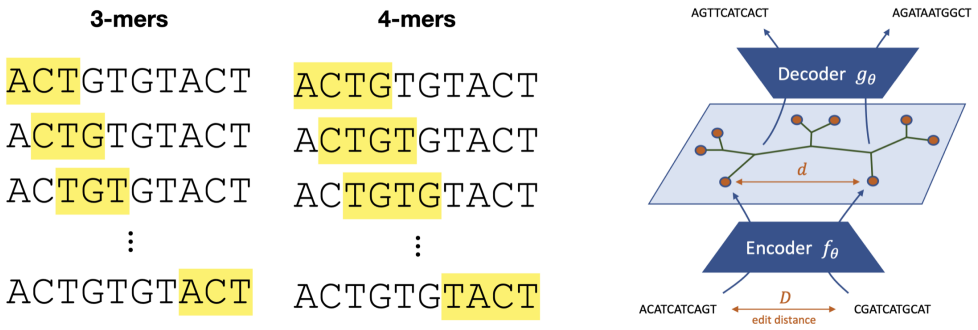

Figure 3: On the left, an illustration of 3-mers and 4-mers in a given sequence. k-mers frequency vectors are computed by sliding a k-sized window over the sequence, counting the occurrences of every possible k-length string and then normalizing the vector. On the right, a visualization of the NeuroSEED [1] framework, designed to learn and encoder function f_θ that preserves the edit distances between the sequence and vector space (D and d).

was encoded through a Convolutional Neural Network (CNN) on a 128-dimensioned euclidean space. The framework, however, supports encoding on an arbitrary target space. The geometry on which the embeddings lie is particularly relevant for the downstream machine learning model, as the classification performance obtained are bounded to the capability of the model to learn on that embedding space.

2.2 Classification problem formulation

The problem of disease detection from microbiome data (step 2 of Fig. 2) can be formulated as a binary classification task over sets of vectors. A few models were introduced in literature to work specifically on unordered collections of vectors, one of the firsts being `Deep Set` [12]. More models were later introduced, all deriving from the concepts presented by `Deep Set` [13, 14]. The attention mechanism implemented by transformers [15] is believed to be particularly useful for the classification task under study, as it enables the model to reason about interactions between parts of sequence embeddings. The research presented in this paper will both use `Deep Set` [12] and `Set Transformer` [14], a modified version of a transformer model without positional encoding.

One core difference of the proposed data pipeline from the work previously done in literature is the formulation of the classification problem. Most of the research papers available in literature [10, 9] for disease classification from microbiome define step 2 of Fig. 2 as a vector classification problem rather than classifying sets of vectors. This is carried out by either defining sample-wise features, such as microbe abundance profiles [9] or by aggregating the set of embeddings with a predetermined function. `MicroPheno` [10], which is an instance of the latter case, uses the mean of the vectors as aggregation function. Nonetheless, aggregating the sample's sequences with the mean prevents the downstream classification model to learn from potential interactions among the microbes in the sample, which are known to affect the final control/disease classification.

Hence, the advantage of formulating the problem as a classification of sets of vectors is preserving the original set structure of the data. Potentially, this translates to the capability to learn the complex interactions between the sequences (microbes), that could otherwise not be modelled if there was a single, aggregated vector per sample. Furthermore, by working on sets of sequences it is not necessary to come up with a function that aggregates all the vector representations of the sequences into a single one. The only disadvantage of using a model working on sets of vectors is the limited amount of pre-built models designed specifically for the task.

3 Experimental Work and Results

This section discusses the performance of each of the embedding models under test and evaluates the proposed *set classification* approach, comparing it to baseline models widely adopted in microbiome analysis literature. The section starts by explaining how the data were processed prior to running the classification task.

3.1 Dataset preprocessing

The final dataset comprises 333 controls and 719 cases and was subdivided in *train*, *validation* and *test* set following a 0.7, 0.15, 0.15 partition scheme. The `1052` dataset samples went through a pre-processing phase where sequences with ambiguous nucleotide bases[1] were dropped. Additionally, for each patient, a random sample of 1000 sequences was selected to keep the dataset small enough to be loaded in memory [2]. Crohn's disease cases were mapped to the label `1` and controls to `0`. Oversampling of the underrepresented label was applied to the training set to mitigate the effects of class imbalance. The logic was implemented through the `pytorch`'s `WeightedRandomSampler` that at each epoch would be used by the `DataLoader` to build balanced batches for training.

