RegMIL: Phenotype Classification from Metagenomic Data

Mohammad Arifur Rahman*
George Mason University
Fairfax, Virginia
mrahma23@gmu.edu

Huzefa Rangwala†
George Mason University
Fairfax, Virginia
rangwala@cs.gmu.edu

ABSTRACT

Advances in DNA sequencing technologies have paved the way for Metagenomics; defined as the collective sequencing of co-existing microbial communities in a host environment. Several researchers and clinicians have embarked on studying the role of these microbes with respect to human health and diseases, whereas others are using metagenomics to monitor the impact of external factors (e.g., Gulf oil spill) on ecosystems. Lack of accurate and efficient analytical methods is an impediment for the identification of the function and presence of microbial organism within different clinical samples, reducing our ability to elucidate the microbial-host interactions and discover novel therapeutics. In this paper, we present a Multiple Instance Learning (MIL) based computational pipeline to predict clinical phenotype from metagenomic data. We use data from well-known metagenomic studies of Liver Cirrhosis and Inflammatory Bowel Disease (IBD) to evaluate our approach. We show that our proposed pipeline outperforms comparative MIL as well as non-MIL methods.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning**; *Neural networks*; • **Applied computing** → **Bioinformatics**;

KEYWORDS

Multiple Instance Learning (MIL), Neural Network, Metagenomics, Phenotype Prediction

ACM Reference Format:

Mohammad Arifur Rahman and Huzefa Rangwala. 2018. RegMIL: Phenotype Classification from Metagenomic Data. In *ACM-BCB'18: 9th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics, August 29-September 1, 2018, Washington, DC, USA*. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3233547.3233585>

1 INTRODUCTION

Trillions of microbial cells in the human body are collectively referred to as the *human microbiome* [34][4]. The human microbiome

has come under investigation with the advancement of Next Generation Sequencing (NGS) technologies. Raw metagenomic data consists of a large number of short contiguous nucleotide subsequences called *reads* (of length 75 to 500 base pairs) [22]. Each of these reads is from different positions of the microbial genome. The large volume of short sequences from different organisms makes the analysis of metagenomic data challenging [14]. An additional level of complexity arises because the composition, abundance, diversity and genome lengths associated with these bacterial communities are unknown.

Researchers have discovered that microbes contribute to health conditions [4]. Metagenomics allows us to investigate the abundance of these microbes in a host collectively and analyze their interactions and effects on the host. Metagenomics also allows for discovery of novel microbial organisms. Most microbes have not been laboratory-cultured and thus remain unknown [12]. In a typical Metagenome-wide association study (MWAS), the output of NGS technologies from patient samples is millions of sequence reads. Associated with the patient may be demographic and pathological information such as health state, diet, age, sex, alcohol consumption and many more. This data is then analyzed to find associations between microbial communities and diseases such as obesity [33], neurological diseases [13], Crohn's disease [10], type-2 diabetes [27], liver cirrhosis [28] and others. Most of the studies utilize conventional supervised learning methods where metagenomic data is searched with respect to reference databases that contain gene sequences of known microbes. A patient sample with millions of reads is represented as a single vector which is used to train classification models. There are three key challenges with this approach:

- Contribution of a DNA sequence to a disease is not easily discovered. Microbiome creates a complex network in a host and subtle changes in their interactions can cause a shift from healthy to disease state [36].
- Discriminatory information from millions of sequences is not properly utilized because conventional supervised learning methods aggregate them as a single vector and use it for training a model.
- Use of reference genome database does not allow for discovery of novel microbial sequences from metagenomic data.

Within the Multiple Instance Learning (MIL) formulation we have several bags (or groups) of data instances. The labels of these bags are known but labels of instances in bags are unknown. In this case, we have a person (bag) and a label for each person (healthy or patient), but no labels for each sequence reads (data instances). MIL-based approaches overcome the aforementioned challenges of conventional supervised learning methods by first attempting to label each data instances then aggregate the instance labels to

*Graduate Student, Department of Computer Science

†Associate Professor, Department of Computer Science

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ACM-BCB'18, August 29-September 1, 2018, Washington, DC, USA

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-5794-4/18/08...\$15.00

<https://doi.org/10.1145/3233547.3233585>

classify the bags. In the rest of the paper, we refer to a bag as a collection of sequences pertaining to a sample (healthy or patient) and instance as a single sequence. We consider disease state as *positive* and healthy state as *negative*.

Most Multiple Instance Learning (MIL) approaches [7][3][6][2] do not scale well with large datasets. These approaches rely on computing kernels (similarities between all pairs of data instances). For N instances, the size of the kernel is $O(N^2)$, which makes it infeasible in our setting where N is more than 83 million (Section 4.1).

Our contributions in this paper are as follows:

- We develop a scalable MIL-based formulation for phenotype e.g., disease prediction from metagenomic data. We use a fast, hash-based clustering known as Canopy clustering to score instances in training set. These scores estimate the contribution of an instance (sequence) to the disease.
- We use the instance scores of the training set to train a two-layer neural network based regression model to score instances in the test set. This allows for fast scoring of the millions of instances within the test set during prediction phase.
- We create a histogram based bag-level feature representation by taking contributions of each instance to train a classifier e.g., random forest to accurately classify bags.
- Our proposed approach can predict the state of a person with high accuracy and ranks the sequences based on their respective contributions to the disease.

