5-2023

Computational Analysis of Microbial Sequence Data Using Statistics and Machine Learning

Zhixiu Lu
University of Tennessee, Knoxville, zlu21@vols.utk.edu

To the Graduate Council:

I am submitting herewith a dissertation written by Zhixiu Lu entitled "Computational Analysis of Microbial Sequence Data Using Statistics and Machine Learning." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Computer Science.

Scott J. Emrich, Major Professor

We have read this dissertation and recommend its acceptance:

Scott Emrich, Michela Taufer, Stephanie Kivlin, Audris Mockus

Accepted for the Council:

Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Computational Analysis of Microbial Sequence Data Using Statistics and Machine Learning

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Zhixiu Lu

May 2023

Acknowledgements

I would like to express my deepest gratitude to my mentor Dr. Scott Emrich for his supervision, guidance and support throughout my research and study. My skillset and scientific view of bioinformatics has expanded because of him. I appreciate my dissertation committee members, Dr. Audris Mockus, Dr. Michela Taufer and Dr. Stephanie Kivlin for their helpful discussions and critiques. I also want to thank my father, Dr. Caijian Lu and my wife, Susi Dai for their encouragement and support along the way.

Abstract

Since the discovery of the double helix of DNA in 1953 (74), modern molecular biology has opened the door to a better understanding of how genes control chemical processes within cells, including protein synthesis. Although we are still far from claiming a complete understanding, recent advances in sequencing technologies, increased computational capacity, and more sophisticated computational methods have allowed the development of various new applications that provide further insight into DNA sequence data and how the information they encode impacts living organisms and their environment.

Sequencing data can now be used to start identifying the relationships between microorganisms, where they live, and in some cases how they affect their host organisms. We introduce and compare methods used for this bioinformatics application, and develop a machine learning model that can be used to effectively predict environmental factors associated with these microorganisms.

Codon Usage Bias (CUB), which refers to the highly non-uniform usage of codons that code for the same amino acid has been known to reflect the expression level of a protein-coding gene under the evolutionary theory that selection favors certain synonymous codons. Traditional methods used to estimate CUB and its relation with protein translation have been proven effective on single-celled organisms such as yeast and *E. coli*, but their applications are limited when it comes to more complex multi-cellular organisms such as plants and animals. To extend our abilities to further understand the relations between codon usage patterns and the protein translation

processes in these organisms, we develop a novel deep learning model that can discover patterns in codon usage bias between different species using only their DNA sequences.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction And Background

For over half of a century, studies centering around genetics have been an essential piece to the formation and development of modern evolutionary theory (74). More recently, and in large part facilitated by the global human genome sequencing efforts of the 1990s (45), DNA sequencing costs have been decreasing exponentially as shown in Figure 1.4. As a concrete example, the cost of whole genome sequencing for a single human went down from 100 million to less than 1000 dollars in the past twenty years. Such a rapid decrease in genome sequencing cost spawned by new sequencing platforms has promoted a vast number of new applications of genomics data.

This dissertation considers the intersection between more traditional bioinformatics modeling with emerging machine learning advances, specifically more deep-learning-inspired approaches for biological data. On a more general level, the broader field of bioinformatics is a scientific subdiscipline that leverages computational techniques to process biological data. In this chapter, we provide background information of relevance to later chapters with respect to both DNA and protein sequences and conclude with a brief summary of overall and novel contributions in the dissertation for the analysis of largely microbial sequence data.

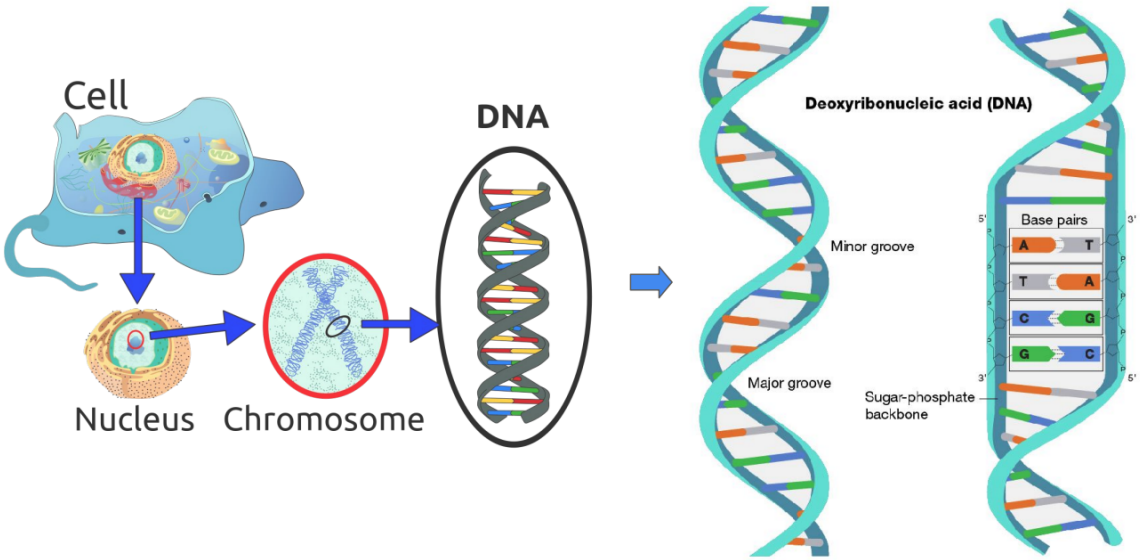

Figure 1.1: Double Helix Structure of DNA

Image credit: Sponk, Tryphon, Magnus Manske, User:Dietzel65, LadyofHats (Mariana Ruiz), Radio89, CC BY 3.0, via Wikimedia Commons (Left)
NIH - DEOXYRIBONUCLEIC ACID (Right)

1.1 Background

1.1.1 DNA Sequences

DNA is the hereditary material of all living organisms. Information within DNA is represented by a code consisting of four nucleotides: Adenine (A), Guanine (G), Cytosine (C), and Thymine (T).

1.1.2 Protein Translation

The coding region of a gene, also known as the coding DNA sequence (CDS), is the portion of a gene that during a process called protein translation has the "recipe" for the sequence of amino acid residues that can build a protein.