3.2 Embedding model evaluation

As mentioned in the introduction, the metric used to compare the performance of the classifiers is the area under curve of the receiver operating characteristic (Fig. 5). The improvements over time of the model are shown through the cross-entropy loss curves on the train and validation set (Fig. 4).

Loss curves. The *set classification* task was performed with two models: `Set Transformer` [14] and `Deep Sets` [12]. Regularization through *weight decay* was applied to reduce overfitting on all the training tasks performed. Nevertheless, the loss curves on k-mers embeddings (Fig. 4) show signs of overfitting, possibly due to the sub-sampling step described in 3.1. Specifically, the divergence between *validation* and *training* loss (3-mer and 4-mer graphs, Fig. 4) appeared to be influenced by which samples were distributed to *train*, *validation*, and *test* set, hinting that some dataset records might be more meaningful than others to learn generalized features from (an argumentation for the claim is provided in the Limitations section under the paragraph "Deviation of results"). Figure 4 shows how the classification

[1] Ambiguous nucleotide bases are reads where the sequencing machine could not determine the actual base.

[2] The training was performed on a compute instance provided by Google Colab Pro. The machine features 12.6 GB of RAM and an NVIDIA Tesla P100 (GPU RAM 16 GB).

models learn better from k-mers compared to NeuroSEED embeddings. Additionally, a higher k-value appears to improve the model's learning capacity, although this hypothesis should be thoroughly tested in future research by going above $k = 4$. The loss curve on NeuroSEED embeddings is particularly interesting, as it suggests that both `Deep Set` and `Set Transformer` cannot learn from the feature vectors generated by the framework on the euclidean space. Different results might be obtained by changing the classification model architecture or geometry of the space where NeuroSEED embeddings lie.

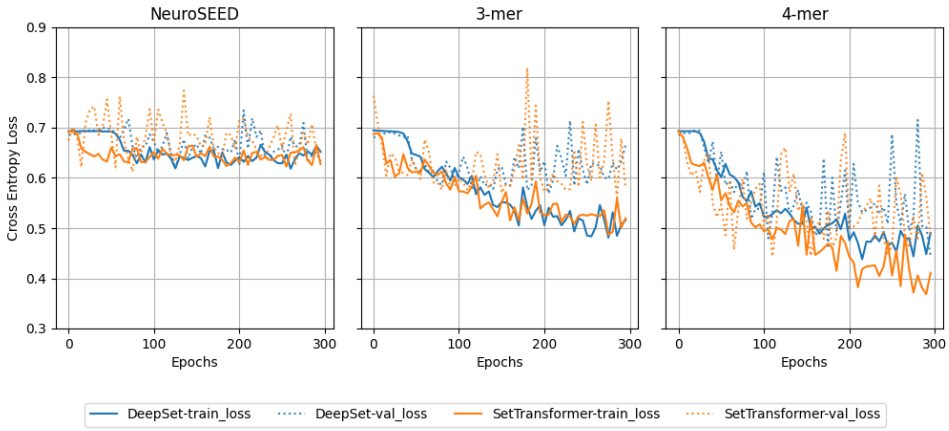

Figure 4: Cross-entropy loss for `Deep Sets` and `Set Transformer` on train and validation set as a function of the number of epochs. On the left, the plot shows the classification improvements on the NeuroSEED's embeddings, at the center the 3-mers and on the right 4-mers. The second and third figures from the left show signs of overfitting (divergence between training and validation loss), for which a possible explanation is reported in "Deviation of results" under Limitations.

Classification performance. The two *set classification* models, perform similarly on every embedding-dataset, suggesting that the macro-performance of the classification depends on the embedding method and formulation of the task (i.e *set classification* vs. *vector classification*) rather than on the model itself (Fig. 5). The hypothesis is confirmed by the average of the Area Under curve of ROC (AUROC) across all the tested models (Fig. 6), which yields a 0.715 for NeuroSEED, 0.758 for 3-mers, and 0.822 for 4-mers with only a marginal increase in standard deviation. The overall best classifier is `SetTransformer` on 4-mers, reaching an AUROC of 0.884 (Fig. 5, 6).