We evaluate our proposed pipeline on two large-scale metagenomic datasets of Liver Cirrhosis disease [28] and Inflammatory Bowel Disease (IBD) [26]. We compare the results of our method against popular MIL based classification approaches such as MISVM [3], sbMIL [6] and GICF [17] as well as non-MIL based classification approaches such as method [28] proposed by Qin et al. and MetAML [24]. We show that our proposed approach is efficient and scalable. We refer to our approach as Regression based Multiple Instance Learning (RegMIL). We have also released codes¹ for our pipeline under the open-source MIT license.

2 RELATED WORK

2.1 Metagenomics And Machine Learning

The effects of microbial diversity reduction on obesity have been investigated by Turnbaugh et al. [33]. Other diseases such as oral diseases [5], neurological diseases [13] and Crohn's disease [10] have found to be related to microorganisms. Saulnier et al. compared the gut microbiomes of healthy children and pediatric patients with irritable bowel syndrome (IBS) [31] and identified significant differences. Qin et al. used 345 gut microbial samples from type-2 diabetes (T2D) patients and healthy controls for statistical analysis and marker identification [27]. The authors used Maximum Relevance Minimum Redundancy (mRMR) feature selection to identify gene markers as features and Support Vector Machine (SVM) for type-2 diabetes classification. A similar approach was used by Qin et al. to indicate the differences in microbiota-targeted bio-markers between patients with liver cirrhosis disease and healthy controls

[28]. A total of 46,000 genes with the half enriched in liver cirrhosis patients and the other half in healthy controls were identified using a reference database [28] prior to disease prediction. Both of these approaches require extensive preprocessing of samples as well as the use of reference genome databases.

Another popular method is MetAML [24] which first identifies marker genes and species-level abundance using the MetaPhlAn2 [32] method and uses this output as features for classifiers. MetaPhlAn2 is a computational tool for profiling the composition of microbial communities from metagenomic shotgun sequencing data. The use of reference genome databases of known organisms for microbial abundance calculation prohibits identification of novel gene sequences that may contribute to the disease state.

We intend to use de-novo techniques without resorting to any reference genome database. Also, our proposed approach assigns each sequence within a sample; a score indicating its association with a disease regardless of the origin of that sequence.

2.2 Multiple Instance Learning (MIL)

Multiple Instance Learning (MIL) problem was first proposed by Dietterich for drug activity prediction [7]. In the drug activity prediction problem, molecules may assume many alternative low-energy states. But few of those states are suitable for drug discovery process. The problem involves i) predicting whether a new molecule is qualified to be a drug lead by analyzing a collection of known molecules and ii) if the molecule is a drug lead/candidate then, the approach predicts which of its low-energy shapes are responsible for that. Thus, in the original formation of MIL, a bag is classified as *positive* if one or more instances within it are positive, while a negative bag contains only negative instances. This is commonly referred to as the standard multiple instance assumption [2]. In the original paper, Dietterich developed a solution based on axis-parallel rectangles to solve this problem and showed it to be significantly more effective than the standard supervised learning approach [7]. However, the standard assumption is not appropriate as negative bags may contain combinations of positive and negative instances.

Different formulations of the MIL problem have been developed over the years. Amores reviewed these various methods and compared their effectiveness for classification [2]. MISVM [3] and sbMIL [6] are two of the classical Multiple Instance Learning algorithms which follow the standard MIL assumption. These methods use local information based comparisons between individual instances and treat bag labels as aggregations of instance labels. Liu et al. proposed key instance detection by finding the instances that contribute the most to bag labels [20]. Kotzias et al. proposed a MIL formulation titled Group-Instance Cost Function (GICF) where negative bags contain some positive instances and developed a general cost function for determining individual instance labels from group labels [17]. In contrast with the standard assumption, this can be referred to as the *collective* assumption [2]. This is more applicable in metagenomics because a healthy person may have some pathogens while a patient may have some common microbial signatures as a healthy person.

LaPierre et al. proposed a phenotype prediction from metagenomic data [18] that utilizes instance space. This approach used the distances in instance space to represent bags which were then

¹<https://github.com/mrahma23/RegMIL>

Figure 1: This figure illustrates our proposed preprocessing steps which include sequence assembly, binning, clustering within bins, kmer counting and Locality Sensitive Hash (LSH) calculations.

classified by Support Vector Machine (SVM) classifier. But it can label only a limited number of instances. To our knowledge, no MIL based approach has been proposed that can score each instance in the large-scale setting seen here while providing competitive predictive accuracy.

The possibility of labeling data on bag-level (healthy controls or patients) as well as instance-level (sequences/contigs) within those bags makes Multiple Instance Learning (MIL) paradigm well suited for the problem of phenotype prediction from metagenomic data.

3 METHODS

Our proposed approach takes the collection of sequence reads from healthy controls and patients as input. It provides two types of outputs: (i) a score for each sequence that signifies the contribution of the sequence to the disease and (ii) label (healthy or patient) of the samples based on their sequence scores. In this section, we briefly describe our proposed approach step-by-step. First, we perform sequence assembly, binning, kmer counting and Locality Sensitive Hash (LSH) on the metagenomic data which is briefly described in Section 3.1. Then we apply Canopy clustering on the LSH of instances from training bags to generate instance scores (Section 3.2.1 and Section 3.2.2). In the next step, we use instance scores in the training set to train a two-layer neural network regression model (Section 3.2.3). The regression model then efficiently assigns scores to millions of instances in the test set (Section 3.2.3). Finally, we use a histogram based bag-level feature representation using the instance scores to classify bags (Section 3.3).

3.1 Preprocessing

Figure 1 shows the overall preprocessing steps to convert from raw sequence data to the input format of our proposed MIL based pipeline. Metagenomic data contains fragments of nucleotide sequences known as *reads*. The assembler uses alignment and merging to produce long contiguous sequences (contigs) which are likely to be part of an organism. We perform sequence assembly on the

raw sequences using the metaSPAdes [23] method. Our next preprocessing step is *binning*. Binning is a process by which assembled contigs are assigned to groups or *bins*. Assembled sequences from same or similar organisms are likely to be in the same bin. We use MyCC [19] an automated metagenomic binning tool for this step.