1.1.3 Codons

A codon is a sequence of three DNA or RNA nucleotides that corresponds to a specific amino acid or stop signal during protein translation. It follows that the genetic code includes $4^3 = 64$ possible permutations of three-letter nucleotide sequences that can be made from the four nucleotides. Of the 64 codons, 61 represent amino acids, and three are stop signals. For example, as shown in the cDNA codon table, the codon TTT corresponds to the amino acid phenylalanine, and TAA is a stop codon.

1.1.4 Codon Usage Bias

Codon Usage Bias (CUB), which refers to the highly non-uniform usage of codons that code for the same amino acid—which are called synonymous codons—has been studied for decades. Many factors have been proposed to help explain this observation including gene expression level, mutational preferences of a given organism, amino acid conservation constraints, and protein hydropathy (5).

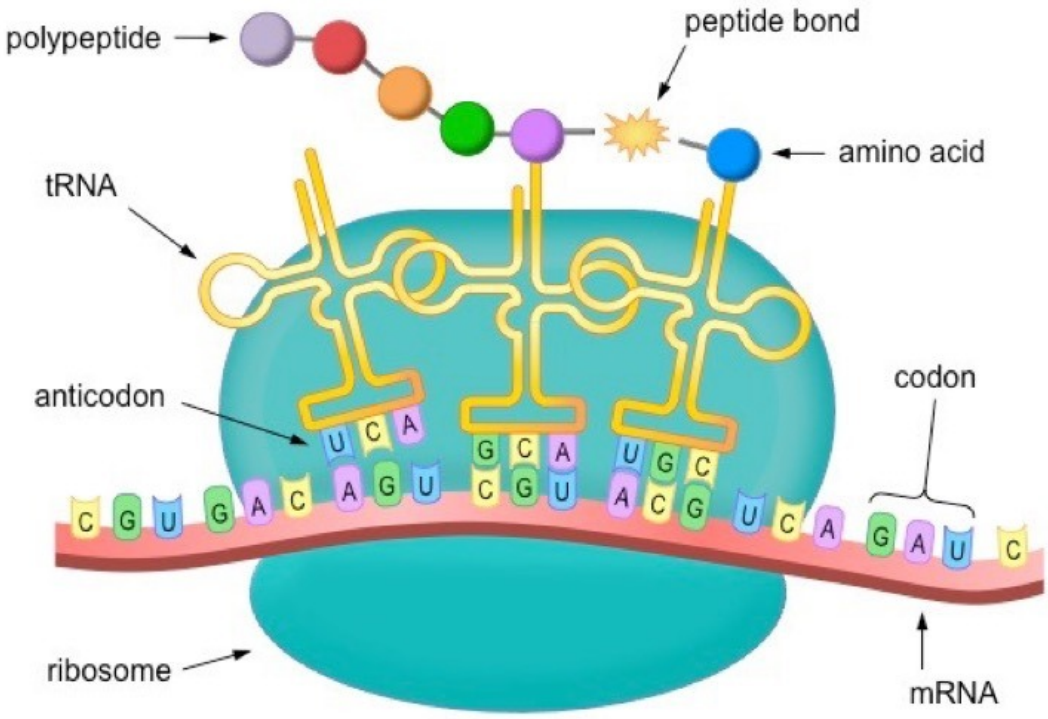

Figure 1.2: Visualization of the Translation Process
Image credit: Bioninja. 2022. *Translation*. Available at: https://ib.bioninja.com.au/.