Baselines performance. With the aim of providing a reference for the performance of `Set Transformer` [14] and `Deep Sets` [12], ROC curves and their AUC are also provided for a set of baseline *vector classification* models. To turn the *set classification* problem into a *vector classification* problem, each set of sequences was reduced to a single one by using an element-wise mean as described in [10] and then normalized. In the *vector classification* task (i.e. on baseline models), the dataset was divided into train and test sets, with train accounting for 70% of the samples. The baselines analyzed in this research are Random

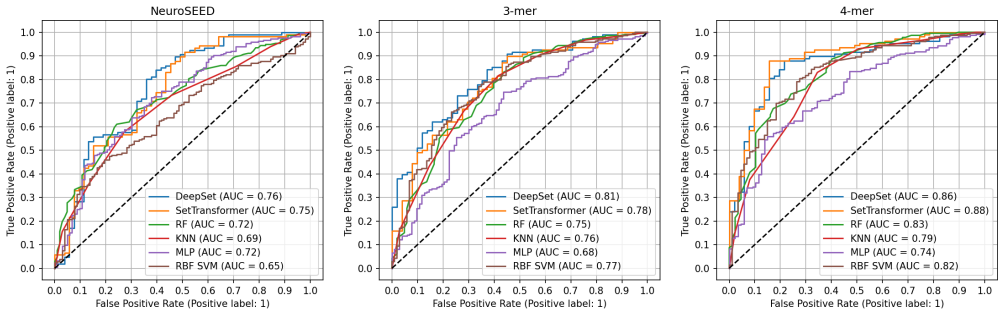

Figure 5: Receiver Operating Characteristic curves for models operating on *sets of vectors* and baselines operating on *vectors*. From left to right, the embeddings where generated with NeuroSEED, and 3-mer, and 4-mer frequency vectors. The AUROC scores are also presented table in 6 for easier interpretation.

Forest, K-Nearest Neighbours, Multi-Layer Perceptron, and a Support Vector Machine with Radial Basis Function kernel. Figure 5, provides a snapshot of the performance of all the models, where Random Forest is the best baseline for two out of three embedding methods. Overall, the models working on sets outperform the ones working on vectors for each embedding method (Fig. 6), proving that the proposed *set classification* approach is more effective than the widely-adopted *vector classification* from microbiome analysis literature.

4 Responsible Research

Reproducibility of the experiments was ensured by utilizing a publicly available dataset, sharing the processed embedding-dataset and creating a public GitHub repository for the full codebase. Furthermore, a minimal version of the code to train and evaluate an embedding method through the *set classification* task is provided via a Colab notebook. The link to the notebook is available below.

Further steps were also taken to mitigate the classification model's bias (oversampling, as described in 3.1) and overfitting (regularization through weight decay, as described in 3.2) to avoid overestimating the model's performance. Additionally, baseline results (section 3.2) were presented to give a clear picture of what can be considered the performance of the approach previously used in literature.

4.1 Dataset availability

The raw dataset is avaiable through the Quiita database [7]. The three processed embedding-dataset can be downloaded as a collection of numpy (`.npz`) archives at the following link.

4.2 Source code

The source code of the project is available in two forms: a GitHub repository and a Google Colab notebook. The repository contains the whole code and helper scripts used for the

project. The Colab notebook contains a minimal implementation to run the classification model training and evaluation. The notebook is designed to automatically download the dataset required. Running the code on a GPU instance will significantly speed up the training process.

5 Limitations

This paragraph introduces the limitations of the presented work. These can be seen as starting points for future research.

Scope of the research. Due to time constraints, the research could only be conducted on a single dataset of IBD patients. The performance obtained with this dataset might not be representative of the performance obtainable on a dataset of another pathology. Further research should be conducted on multiple datasets to understand the generalizability of the work presented.

Alignment-based embeddings. This study compared only alignment-free embedding methods. Another methodology adopted in literature to build feature vectors utilizes sequence alignment to construct Operational Taxonomic Units (OTU). This method is described by DeepMicro [9], which uses as input for the classifier microbe profiles encoded in a latent space through various encoder models. Their reported AUC for the best-performing model on a different IBD dataset is 0.95. Although this result might look impressive compared to the one presented in this work, it is hard to understand whether it is could be reproduced on a minimally different setup. The dataset they used contains only 110 samples, with the minority class (controls) accounting for a small 23% of the total amount of samples. The test set contains only 22 records (thus ∼5 controls) and the model's loss during training is computed on the validation set instead of the training set. No technique to mitigate the model bias was deployed. The authors do not disclose the mapping of the binary class to `0` and `1`, which is relevant to understanding how the ROC's True Positive and True Negative ratio are computed. As TP and TN ratios are particularly sensitive to class imbalance, the results presented are hard to interpret and compare.