Finally, we use UCLUST [9] sequence clustering tool inside each bin. UCLUST is a popular greedy sequence clustering method that groups a set of similar sequences into clusters. A cluster is defined by one sequence, known as the centroid and every sequence in the cluster must have the similarity above a given identity threshold with the centroid. We retain the centroids after clustering the bins and ignore other sequences. This step removes redundant sequences and reduces the total number of sequences in the data.

3.2 Instance Scoring

Figure 2 shows our proposed pipeline for scoring instances in the training set. Figure 3 shows our proposed pipeline for scoring instances in the test set.

3.2.1 Canopy Clustering on the Instances of Training Set. First, we combine sequences of all bags in training set and perform clustering to rank instances in those bags based on the cluster output. Clustering can be computationally demanding due to a large number of instances in the training set. Thus we choose Canopy as an efficient and scalable clustering process for large-scale metagenomic data [29]. Canopy clustering [21] uses two distance thresholds, one *soft* threshold, T_1 and another *tight* threshold, T_2 . It calculates the distance between a data point and canopy (cluster) centroid. If the distance is within T_2 , it assigns the data point to that canopy and removes the data point from the dataset. If the distance is within T_1 it assigns the data point to the canopy but retains the data point for a better assignment in future. Canopy centroids are selected randomly until all data points are assigned to at least one canopy.

Canopy clustering requires an efficient and fast distance measure. Locality Sensitive Hashing (LSH) [11] provides a solution for the approximate or exact near neighbors search problem. Sequence

Figure 2: This figure illustrates how we generate positivity scores of instances in the training set with Canopy clustering. Positivity score of a canopy is the ratio of the number of member instances in that canopy which are from patient's bags to the total number of member instances in that canopy. Member instances inside a canopy share the same score.

Figure 3: This figure illustrates how we assign positivity scores to instances of bags in the test set. A two-layer neural network based regression model is created based on instances in the training set and their corresponding positivity scores. This model then scores each of the instances in the test set.

reads are represented with kmers, defined as a contiguous subsequence of length k . We construct the family of LSH functions with bit sampling [15] based on the normalized kmer counts. We compute LSH for each sequence and use normalized hamming distance for fast distance estimation for the Canopy clustering.

3.2.2 Assigning Positivity Scores to the Instances of Training Set. Positivity score is defined as the ratio of the number of sequences in a canopy from positive bags (patients) to the total number of sequences in that canopy. We assign this positivity score to each member sequence in the canopy. If an instance is the member of multiple canopies and receives multiple positivity scores then we take the average score. Canopies may share similar instances which emblemize the fact that abundance of certain microbes is likely to

be one of the differentiating factors in the disease state. Averaging the positivity scores provides a better aggregation of positivity scores of an instance by taking its contributions to multiple canopies. We apply Softmax function on the positivity scores of each training bag to normalize them in the range $[0, 1]$. Given M dimensional vector z of real numbers, the softmax function for element z_i for $i \in 1, \dots, M$ is defined as:

$$\sigma(z_i) = \frac{\exp(z_i)}{\sum_{j=1}^M \exp(z_j)} \text{ for } j = 1, \dots, M \quad (1)$$

3.2.3 Assigning Positivity Scores to the Instances of Test Set. Once we have positivity scores for each of the instances in training set, we train a two-layer neural network regression model. The input

Figure 4: This figure illustrates histogram based feature representation of bags from instance scores.

Figure 5: This figure illustrates class label prediction of bags using bag-level features.

to this regression model is the normalized kmer counts of the instances in the training set and their corresponding positivity scores. This regression model encodes the process of (i) converting the normalized kmer counts of instances in the test set to LSH, (ii) finding the nearest canopy centroid for each instance in the test set and (iii) generating positivity scores of each instance in the test set.

The regression model will provide the most likely positivity score without explicitly converting the test instances to LSH and finding their nearest canopies. This makes the pipeline efficient during prediction phase. Since there are millions of sequences in the training set, the neural network based regression model will be exposed to a large number of examples for training. We apply the Softmax function on the positivity scores of each bag in the test set to normalize them in the range [0, 1]. An instance from a test bag that belongs to a canopy where the majority of members are from patient's bags will receive higher positivity score. A lower positivity score, on the other hand, indicates that the instance belongs to a canopy with most of its members from healthy bags.

3.3 Bag Labeling

Unlike the huge volume of instances (sequence reads), the number of bags (healthy controls and patients) in metagenomic data is small. We formulate feature representation for bags such that it utilizes instance scores. We use a histogram based approach to build bag-level feature vectors. Instance scores are within the range of [0-1]. We divide this range into k equal groups. Each of these

groups become bag level features. Value of a feature is the ratio of instances in the bag with scores in that group to the total number of instances in the bag. Figure 4 shows the bag level feature vector creation mechanism from instance scores in our proposed model.

A bag level feature vector represents the relative frequencies of ranges of positivity scores in a sample. A healthy sample will have lower values for features representing frequencies of high positive instances and higher values for features representing low positive instances compared to patient samples. Conversely, a patient sample will have higher values for features representing frequencies of high positive instances and lower values for features representing frequencies of low positive instances compared to healthy samples. We train a classifier e.g. Random Forest (RF), Support Vector Machines (SVM) or Logistic Regression (LR) based classifiers with the bag-level features and classify bags as positive (patient) or negative (healthy). Figure 5 shows our proposed bag-level classification pipeline.