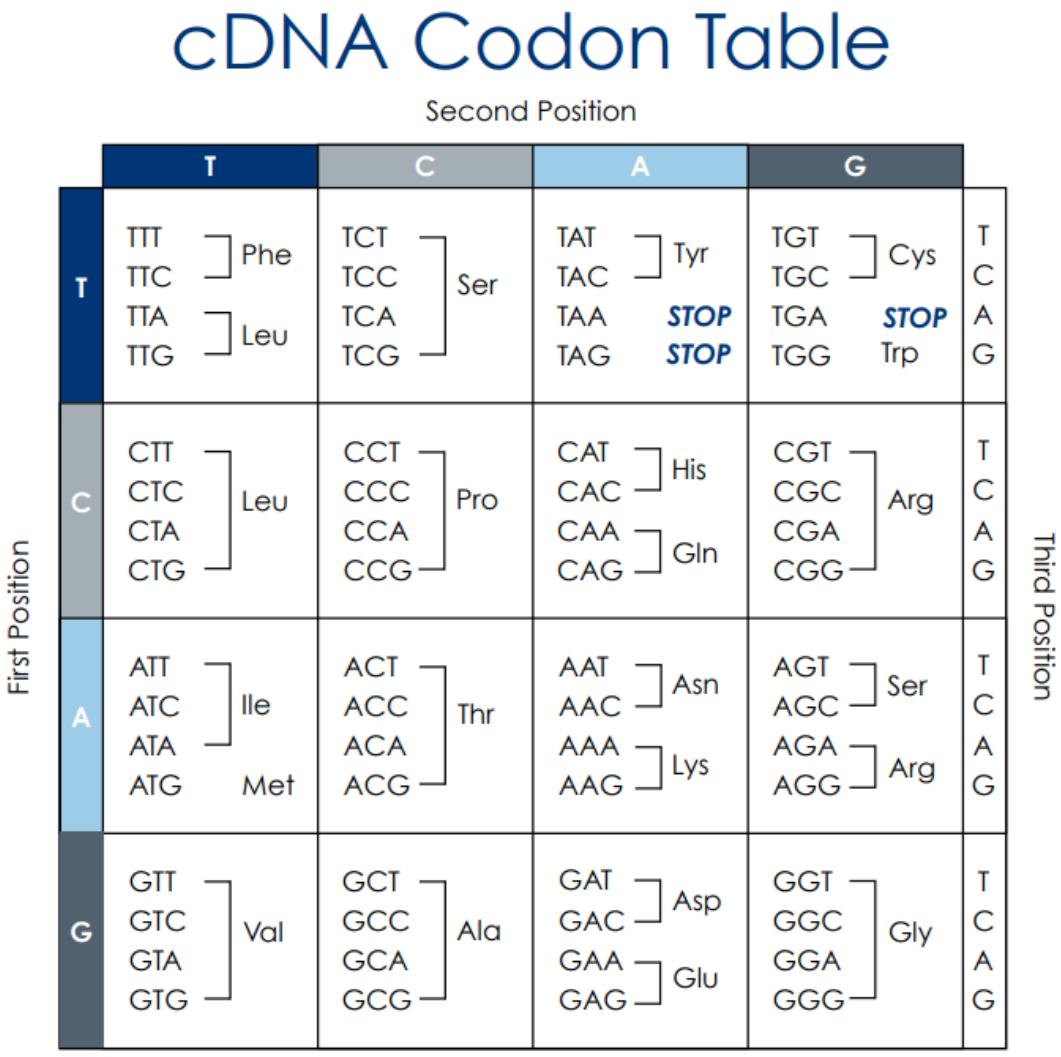

Figure 1.3: cDNA Tables For Codons Used to Translate into Amino Acids

Image credit: Genomenon Codon Charts 2022

1.1.5 Amplicon sequence variants

An amplicon sequence variant (ASV) is an inferred single DNA sequence high-throughput marker gene analysis. In this de novo process, erroneous biological sequences are removed during PCR and sequencing, under a denoising model that assumes biological sequences are more likely to be repeatedly observed than error-containing sequences. (17)

1.1.6 Operational Taxonomic Units

Another standard unit for marker-gene analysis is the operational taxonomic unit (OTU), generated by clustering sequences based on a threshold of similarity. Compared to ASVs, OTUs reflect a coarser notion of similarity. This can be a point of contention between biologists given that OTUs have been shown to be slightly less informative compared to ASVs; however, the difference is usually not significant (17).

1.2 Contributions in this dissertation

This dissertation is a collection of bioinformatics efforts that mostly fall into two sub-disciplines – analysis of protein sequences and applications for use of microbial sequence data, with an emphasis on microbial ecology. In Chapter 2, we develop new applications of microbiome next-generation sequence data. We first describe a novel method we developed to make inferences about multiple environmental conditions, especially in the area of monitoring recent forest fires based on soil microbiome data collected from nearby Smoky Mountain National Park. We then consider additional microbiome data and consider multiple approaches to effectively build machine learning classification frameworks for them. We ultimately focus on a recent female microbiome data challenge whose data can be leveraged to better predict preterm birth during pregnancy. Our ecological framework has been implemented as a tool called "SelectMicro" and an Application note submission is planned for

Bioinformatics. Our prediction framework for preterm birth microbiome data has led to new internal University of Tennessee (UT) funding and a new collaboration with leading Women's health researchers at the UT Medical Center starting at the end of 2022.

In Chapter 3, we describe the effects of codon usage bias on the process of protein translation. Codon usage bias has been known to reflect the expression level of a protein-coding gene under the evolutionary theory, that selection favors certain synonymous codons. We extended a prior framework that incorporated another evolutionary factor, namely mutation bias and its effect on codon usage. We describe an improved method, which we call MLE-Phi, that has much greater computation efficiency and a wider range of applications than the more basic modeling framework on which it is based. This work received the best paper award at the 12th *International Conference on Bioinformatics and Computational Biology* in 2021.

Although measuring the effect of functional codon bias in simple organisms such as yeast and *E. coli* has proven to be effective and accurate, codon-based methods perform less well in higher organisms such as plants and humans. Chapter 4 is the first attempt that we know about that discovers potential patterns of codon usage using natural language processing (NLP) approaches. We also include a brief discussion of a similar community project that we took part in that uses NLP on random promoter sequences with the goal of predicting gene expression. We conclude that NLP has great potential to make improved inferences about gene expression in higher organisms. Further, we speculate that transformer-based frameworks, such as BERT (Bidirectional Encoder Representations from Transformers) can be used to "learn" codon preferences of an organism and facilitate improved heterologous gene expression, i.e., optimally producing protein from a gene from one species in a well-characterized alternative species like *E. coli* (18) We leave building NLP models of codons for this new application for future work.

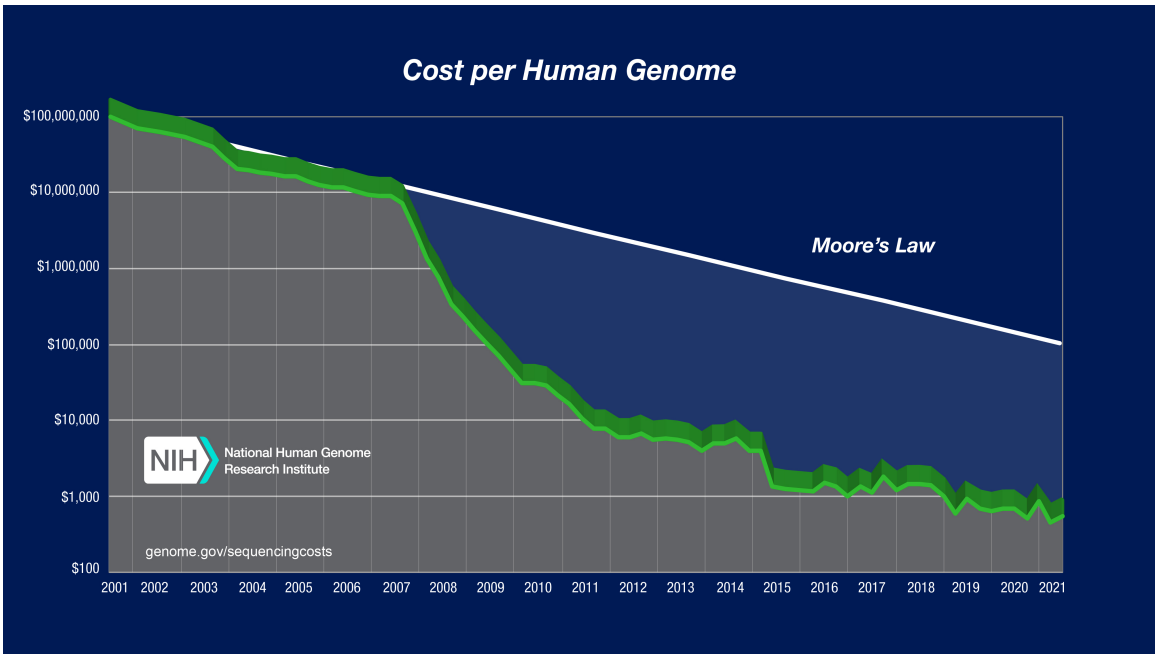

Figure 1.4: Cost of Whole Human Genome Sequencing for Past 20 Years (NIH)

Chapter 2

Applications for Microbiome Next Generation Sequence Data

Microbiomes, as a vital part of many environments, can interact with plants and promote their growth. Moreover, the composition of soil microbiomes can reflect multiple properties of a site from which a sample is taken (30). Most current microbiome studies focus on clinical gut data to aid in disease prediction (30; 43).