Deviation of results. It is noteworthy that during training the difference between validation and train loss (overfitting, noticeable in Fig. 4) seemed to change based on how samples of the dataset were distributed between *train*, *validation* and *test* set. The underlying causes of this behaviour are most likely two: a (relatively) low amount of records in the dataset and sub-sampling of the sequences per patient. While the former is a constraint that characterizes most microbiome datasets, the latter could be improved. In this work, the random sub-sampling selected 10^3 sequences from every patient record due to the limited resources available (i.e. RAM and compute time). As Fig. 1 shows, more than 70% of patients have at least 10^4 sequences. For those records, the sub-sampling is getting at most a tenth of the actual amount of sequences. The question of whether picking a random sample of a tenth of the microbial population is sufficient to represent the original distribution is still unanswered. Future research should further quantify this aspect and show the variation in classification performance as the cardinality of the sub-sample approaches the actual amount of sequences of the patient.

6 Conclusions and Future Work

Section 3 discussed the experimental work, whereas this section aims at taking a more holistic approach, analyzing the insights and introducing future work.

Best embedding method. Overall, the best AUC was obtained on 4-mer embeddings running `Set Transformer` (AUC = 0.884), with `Deep Set` nearly matching the performance (AUC = 0.864), as shown by Fig. 6. At first thought, k-mers might not seem particularly meaningful features to provide a model for disease detection. Looking at the performance metrics, however, they yield reasonably good performances, especially if the signal-to-noise ratio is taken into account. This likely means that, as features, k-mers do encode useful information for disease detection tasks. For instance, it could be that certain microbes used as diagnostic biomarkers for Crohn's disease present a higher concentration of certain mutations. This would make a specific subset of k-mers a good predictor for the disease. Due to time constraints, the testing of NeuroSEED was limited to the embeddings generated in the euclidean space. In the future, using other geometrical spaces or model architectures might boost the classification performance on NeuroSEED embeddings. However, the picture is clear on the tested embedding models: k-mers offer easier-to-learn features compared to NeuroSEED (which leads to better classification metrics), with an interesting trend arising from the k-value. Higher k-values appear to improve the model's capability to learn, although for every k increment there is a four-fold increase in the dimensions of the embedding. Thus, the k-mer size increase is still bounded to low k-values. Future research should evaluate NeuroSEED embeddings on other geometries, test k-mers for higher k-values and provide a rigorous comparison with an alignment-based approach.

	AUROC		
Model	**NeuroSEED**	**3-mers**	**4-mers**
RF	0.721	0.755	0.826
KNN	0.688	0.757	0.792
MLP	0.718	0.678	0.743
RBF SVM	0.646	0.769	0.822
Set Transformer	0.751	0.779	0.884
Deep Set	0.765	0.814	0.864
Average	0.715	0.758	0.822
Std.	0.043	0.045	0.051

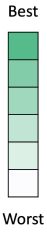

Figure 6: Area Under curve of ROC (AUROC) by classifier and embedding model. Displayed on yellow background the models working on sets of vectors, while on white background models working on vectors. Models working on sets show consistently higher performance compared to baselines.

Classification task formulation. Section 3 shows how the formulation of the classification problem influences the results of the classification itself. In particular the proposed *set classification* formulation, adopting models such as `Set Transformer` and `Deep Set`, was demonstrated to be more effective than the widely adopted *vector classification* in microbiome analysis literature. As mentioned in the previous paragraph, a rigorous comparison with alignment-based embedding methods would complete the current analysis, providing

an important insight to researchers working on disease detection from microbiome: *is it possible to achieve the same disease-detection performance with alignment-free embedding methods?* This question is particularly relevant for the field as the multiple sequences alignment problem is NP-complete. Being able to skip its computation would save time and resources.

In conclusion, this work presented how alignment-free embedding methods perform on a disease detection task from microbiome sequencing data. The best results were achieved by 4-mer embeddings, however, future research should investigate the influence of the subsampling step and the geometry of the space where the embeddings lie. As expected, formulating the problem as a *set classification* yields better performances compared to *vector classification*. Readers interested in reproducing the results can use the links provided in section 4 to download the dataset and codebase or simply run the notebook through Google Colab.