4 EXPERIMENTS

4.1 Dataset

We used two popular metagenomic datasets to evaluate our proposed approach. We used metagenomic data of liver cirrhosis patients and healthy controls [28]. This data was collected from 237 Chinese individuals. Qin et al. divided this data into two cohorts titled discovery and validation cohorts. Discovery cohort consists

Table 1: Dataset Statistics

Type	Datasets	Unprocessed		Processed	
		Total # of Sequences	Total Size	Total # of Sequences	Total Size
Full Datasets	Liver Cirrhosis [28]	14,442,320,193	2153.61 GB	83,503,278	1058.9 GB
	IBD [26]	8,037,389,400	537.39 GB	18,180,961	74.27 GB
Sampled Liver Cirrhosis Dataset	0.1% instances per bag ^a	N/A	N/A	137,681	824.68 MB
	0.2% instances per bag ^a	N/A	N/A	261,483	1.28 GB
	0.3% instances per bag ^a	N/A	N/A	384,461	1.72 GB
	0.4% instances per bag ^a	N/A	N/A	512,183	2.17 GB
	0.45% instances per bag ^a	N/A	N/A	573,628	2.39 GB
	0.5% instances per bag ^a	N/A	N/A	643,782	2.84 GB
	0.55% instances per bag ^a	N/A	N/A	712,123	3.21 GB
	0.1% instances per bag ^b	N/A	N/A	106,432	784.31 MB
	0.2% instances per bag ^b	N/A	N/A	219,161	1.07 GB
	0.3% instances per bag ^b	N/A	N/A	318,706	1.21 GB
Sampled IBD Dataset	0.4% instances per bag ^b	N/A	N/A	421,126	1.67 GB
	0.45% instances per bag ^b	N/A	N/A	472,324	2.06 GB
	0.5% instances per bag ^b	N/A	N/A	526,463	2.34 GB
	0.55% instances per bag ^b	N/A	N/A	582,319	2.46 GB

^a Percentage of top ranked positive instances sampled from each bag in discovery cohort of liver cirrhosis dataset.

^b Percentage of top ranked positive instances sampled from each bag in IBD dataset.

of 98 patients and 83 healthy controls. Validation is an independent cohort with 25 patients and 31 controls. The second dataset we used in this paper is the metagenomic data from Inflammatory Bowel Disease (IBD) [26] patients. The data was retrieved from 124 European individuals including 25 patients with IBD, 27 patients with Ulcerative Colitis (UC) and 12 patients with Crohn's disease (CD). Since the number of IBD patient in the second dataset is only 25, we consider IBD, Ulcerative Colitis and Crohn's disease as the disease state.

Table 1 shows the total number of sequences and disk sizes of the original datasets as well as smaller sampled datasets. Smaller sampled datasets were created by downsampling the original liver cirrhosis and IBD datasets which were required to run some of the Multiple Instance Learning (MIL) based approaches that do not scale with such large data (Section 5.2).

4.2 Software Details

We used metaSPAdes² [23] to perform assembly on raw sequence data. A benchmark comparison of sequence assembly tools [35] shows that metaSPAdes perform better than other assembly tools. We used MyCC³ [19], a de-novo binning tool to bin assembled sequence contigs. In order to remove redundancy inside a bin, we used UCLUST⁴ [9], one of the most widely used and cited metagenome clustering methods.

For the implementation of MISVM [3] and sbMIL [6], we used an open-source Python implementation by Doran [8], which is

available on GitHub⁵. We implemented GICF [17] in Python 2.7.0 with squared loss for errors and logistic regression for classification as in the original paper [17]. Method (mRMR-SVM) [28] proposed by Qin et al. and MetAML [24] are the two popular non-MIL baseline methods that we used in this paper for comparative analysis. Due to the complexity of extensive sequence preprocessing steps and unavailability of source codes for mRMR-SVM [28], we were unable to replicate the results for this baseline method and cited the results in Section 5 for comparative analysis. We implemented the pipeline of MetAML [24] following the directions of Pasolli et al. for MetAML. We used MetaPhlAn2⁶ and generated species-level abundance values for feature representation.

We used the Multilayer Neural Network Regression, Random Forest, Support Vector Machine and Logistic Regression based classifiers from the python package scikit-learn⁷ v0.19.0 [25] as this package provides comprehensive implementations of these methods. We refer to our developed pipeline by RegMIL.

4.3 Evaluation Metrics

We assess the success of our MIL-based pipeline in several ways. The simplest measure is accuracy which measures the percentage of instances that are classified correctly. Accuracy is represented by

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (2)$$

²<http://cab.spbu.ru/software/spades/>

³<https://sourceforge.net/projects/sb2nhri/files/MyCC/>

⁴https://www.drive5.com/usearch/manual/uclust_algo.html

⁵<https://github.com/garydoranjr/misvm>

⁶<https://bitbucket.org/biobakery/metaphlan2>

⁷<http://scikit-learn.org/stable/index.html>

where TP, TN, FP, and FN represent true positives, true negatives, false positives and false negatives respectively. Accuracy, as an evaluation metric, can be biased for imbalanced data. Precision measures the percentage of positive predictions that were correct, whereas recall measures the percentage of positive examples that were correctly predicted (or retrieved).

$$\text{Precision} = \frac{TP}{(TP + FP)} \quad (3)$$

$$\text{Recall} = \frac{TP}{(TP + FN)} \quad (4)$$

We also use the Area Under Curve of the Receiver Operating Characteristic (AUC-ROC), which is the plot of FP rate vs TP rate. We repeated each experiment ten times and calculated the margin of error for the mean with 95% confidence interval. The margin of error can be calculated as:

$$ME = t_{[1-\alpha/2, n-1]} \times \frac{\sigma_s}{\sqrt{n}} \quad (5)$$

where $(1 - \alpha)$ is the significance level, $t_{[1-\alpha/2, n-1]}$ is critical value of the t distribution with $n - 1$ degrees of freedom for an area of $\alpha/2$ for the upper tail, σ_s is the sample standard deviation and n is the sample size.