An increasing number of scientific applications now rely on high-throughput sequencing. For example, sequence-inferred microbial abundance data have been combined with environmental factors to prioritize engineering strategies where technical or financial resources are limited (12). In this chapter, we consider associating species occurrence information with certain environments or clinical factors using statistical learning methods. Rather than focusing solely on classification and prediction tasks, we present a framework that aims to extract the most informative features that will increase microbiome classification performance, reduce computational time, and allow for biological interpretation. We show that our approach is capable of predicting various factors with high accuracy and precision. Furthermore, with a novel multi-label classification framework that we call "SelectMicro," we show that this framework can predict available environmental factors with high precision and accuracy.

Further, some of the selected sequence-based markers appear to also be promising indicator species for microbial ecology applications. This will enable making future insights about microbial data and how it can be associated with different types of environments.

2.1 Associating microbiome information with environmental factors

2.1.1 Background

One somewhat common microbial ecology application of interest to biologists is predicting indicator species (13; 70), which can reflect environmental changes over time by associating observed microbiome members with existing environmental data (9; 28; 66). We take this idea one step further: can we use microbiome information to predict environmental factors? Our novel ensemble feature extraction and classification framework is targeted towards sequenced microbiome data, using different datasets with differing scales of external factors.

2.1.2 Available data

We initially tested our approach on three distinct microbiome datasets. We describe these data below.

Smoky Mountain Forest Fire Data

Study site. During late November and early December 2016, the Chimney Tops 2 wildfire burned approximately 44.4 km^2 within the Great Smoky Mountains National Park (GSMNP). On November 28th, the fire moved into the city of Gatlinburg, TN, and burned a further 28.2 km^2. This wildfire was extremely heterogeneous, with unburned sites adjacent to severely burned sites, often within meters of each other.

From within the burn matrix we randomly selected three sites from "high burn" areas, three "low/medium burn" sites, and three "no burn" sites within natural areas of the GSMNP as well as from the exurban area of Gatlinburg, TN.

Bioinformatics. AM fungal sequences are processed using the DADA2 pipeline in R (54). First, primers are trimmed from all sequences and sequence error rates are calculated. Sequences are then merged into unique amplicon sequence variants (ASVs). Finally, chimeras are removed using a *de novo* chimera checker. Because the NS31–AML2 primers may amplify some non-AM fungal fungi, we then BLAST representative sequence reads from each ASV against the MaarjAM database(Opik et al. 2010) and only retain reads that matched a known AM fungal virtual taxonomic unit by at least 97%. Sequences are deposited in the NCBI Sequence Read Archive (BioProject ID: PRJNA771625). All other data are available via the Environmental Data Initiative (EDI), doi:10.6073/pasta/1cac7b2ccd2262773f92600205f1d812.

Rocky Mountains Data

Study sites. We sampled foliar fungal endophytes and root fungi (root endophytes and AM fungi) in the Colorado Rockies at the Rocky Mountain Biological Laboratory, Gunnison County, Colorado, USA (38°57'N, 106°59'W). This region has predictable decreases in air temperature (c. 0.8 °C per 100m;(52)) and declines in soil nutrients with altitude (24), but increases in precipitation, mainly as snow. ((38))

To capture environmental, spatial, and plant-host-specific variation in fungal guilds, we sampled 66 sites encompassing 9 to 13 elevations from each of six elevational gradients in July 2014. Elevational gradients represented separate mountains in the Gunnison Basin and were located within 20 km of each other.

At each location, we sampled nine adult individuals from up to 13 grass species representing five genera (*Poaceae*, subfamily *Pooideae*; *Achnatherum lettermanni, Achnatherum nelsonii, Elymus elymoides, Elymus scribneri, Elymus trachycaulis, Festuca brachyphylla, Festuca saximontana, Festuca thurberi, Poa alpina, Poa*

leptocoma, Poa pratensis, Poa stenantha, and *Trisetum spicatum*). Samples were based on tissue type (leaves v. roots) and plant species within each site.

Bioinformatics. This dataset is processed to generate Operational Taxonomic Units (or OTUs), which are clusters of closely related sequences and therefore slightly different from the exact ASV-based approach used on the previous Smoky Mountains data. Given that microbial ecologists use both types of features (17) we want to consider at least one OTU study as well. We merge paired-end reads using the fastq-mergepairs from USEARCH (v9.2.64) (25) with "fastq-maxdiffs" set to 20 and "fastq-maxdiffpct" set to 10 to ensure proper merging at a low error rate. The merged reads and the forward unmerged reads are then trimmed at the primer sites using cutadapt with "e" set to 0.2, "m" set to 200, and untrimmed reads are discarded. Merged reads are filtered using fastq-filter from USEARCH with "fastq-maxee" set to 1.0. The forward reads are first trimmed to 230 using fastx-truncate from USEARCH with "trunclen" set to 230 and then filtered by fastq-filter from USEARCH with "fastq-maxee" set to 1.0. We then concatenate the merged and forward reads into one file and de-replicate using fastx-uniques from USEARCH with "minuniquesize" set to 2. After these steps, 11,357,274 sequences remain. We cluster these sequences to form OTUs at 97% similarity (Estensmo et al., 2021) using cluster-otus command from UPARSE. All OTUs identified as "fungi" are retained, and OTUs labelled as "unknown" or "unidentified" are manually inspected based on blast results to determine if they are kept for further analysis.

Human Gut Metagenomic Data

Derived from the Human Microbiome Project (39), the human gut metagenomic datasets considered here were generated from six different disease cohorts: inflammatory bowel disease (IBD), type 2 diabetes in European women (EW-T2D), type 2 diabetes in Chinese (C-T2D), obesity, liver cirrhosis (Cirrhosis), and colorectal cancer (Colorectal). These data have been widely used by prior applications of machine learning to microbial communities including metAML (48) and deep-Micro (47). We

note that a direct comparison of our framework with these two specific alternatives is not possible because we do not prioritize single-label predictions. We include these data to assess our model.