7 Acknowledgements

I thank Gabriele Corso and Pietro Liò for their insightful comments and guidance throughout the research and Sofia Di Giorgio, Jasmijn Baaijens and Benedetta Maizza for their review of the manuscript.

References

[1] G. Corso, R. Ying, M. Pándy, P. Veličković, J. Leskovec, and P. Liò, "Neural distance embeddings for biological sequences," 2021.

[2] F. H. Karlsson, V. Tremaroli, I. Nookaew, G. Bergström, C. J. Behre, B. Fagerberg, J. Nielsen, and F. Bäckhed, "Gut metagenome in european women with normal, impaired and diabetic glucose control," *Nature*, vol. 498, pp. 99–103, May 2013.

[3] B. Wang, M. Yao, L. Lv, Z. Ling, and L. Li, "The human microbiota in health and disease," *Engineering*, vol. 3, pp. 71–82, Feb. 2017.

[4] Y. Fan and O. Pedersen, "Gut microbiota in human metabolic health and disease," *Nature Reviews Microbiology*, vol. 19, pp. 55–71, Sept. 2020.

[5] J. Pulikkan, A. Mazumder, and T. Grace, "Role of the gut microbiome in autism spectrum disorders," in *Advances in Experimental Medicine and Biology*, pp. 253–269, Springer International Publishing, 2019.

[6] A. Sboner, X. Mu, D. Greenbaum, R. K. Auerbach, and M. B. Gerstein, "The real cost of sequencing: higher than you think!," *Genome Biology*, vol. 12, no. 8, p. 125, 2011.

[7] A. Gonzalez, J. A. Navas-Molina, T. Kosciolek, D. McDonald, Y. Vázquez-Baeza, G. Ackermann, J. DeReus, S. Janssen, A. D. Swafford, S. B. Orchanian, J. G. Sanders, J. Shorenstein, H. Holste, S. Petrus, A. Robbins-Pianka, C. J. Brislawn, M. Wang, J. R. Rideout, E. Bolyen, M. Dillon, J. G. Caporaso, P. C. Dorrestein, and R. Knight, "Qiita: rapid, web-enabled microbiome meta-analysis," *Nature Methods*, vol. 15, pp. 796–798, Oct. 2018.

[8] R. Leinonen, R. Akhtar, E. Birney, L. Bower, A. Cerdeno-Tarraga, Y. Cheng, I. Cleland, N. Faruque, N. Goodgame, R. Gibson, G. Hoad, M. Jang, N. Pakseresht, S. Plaister, R. Radhakrishnan, K. Reddy, S. Sobhany, P. T. Hoopen, R. Vaughan, V. Zalunin, and G. Cochrane, "The european nucleotide archive," *Nucleic Acids Research*, vol. 39, pp. D28–D31, Oct. 2010.

[9] M. Oh and L. Zhang, "DeepMicro: deep representation learning for disease prediction based on microbiome data," *Scientific Reports*, vol. 10, Apr. 2020.

[10] E. Asgari, K. Garakani, A. C. McHardy, and M. R. Mofrad, "Micropheno: predicting environments and host phenotypes from 16s rrna gene sequencing using a k-mer based representation of shallow sub-samples," *Bioinformatics*, vol. 34, no. 13, pp. i32–i42, 2018.

[11] D. Gevers, S. Kugathasan, L. A. Denson, Y. Vázquez-Baeza, W. V. Treuren, B. Ren, E. Schwager, D. Knights, S. J. Song, M. Yassour, X. C. Morgan, A. D. Kostic, C. Luo, A. González, D. McDonald, Y. Haberman, T. Walters, S. Baker, J. Rosh, M. Stephens, M. Heyman, J. Markowitz, R. Baldassano, A. Griffiths, F. Sylvester, D. Mack, S. Kim, W. Crandall, J. Hyams, C. Huttenhower, R. Knight, and R. J. Xavier, "The treatment-naive microbiome in new-onset crohn's disease," Mar. 2014.

[12] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. Salakhutdinov, and A. Smola, "Deep sets," 2017.

[13] Y. Zhang, J. Hare, and A. Prügel-Bennett, "Deep set prediction networks," 2019.

[14] J. Lee, Y. Lee, J. Kim, A. R. Kosiorek, S. Choi, and Y. W. Teh, "Set transformer: A framework for attention-based permutation-invariant neural networks," 2019.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.