4.4 Hardware Details

We used the ARGO computing cluster available at George Mason University⁸. The computing cluster is configured with 35 Dell C8220 Compute Nodes, each with dual Intel Xeon E5-2670 (2.60GHz) 8 core CPUs, with 64 GB RAM.

4.5 Parameters

We used all default parameters, for metaSPAdes [23] assembly tool and MyCC [19] binning tool. For UCLUST [9] we used 0.90 as identity threshold for clustering. For kmers, the value of parameter k was set to 6. The parameters for bit length of LSH (d), canopy's soft ($T1$) and hard ($T2$) thresholds were set by performing a grid search and validation based on the accuracies at bag-level. For validation purposes, 10 positive (patient) bags and 10 negative (healthy) bags were randomly sampled from training set. We estimated the bit length d of LSH by fixing the soft ($T1$) and tight ($T2$) thresholds to 0.6 and 0.4, respectively. d was found to be proportional to the size of the input dataset. Once bit length for LSH (d) was estimated we tuned values for $T1$ and $T2$ by decreasing $T2$ from 0.4 (*relaxed*) to 0.1 (*strict*) in steps of 0.05 and varying $T1$ from 0.6 to 0.1 in steps of 0.05. The best values for d , $T1$ and $T2$ were found to be 527, 0.4 and 0.25, respectively.

The two-layer neural network regression model [25] was comprised of an input layer, two hidden layers with Relu activation and an output layer with logistic sigmoid activation. We used "Adam" for weight optimization with learning rate set to 0.01. The maximum number of iterations (terminating condition) for the regression model was set to 2000. Rest of the parameters were estimated empirically based on bag-level classification accuracies with the same randomly selected validation samples as before. We found that 20 equally sized histogram-bins for bag-level feature representation

perform well on average. For SVM we used "RBF" kernel with the Kernel coefficient set to 0.1. We used Stochastic Gradient Descent (SGD) based logistic regression classifier with squared loss and L2 penalty where learning rate and maximum iteration were set to 0.01 and 2500, respectively.

5 DISCUSSION AND RESULTS

In Section 5.1, we compare the classification performances of our proposed approach and other non-MIL based approaches such as method [28] proposed by Qin et al. and MetAML [24]. Common MIL based approaches such as MISVM [3], sbMIL [6] and GICF [17] do not scale with such large number of instances. Hence, we could not evaluate these MIL-based approaches using the original datasets. We sampled fewer instances per bag from the original liver cirrhosis and IBD datasets (Section 4.1) using our proposed instance scoring method (Section 3.2). In Section 5.2 we compare the classification performances of MISVM, sbMIL, GICF and our proposed RegMIL on these smaller datasets. In Section 5.3 we report on the computational time and memory requirements. We compare the performances of our proposed approach with different bag-level classifiers in Section 5.4. Finally, we show a case study on the liver cirrhosis dataset in Section 5.5 to validate our instance scoring mechanism.

5.1 Comparison With Non-MIL Based Methods

Table 2: Comparison of mean performance on liver cirrhosis dataset with discovery and validation cohort as training and test sets, respectively. The margin of errors are reported in parentheses.

Methods	Accuracy	AUC-ROC
Qin et al.	<i>Unknown</i> [28] ^a	83.6 [28] ^a
MetAML [24]	0.8571 (0.0461)	0.8593 (0.0417)
RegMIL (RF)	0.8750 (0.0374)	0.8755 (0.0349)

^a implementations are not publicly available

We compared our proposed method RegMIL with non-MIL approaches such as method [28] proposed by Qin et al. and MetAML [24] which were briefly discussed in Section 2. Table 2 shows the mean accuracy and AUC-ROC values of different methods with their corresponding margin of errors for the liver cirrhosis dataset with discovery cohort and validation cohort as training and test set, respectively. Our proposed method RegMIL provides higher mean accuracy and AUC-ROC values of 0.8750 and 0.8755, respectively.

Table 3 compares the mean Leave-One-Out-Cross-Validation (LOOCV) performance of our proposed method with MetAML and the method proposed by Qin et al. on the discovery cohort of liver cirrhosis and IBD dataset. Our proposed method RegMIL provided highest mean accuracy and AUC-ROC values of 0.9281 (with 95% confidence interval: 0.8920-0.9642) of and 0.9272 (with 95% confidence interval: 0.8925-0.9619), respectively. Because implementation of the method proposed by Qin et al. is not publicly available and it is specific to liver cirrhosis data, we could not evaluate this method on the IBD dataset. We see that RegMIL provides higher

⁸<http://orc.gmu.edu/research-computing/argo-cluster/argo-hardware-specs/>

Table 3: Comparison of mean performance (LOOCV) on different datasets with the margin of errors reported in parentheses.