2.1.3 Prior approaches and methods

Statistical approaches for analyzing microbiome data are diverse and often involve principal component analysis (PCA), linear regression, or hierarchical clustering (46). Unfortunately, the "species" matrices derived from ecological microbiome data (either ASV or Operational Taxonomic Unit (OTU)-based) are large, usually sparse, and often exhibit a great deal of variation between samples.

One of the biggest challenges of analyzing microbiome data is their high dimensionality: it's common to have millions of OTUs/ASVs and only dozens of samples to analyze. This introduces the so-called "curse of dimensionality," which refers to the high sparsity induced by high dimensionality that can cripple most statistical and/or machine learning methods.

To avoid the curse of dimensionality, dimension reduction techniques such as PCA (principal component analysis), LDA (Linear Discriminant Analysis), or some variations of deep learning models, e.g., auto-encoders (26), is used to encode the data in a lower dimensional space representation. While these techniques may preserve important information within the original dataset, these representations are often not interpretable. Moreover, microbiome data may also be "noisy" and as such affected by either confounding or unrelated variables.

To help overcome these limitations, deep learning models such as Recurrent Neural Networks (RNN) have been used for both classification and feature dimensionality reduction (65) with some success. Training a typical deep learning model, however, usually requires at least hundreds (and often thousands) of samples to achieve ideal performance (29). Such large numbers of samples are both difficult and costly to collect in more ecological settings. Moreover, a deep learning approach often

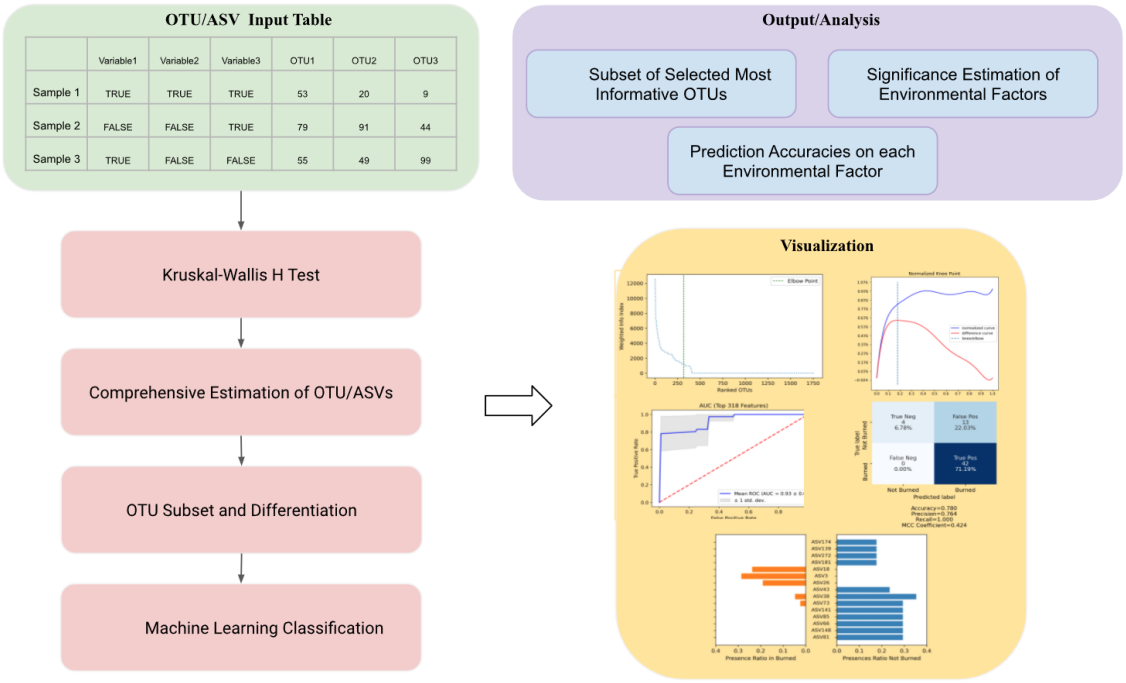

Figure 2.1: Model Pipeline Illustration for Select-Micro

requires converting features (e.g., ASVs) into different (and often less interpretable) representations.

Normalization

Microbiome richness can vary greatly between samples, thus, abundances of microbiomes between samples can also vary greatly. It is therefore important to normalize abundances. We achieve this by converting ASV/OTU abundances from each sample using the simple equation below:

$$P_k = \frac{A_k}{\sum_{i=1}^{n} A_i} \tag{2.1}$$

Where:

P_k : Ratio of OTU/ASV k within a sample
A_k : Raw Abundance of OTU/ASV k
$\sum_{i=1}^{n} A_i$: Sum of OTU/ASV abundances

After conversion from raw abundances to their ratios within a sample, we consider OTU/ASVs that make up $\geq 1\%$ of the total microbiome community as "present" per (51). Next, all present OTU abundances are replaced by their ranks within sample. OTU/ASVs that do not pass the 1% threshold would be considered absent and therefore have the lowest possible rank.

2.1.4 Our novel feature ranking framework for microbiome analysis

We propose to address these challenges using a subset of features (e.g., ASVs) that exhibit the most variation between environmental factors. In short, the basic idea is to find microbial "species" that seems to be the most associated with a certain environmental factor, and then check if these markers can also be

linked to environmental change(s). We test our framework by finding the most relevant features that help uncover associations between microbiomes and known environmental factors. In contrast to the mostly single-label tasks of interest in human gut microbiome studies, more generic ecological studies, especially for soil microbiomes, have multiple environmental factors that could be involved. It is therefore important to look at these factors more comprehensively, and discover environmental factors that have detectable influences on microbial communities.

Kruskal Wallis H-Test

Analysis of variance (ANOVA) is a widely used statistical method to test differences between groups by comparing their means, which can be used to help identify "species" that have significantly different distributions between different environments; however, ANOVA requires that tested variables are homoscedastic, i.e., the residual errors between independent variables (OTU/ASVs) and dependent variable (environmental factor) are consistent. In practice, this means residuals from the ANOVA model should follow a normal distribution, even though previous research has suggested that this is often not a valid assumption for microbiome data (21).

$$H = (N-1)\frac{\sum_{i=1}^{g} n_i \left(\bar{r}_i - \bar{r}\right)^2}{\sum_{i=1}^{g}\sum_{j=1}^{n_i} \left(r_{ij} - \bar{r}\right)^2} \tag{2.2}$$

2.1.5 Results of our and alternative approaches

A broad range of feature selection methods exist and have been widely applied, and many have shown to be highly effective in extracting important features in similar datasets. For this analysis, we considered five different feature selection methods based on different mathematical assumptions. In addition, we considered six different machine learning classifiers that are commonly used, as shown in Figure 2.1. Similar to other frameworks, classification performance, measurements of accuracy, precision, recall, and F1 scores are computed for validation, and each sample was tested using

the Leave-One-Out approach (26). We also include metAML (49), which is a widely used classification metagenomics-based prediction framework that uses LASSO (69) as the feature selection method. metAML also supports using either a vector machine or a random forest, and we report the results of the better performer per (49).

Although our approach is primarily designed to find the microbiome OTUs/ASVs that are most predictive for a group of environmental conditions, for most of the tasks our results are at least as good as the baseline model and MetAML, which runs feature selection on each environmental factor individually.

2.2 Using microbiome data to uncover clinically important information

2.2.1 Background

DREAM challenges are community competitions in which experimentalists "hold out" valuable validation data to test competing computational approaches in areas including systems biology and translational medicine, and therefore can use said data to evaluate the methodology of independent research groups across the world for important bioinformatics problems. More detailed information about Dream challenges can be found at their website. The main advantage for computational researchers such as our group is the collection of metadata, data anonymization (in the case of clinical data), and impact is usually well-defined given a clear biological question and the common evaluation of the computational approach; it is clear which methods work better than others given all groups work on the same data with the same (held out) validation applied.