Methods	Liver Cirrhosis (Discovery Cohort)		IBD	
	Accuracy	AUC-ROC	Accuracy	AUC-ROC
Qin et al.	<i>Unknown</i> [28]	0.9180 [28]	N/A ^a	N/A ^a
MetAML [24]	0.9226 (0.0412)	0.9221 (0.0398)	0.8387 (0.0281)	0.8242 (0.0374)
RegMIL (RF)	0.9281 (0.0361)	0.9272 (0.0347)	0.8466 (0.0352)	0.8442 (0.0262)

^a Method proposed by Qin et al. is specific to liver cirrhosis dataset.

mean accuracy value of 0.8466 (with 95% confidence interval: 0.8114-0.8818) and mean AUC-ROC value of 0.8442 (with 95% confidence interval: 0.8180-0.8704) than MetAML on the IBD dataset. This is because our proposed RegMIL scores each of the sequences based on their contributions to the disease and uses these scores for better bag-level representations of samples than methods which rely on matching sequences with databases of known organisms for feature generation. The number of sequences which are discarded from subsequent analysis due to a lack of match with reference database can be substantial for metagenomic data. Thus RegMIL can better utilize information from metagenomic data and provide high accuracy in disease prediction.

5.2 Comparison With MIL-Based Methods

Common Multiple Instance Learning (MIL) approaches have high computational requirements due to all pairwise comparisons. The datasets used in this paper contain millions of instances per bag. As a result MIL based approaches such as MISVM [3], sbMIL [6] and GICF [17] do not scale for such large datasets due to $O(N^2)$ computations where N is the number of instances. Hence we reduce the number of instances from each of the bags to run these methods and compare with our proposed approach.

Only a small fraction of the large pool of sequences in metagenomic data from healthy controls and patients differs and can be used for determining disease [33]. If we reduce the number of instances of each bag from millions to few thousands by sampling randomly then it is very likely that important sequences that can potentially discriminate between healthy and disease class, will not be selected. So we first ranked the instances in each bag using our proposed instance scoring method (Section 3.2). Then we sampled the top 0.1%, 0.2%, 0.3%, 0.4%, 0.45%, 0.5% and 0.55% high ranked sequences from each bag to create smaller datasets. Table 1 in Section 4.1 shows the statistics of sampled datasets. We ran MISVM, sbMIL, GICF and our proposed RegMIL on these smaller datasets. We used normalized kmer ($k=6$) counts of sequences as input to these methods. Each experiment was repeated ten times. Figure 6a and Figure 6b show the mean AUC-ROC values with standard errors of MIL methods for different percentage of original sequences per bag. We see that our proposed approach provide better AUC-ROC on average compared to others.

5.3 Runtime Comparison

Table 4 compares the computational time and memory requirements during prediction phase of MISVM, sbMIL, GICF, MetAML and RegMIL for sampled liver cirrhosis dataset (discovery cohort) with

Table 4: Classification time and memory usage on subset of instances (sampled datasets with 0.3% instances per bag) in discovery cohort of liver cirrhosis dataset.

Methods	Classification Time	Memory Usage
MISVM [3]	11.26 hours	22.38 GB
sbMIL [6]	13.48 hours	26.19 GB
GICF [17]	9.16 hours	8.37 GB
MetAML [24]	32 minutes	3.4 GB
RegMIL (RF)	6.24 minutes	2.6 GB

0.3% instances per bag. sbMIL consumed the highest amount of resources, 13.48 hours for classification and 26.19 GB of memory on average. Our proposed pipeline RegMIL took only 6.24 minutes to classify and consumed 2.6 GB of memory. RegMIL achieves efficiency by avoiding kernel based computations and expensive nearest cluster centroid identification. Rather it only retains the neural network regression model and bag-level classification model.

5.4 Random Forest (RF) vs Others

We wanted to compare the performance of our proposed RegMIL pipeline for different classifiers at bag-level. Table 5 shows the mean Leave-One-Out-Cross-Validation (LOOCV) performances over ten repeated experiments with different classifiers at bag level. We used Support Vector Machine (SVM), Logistic Regression (LR) based classifier and Random Forest (RF). We observe that Random Forest (RF) yielded better accuracy and AUC-ROC values overall for the discovery cohort of liver cirrhosis dataset and IBD dataset. Other bag-level classifiers such as SVM and LR also provide high accuracies and AUC-ROC values. We observe that the differences in the average performances of classifiers at bag level are very small.

5.5 Case Study: Instance Level Validation For Liver Cirrhosis Dataset

Figure 7 shows the BLAST⁹ [1] results of the top 15 unique positive sequences predicted by our proposed RegMIL from positive (patient) bags of validation cohort of liver cirrhosis dataset. These sequences are likely to contribute to liver cirrhosis. Basic Local Alignment Search Tool (BLAST) [1] finds regions of local similarity between sequences by aligning the query with the database of known sequences. After completion of our proposed pipeline,

⁹<https://blast.ncbi.nlm.nih.gov/Blast.cgi>

Figure 6: Figure 6a and 6b show the mean AUC-ROC values with standard errors from different approaches for sampled liver cirrhosis dataset (discovery cohort) and IBD dataset, respectively.

Table 5: Mean performance (LOOCV) of RegMIL with different classifiers at bag level with the margin of errors reported in parentheses.

Methods	Liver Cirrhosis (Discovery Cohort)		IBD	
	Accuracy	AUC-ROC	Accuracy	AUC-ROC
RegMIL (SVM)	0.8950 (0.0321)	0.8948 (0.0310)	0.8237 (0.0214)	0.8172 (0.0329)
RegMIL (LR)	0.9015 (0.0216)	0.8998 (0.0281)	0.8067 (0.0267)	0.8130 (0.0322)
RegMIL (RF)	0.9281 (0.0361)	0.9272 (0.0329)	0.8466 (0.0352)	0.8442 (0.0262)

Figure 7: Top 15 highly positive instances in patient's bags from validation cohort of the liver cirrhosis dataset with their respective BLAST matches and mean positivity scores.

BLAST results for each instance in patient's bags of validation cohort were retrieved. In case of multiple occurrences of same BLAST matches, their instance scores were averaged. Unmatched instances were ignored and remaining instances were sorted according to descending order of their positivity scores.

Streptococcus Salivarius, Streptococcus Anginosus, Clostridium Bolteae, Veillonella Parvula and Haemophilus Parainfluenzae are species that matched with some of the highly positive instances reported by our proposed method. These are also found to be dominant with high abundance in patients with liver cirrhosis [28], confirming the correctness of our instance scoring mechanism. Both Streptococcus and Veillonella genus are associated with oral cavity indicating that they invaded patient's gut from oral origin [30]. Some species from Clostridium genus such as Clostridium Bolteae are known to be associated with gut-related disorders such as Inflammatory Bowel Disease (IBD) and infections [16]. Moreover, we have found other highly positive sequences which were not shown to be significant for liver cirrhosis in the study of the dataset that we used [28]. For example, Klebsiella Pneumoniae have become important pathogens and contribute to liver abscesses [37]. This shows that our proposed approach is capable of identifying novel markers and more pathogens contributing to a disease than the methods which use reference gene sequences.

Lactobacillus Salivarius and Bacteroides Clarus were found to receive higher positivity scores from RegMIL but they are not reported to be directly associated with liver cirrhosis or other diseases in current literature.

6 CONCLUSION

We demonstrate an efficient Multiple Instance Learning (MIL) based pipeline for phenotype classification from metagenomic data that performs well compared to other standard MIL and non-MIL based approaches. Our proposed approach is capable of enumerating each sequence based on its contributions to the phenotype. This enables us to not only classify a person's state as healthy or disease but also identify the pathogens that contribute to that state which has clinical significance. Finally, it utilizes information from millions of instances effectively to classify the samples at bag-level. More generally, we have shown the effectiveness of Multiple Instance Learning (MIL) methods within metagenomics and phenotype prediction. Our proposed approach is a relatively simple and easy to implement pipeline. It has shown both strong results and significant potential for further improvements.

ACKNOWLEDGMENTS

This work is supported by NSF Career Award 1252318 to Huzefa Rangwala.