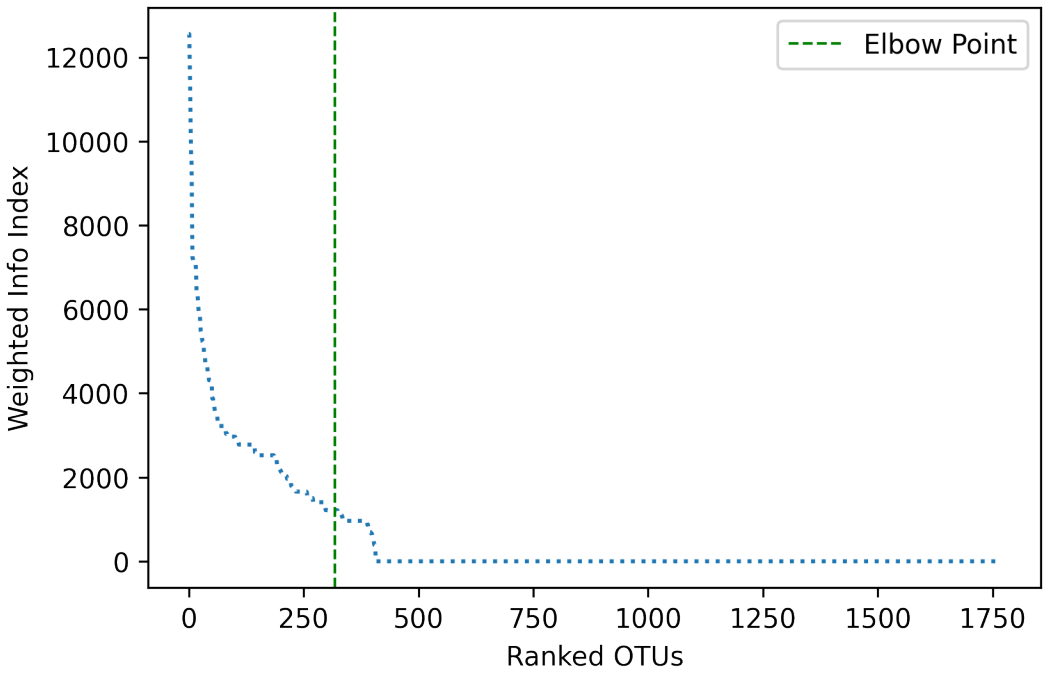

Figure 2.2: OTU Feature Scores along ranks

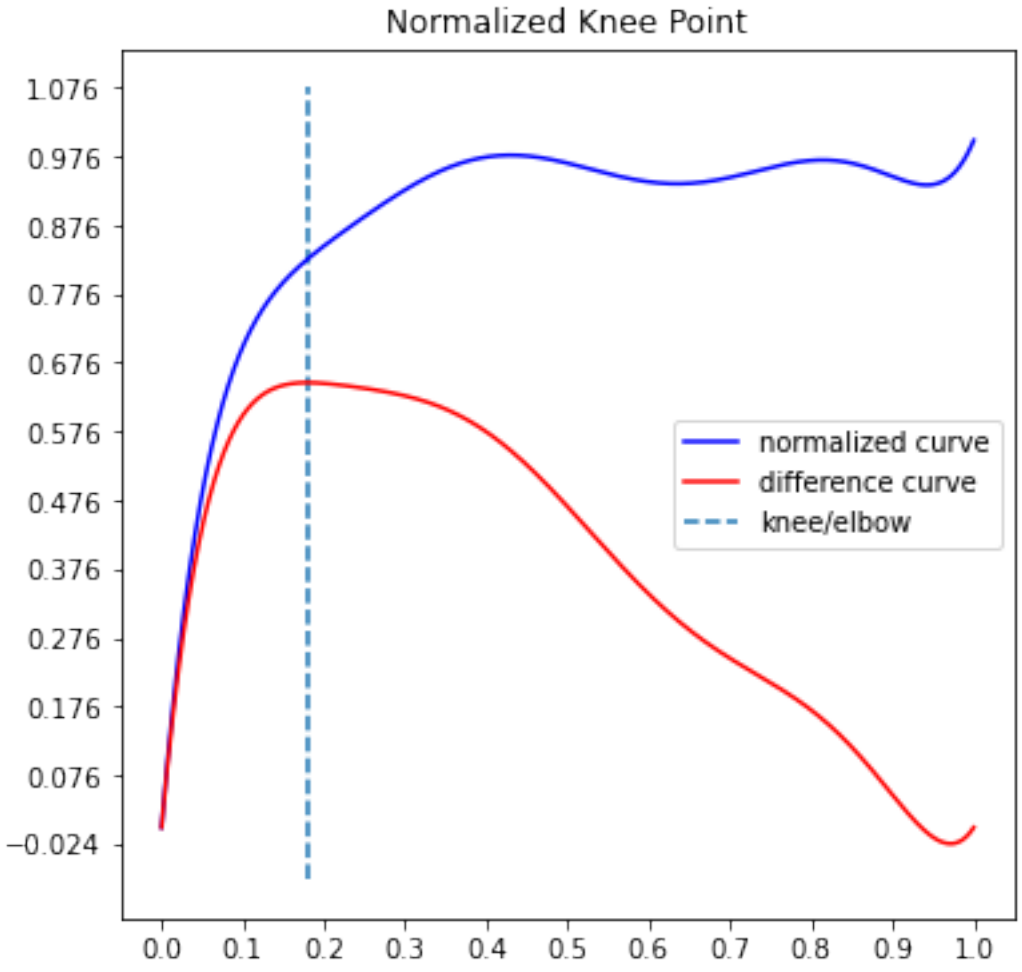

Figure 2.3: Normalized Knee point calculated for feature number cut off

Data	Envrionment Labels	AUC (Random Forest)	AUC (MetAML)	AUC (OurModel)
Smoky Moutain Data	Burned vs Not Burned	0.93 (±0.06)	0.94 (±0.06)	0.96 (±0.03)
	Urban vs Rural	0.64 (±0.10)	0.67 (±0.10)	0.74 (±0.10)
	Anual vs Perenial	0.68 (±0.11)	0.74 (±0.09)	0.69 (±0.10)
Colorado Data	Avery	0.79 (±0.14)	0.84 (±0.13)	0.88 (±0.06)
	Cinnamon	0.78 (±0.18)	0.80 (±0.16)	0.69 (±0.11)
	Hunter Hill	0.76 (±0.07)	0.80 (±0.10)	0.86 (±0.07)
	Ruby	0.76 (±0.09)	0.86 (±0.10)	0.89 (±0.06)
	Teocali	0.80 (±0.08)	0.81 (±0.09)	0.85 (±0.07)
	Treasury	0.51 (±0.01)	0.63 (±0.05)	0.61 (±0.11)

Figure 2.4: Model Comparison

Data	Envrionment Labels	Random Forest	MetAML	Our Model
Smoky Moutain Data	Burned vs Not Burned	1768	333	65
	Urban vs Rural		284	
	Anual vs Perenial		346	
Colorado Data	Avery	2589	586	166
	Cinnamon		567	
	Hunter Hill		603	
	Ruby		538	
	Teocali		468	
	Treasury		491	

Figure 2.5: Feature Number Comparison

2.2.2 The role of microbiome analysis

The DREAM challenge organizers hypothesized that patient microbiome data could be used to predict which women were at a higher risk of preterm birth.

We focused on the challenge's Task 1 which aimed to identify women at high risk of preterm birth greater than or equal to 37 weeks. The test data only included samples collected no later than 32 weeks of gestation.

2.2.3 Our approach

We initially approached this challenge by treating it as a binary classification task (preterm birth vs not preterm birth), using a 70-30 train test split ratio. After making preliminary benchmarks using the microbiome data matrix with around 20 standard classifiers used, we picked the top three performing classifiers (SVM with RBF kernel, AdaBoost, and Random Forest). With further benchmarks on these three models using AUC score calculated with cross-validation, we find that Random Forest slightly outperformed the other two classifiers consistently, so we decided to use a single model moving forward. Our implementation is publicly shared at github.

Based on our experience with environmental microbiome data we focused primarily on both feature engineering, e.g. alternative feature selection methods, and attempts to leverage external information—and specifically clinical metadata—to perform performance, similar to how environmental data boosted performance earlier in the chapter.

None of the feature selection methods significantly improve model performance. Using simple linear regressions, we find that three provided labels in the metadata: collection week, race, and age, correlated with the binary labels (preterm birth vs non-preterm birth). So we decided to augment the training data by transforming these labels and treating them as additional features alongside the microbiome ASVs.

Collection Week: We simply added this label as a feature alongside the ASVs, after scaling it to be between 0 and 1 using min-max scaling.

Age: Since this label is noisier (containing missing data points or ambitious range), we first tried replacing these missing values and ambiguous ranges with their median age values, then adding it as another feature for training like collection week, based on the benchmarks of the model prediction, adding continuous estimated age as features did not help the model gain any performance. We then tried to convert the age vector to a binary feature vector, using different cutoffs, after multiple testings, we find that using 35 as the age cutoff and adding this binary age feature vector (0 for age < 35 and 1 for age $>=35$) consistently drives the model prediction performance higher, we tested different values of age cutoff, and find 35 to be the best cutoff, with 31 being a close second.

Race: We transformed this label by calculating the ratio of positive labels (preterm birth) amongst each race group, and use this ratio as the estimate of the risk coefficient for each race group, then normalize this ratio with min-max scaling to fit the range of 0 and 1, with 0 meaning least risk and 1 being the highest risk.

2.2.4 Results

After multiple tests with AUC scores from cross-validation, we see a consistent again in performance after we transform these three given labels in the metadata and add them as features alongside the microbiome ASVs. Based on the submission feedback calculated from the test dataset, our AUC score increased from 0.610 to 0.6228 by adding these three labels, further confirming that these metadata labels can be potentially used to help the model make better predictions if used correctly.

2.3 Conclusion

In this chapter, we have introduced multiple applications of microbial data and how approaching these with different computational methods can lead to results with different performances, and there is no singular method that can be used to effectively

understand different datasets, in the pre-term challenge, we show that the although the model difference is significant, especially between decision tree-based methods and non-decision tree-based methods, the model differences between decision tree based models are marginal. Instead, adept reweighing of samples leads to much more significant differences in model performances, which suggested that the quality of each sample in the training set is variable. For smoky mountain soil data and the Colorado gradient soil data, different classification models show a much larger difference in performance, and the ranking of features is more important in these datasets, these different elements that attribute to significant model performance improvement can further provide us with insights about each dataset and understanding the biological implications behind these elements.