REFERENCES

- [1] Stephen F Altschul, Warren Gish, Webb Miller, Eugene W Myers, and David J Lipman. 1990. Basic local alignment search tool. *Journal of molecular biology* 215, 3 (1990), 403–410.
- [2] Jaume Amores. 2013. Multiple instance classification: Review, taxonomy and comparative study. *Artificial Intelligence* 201 (2013), 81–105.
- [3] Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann. 2003. Support vector machines for multiple-instance learning. In *Advances in neural information processing systems*. 577–584.
- [4] Fredrik Bäckhed, Ruth E Ley, Justin L Sonnenburg, Daniel A Peterson, and Jeffrey I Gordon. 2005. Host-bacterial mutualism in the human intestine. *science* 307, 5717 (2005), 1915–1920.
- [5] Pedro Belda-Ferre, Luis David Alcaraz, Raul Cabrera-Rubio, Héctor Romero, Aurea Simon-Soro, Miguel Pignatelli, and Alex Mira. 2012. The oral metagenome in health and disease. *The ISME journal* 6, 1 (2012), 46.
- [6] Razvan C Bunescu and Raymond J Mooney. 2007. Multiple instance learning for sparse positive bags. In *Proceedings of the 24th international conference on Machine learning*. ACM, 105–112.
- [7] Thomas G Dietterich, Richard H Lathrop, and Tomás Lozano-Pérez. 1997. Solving the multiple instance problem with axis-parallel rectangles. *Artificial intelligence* 89, 1-2 (1997), 31–71.
- [8] Gary Doran and Soumya Ray. 2014. A theoretical and empirical analysis of support vector machine methods for multiple-instance classification. *Machine Learning* 97, 1-2 (2014), 79–102.
- [9] Robert C Edgar. 2010. Search and clustering orders of magnitude faster than BLAST. *Bioinformatics* 26, 19 (2010), 2460–2461.
- [10] Dirk Gevers, Subra Kugathasan, Lee A Denson, Yoshiki Vázquez-Baeza, Will Van Treuren, Boyu Ren, Emma Schwager, Dan Knights, Se Jin Song, Moran Yassour, et al. 2014. The treatment-naïve microbiome in new-onset Crohn's disease. *Cell host & microbe* 15, 3 (2014), 382–392.
- [11] Aristides Gionis, Piotr Indyk, Rajeev Motwani, et al. 1999. Similarity search in high dimensions via hashing. In *Vldb*, Vol. 99. 518–529.
- [12] Jo Handelsman. 2004. Metagenomics: application of genomics to uncultured microorganisms. *Microbiology and molecular biology reviews* 68, 4 (2004), 669–685.
- [13] Elaine Y Hsiao, Sara W McBride, Sophia Hsien, Gil Sharon, Embriette R Hyde, Tyler McCue, Julian A Codelli, Janet Chow, Sarah E Reisman, Joseph F Petrosino, et al. 2013. Microbiota modulate behavioral and physiological abnormalities associated with neurodevelopmental disorders. *Cell* 155, 7 (2013), 1451–1463.
- [14] Philip Hugenholtz and Gene W Tyson. 2008. Microbiology: metagenomics. *Nature* 455, 7212 (2008), 481.
- [15] Piotr Indyk and Rajeev Motwani. 1998. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*. ACM, 604–613.
- [16] James M Kinross, Ara W Darzi, and Jeremy K Nicholson. 2011. Gut microbiome-host interactions in health and disease. *Genome medicine* 3, 3 (2011), 14.
- [17] Dimitrios Kotzias, Misha Denil, Nando De Freitas, and Padhraic Smyth. 2015. From group to individual labels using deep features. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 597–606.
- [18] Nathan LaPierre, Mohammad Arifur Rahman, and Huzefa Rangwala. 2016. CAMIL: Clustering and Assembly with Multiple Instance Learning for phenotype prediction. In *Bioinformatics and Biomedicine (BIBM), 2016 IEEE International Conference on*. IEEE, 33–40.
- [19] Hsin-Hung Lin and Yu-Chieh Liao. 2016. Accurate binning of metagenomic contigs via automated clustering sequences using information of genomic signatures and marker genes. *Scientific reports* 6 (2016), 24175.
- [20] Guoqing Liu, Jianxin Wu, and Z-H Zhou. 2012. Key instance detection in multi-instance learning. In *Asian Conference on Machine Learning*. 253–268.
- [21] Andrew McCallum, Kamal Nigam, and Lyle H Ungar. 2000. Efficient clustering of high-dimensional data sets with application to reference matching. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 169–178.
- [22] Joachim Messing, Roberto Crea, and Peter H Seeburg. 1981. A system for shotgun DNA sequencing. *Nucleic acids research* 9, 2 (1981), 309–321.
- [23] Sergey Nurk, Dmitry Meleshko, Anton Korobeynikov, and Pavel A Pevzner. 2017. metaSPAdes: a new versatile metagenomic assembler. *Genome research* 27, 5 (2017), 824–834.
- [24] Edoardo Pasolli, Duy Tin Truong, Faizan Malik, Levi Waldron, and Nicola Segata. 2016. Machine learning meta-analysis of large metagenomic datasets: tools and biological insights. *PLoS computational biology* 12, 7 (2016), e1004977.
- [25] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research* 12, Oct (2011), 2825–2830.
- [26] Junjie Qin, Ruiqiang Li, Jeroen Raes, Manimozhayan Arumugam, Kristofer Solvsten Burgdorf, Chaysavanh Manichanh, Trine Nielsen, Nicolas Pons, Florence Levenez, Takuji Yamada, et al. 2010. A human gut microbial gene catalogue established by metagenomic sequencing. *nature* 464, 7285 (2010), 59.
- [27] Junjie Qin, Yingrui Li, Zhiming Cai, Shenghui Li, Jianfeng Zhu, Fan Zhang, Suisha Liang, Wenwei Zhang, Yuanlin Guan, Dongqian Shen, et al. 2012. A metagenome-wide association study of gut microbiota in type 2 diabetes. *Nature* 490, 7418 (2012), 55.
- [28] Nan Qin, Fengling Yang, Ang Li, Edi Prifti, Yanfei Chen, Li Shao, Jing Guo, Emmanuelle Le Chatelier, Jian Yao, Lingjiao Wu, et al. 2014. Alterations of the human gut microbiome in liver cirrhosis. *Nature* 513, 7516 (2014), 59.
- [29] Mohammad Arifur Rahman, Nathan LaPierre, Huzefa Rangwala, and Daniel Barbara. 2017. Metagenome sequence clustering with hash-based canopies. *Journal of bioinformatics and computational biology* 15, 06 (2017), 1740006.
- [30] Anthony H Rogers. 2008. *Molecular oral microbiology*. Horizon Scientific Press.
- [31] Delphine M Saulnier, Kevin Riehle, Toni-Ann Mistretta, Maria-Alejandra Diaz, Debasmita Mandal, Sabeen Raza, Erica M Weidler, Xiang Qin, Cristian Coarfa, Aleksandar Milosavljevic, et al. 2011. Gastrointestinal microbiome signatures of pediatric patients with irritable bowel syndrome. *Gastroenterology* 141, 5 (2011), 1782–1791.
- [32] Duy Tin Truong, Eric A Franzosa, Timothy L Tickle, Matthias Scholz, George Weingart, Edoardo Pasolli, Adrian Tett, Curtis Huttenhower, and Nicola Segata. 2015. MetaPhlAn2 for enhanced metagenomic taxonomic profiling. *Nature methods* 12, 10 (2015), 902.
- [33] Peter J Turnbaugh, Micah Hamady, Tanya Yatsunenkov, Brandi L Cantarel, Alexis Duncan, Ruth E Ley, Mitchell L Sogin, William J Jones, Bruce A Roe, Jason P Affourtit, et al. 2009. A core gut microbiome in obese and lean twins. *nature* 457, 7228 (2009), 480.
- [34] Peter J Turnbaugh, Ruth E Ley, Micah Hamady, Claire M Fraser-Liggett, Rob Knight, and Jeffrey I Gordon. 2007. The human microbiome project. *Nature* 449, 7164 (2007), 804.
- [35] John Vollmers, Sandra Wiegand, and Anne-Kristin Kaster. 2017. Comparing and evaluating metagenome assembly tools from a microbiologist's perspective—Not only size matters! *PLoS one* 12, 1 (2017), e0169662.
- [36] Jun Wang and Huijue Jia. 2016. Metagenome-wide association studies: fine-tuning the microbiome. *Nature Reviews Microbiology* 14, 8 (2016), 508.
- [37] Ya-Sung Yang, LK Siu, Kuo-Ming Yeh, Chang-Phong Fung, Shenq-Jie Huang, Han-Chang Hung, Jung-Chung Lin, and Feng-Yee Chang. 2009. Recurrent Klebsiella pneumoniae liver abscess: clinical and microbiological characteristics. *Journal of clinical microbiology* 47, 10 (2009), 3336–3339.