Chapter 3

Codon Usage Models and MLE-Phi

The main body of the following text was published in BICOB (2020), https://doi.org/10.29007/87r9

3.0.1 Introduction

Codon usage bias, which refers to using synonymous codons that code for the same amino acid at different rates, has been studied for decades. Codon usage bias has been known to reflect the expression level of a protein-coding gene under the evolutionary theory that selection favors certain synonymous codons. For example, the Codon Adaptation Index (CAI) relies on relative synonymous codon usage observed in highly expressed genes, and has been effective at predicting gene expression in unicellular microorganisms (63). Inspired by CAI, tAI goes further and incorporates tRNA gene copy number that exhibits a high and positive correlation with overall rRNA abundance (55). The underlying assumption behind CAI and tAI is proteins with higher expression contain more optimal codons. Because optimal codons help achieve faster translation with less error, protein-coding genes with a higher ratio of optimal codons likely have experienced more positive selection over time.

Codon usage within multi-cellular organisms with smaller effective population sizes—such as flies, plants and humans—should be less directly affected by selection

(55; 73). To improve prediction performance for all organisms, the Mutation-Selection-Drift balance model was proposed in which selection favors optimal codons and less optimal codons persist due to genetic drift. Codon bias can therefore be thought of a balance between both mutation (e.g., GC content of an organism) and selection (e.g, either high expression or a focus on higher accuracy).

3.0.2 Prior work

ROC-SEMPPR uses a Bayesian Markov chain Monte Carlo (MCMC) to estimate the strength of selection on codon usage (31). Because this model considers both selective pressure and mutational bias, it can be more comprehensive than models that rely solely on features in highly expressed genes. This advantage is not "free": ROC-SEMPPR's MCMC calculations are also significantly more computationally intensive than most traditional codon usage models. For example, using the current implementation of ROC-SEMPPR required about 19 hours to process 8.5 Mb of yeast genome data in early 2020. Codon-specific metrics such as CAI and tAI, on the other hand, are much faster because they use rely on pre-computed values. For example, the CAI estimate for any given gene sequence is simply the geometric mean of each codon's respective value under the model.

3.0.3 Developing a faster codon usage model

ROC-SEMPPR is capable of calculating codon-specific estimates of selection pressure and mutation bias. These estimates have been used to estimate gene expression (Φ) based on this previous equation from (31).

$$p_i = \frac{\exp\left[-\Delta M_{i,1} - \Delta\eta_{i,1}\Phi\right]}{\sum_{j=1}^{n_g}\exp\left[-\Delta M_{j,1} - \Delta\eta_{j,1}\Phi\right]} \tag{3.1}$$

We previously leveraged ideas from ROC-SEMPPR to develop a faster, more flexible codon usage model that also relies on pre-computed values. This new method, which we called MLE-Φ (Maximum Likelihood of Φ), estimates the protein synthesis

rate Φ on arbitrary intervals using previously computed ROC-SEMPRR parameters. With this modified Φ estimation framework, we can also predict gene expression at a much finer grain than prior efforts.

3.0.4 Preliminary results

$\Delta\eta$ is the ROC-SEMPPR measure of relative translation inefficiency for synonymous codons, scaled relative to the preferred codon under selection pressure (Preferred codon has a $\Delta\eta = 0$). In other words, the higher $\Delta\eta$ is, the less efficient the codon is compared to the preferred codon for a specific amino acid. ΔM describes the ratio of the frequencies of one codon relative to the reference under pure mutation; it represents how mutational favored (mutation biased) a codon is relative to the preferred codon. Mutation rates are not always equal, so when there is little selection acting on codon usage (e.g., when gene expression is very low), codon frequencies will be dominated by these more mutation-favored codons as detailed in Gilchrist 2015 (31).

$$\prod_{n}^{n+k} \frac{\exp\left[-\Delta M_{i,1} - \Delta\eta_{i,1}\Phi\right]}{\sum_{j=1}^{n_g} \exp\left[-\Delta M_{j,1} - \Delta\eta_{j,1}\Phi\right]} \tag{3.2}$$

Here n marks the start position of a codon window/interval that spans k codons (when this formula is applied to an entire gene, $n = 0$ and k = gene length / 3). By finding a Φ that maximizes the output probability for this specific window, we can get a effective estimate of Φ much faster, especially since MLE-Φ is optimized by Newton's root approximation method. In experimental studies such as a ribosome footprint count analysis (local translation rates), it has been shown that the ribosome covers about 10 codons in a transcript, suggesting an ideal value for k should be approximately ten for modeling protein translation.

Implementation of MLE-Φ and respective computed values of $\Delta\eta$ and ΔM for several most studied organisms are hosted on **GitHub**, and using these pre-computed values of $\Delta\eta$ and ΔM to find the maximum likelihood of Φ significantly reduces

computation time. In Table 4.4 we benchmark both methods using different model organisms. Although this approach required running the original ROC-SEMPPR at least once, our method can produce subsequent estimates for these organisms in only seconds (versus days in some cases).

MLE-Φ closely approximates MCMC-based Φ (Figure 3.1). As expected, the agreement of the two approaches tends to be better for highly expressed genes. This is to be expected as codon usage bias should have stronger detectable effects on genes with higher expression (32; 35; 71). Low expression genes correlate less well, in part because they tend to be noisier and harder to measure experimentally (15; 68). Even so, there is a clear and strong correlation between the original and our new approach with an overall correlation coefficient of 0.93.

Comparison of tAI, CAI and MLE-Φ Estimates

These comparisons suggest that the prediction of gene expression is significantly improved under our framework, and suggests that quantification of mutation bias is essential for fully understanding synonymous codon usage. We also propose an improved method, namely MLE-Φ, with much greater computation efficiency and a wider range of applications. An implementation of this method is provided at https://github.com/zlu-volyote/MLE-Phi.

We tested the performance of Φ estimation using empirical gene expression data relative to both CAI and tAI. This assessment will determine what effects (if any) incorporating mutation bias (ΔM) has on our predictions. We computed these gene expression measurements and then computed their correlation using the same approach as Causton et al. (2001). MLE-Φ's correlation is always higher than CAI for all data, and higher than tAI for 3/5 data sets considered (see Table 3.1). Combined, this supports using our new Φ estimation framework for predicting gene expression.

Because the more inclusive MLE-Φ model should perform better than CAI and tAI for more complex organisms, we next decided to compare different metrics for organisms under less selection pressure. Although MLE-Φ has the highest overall

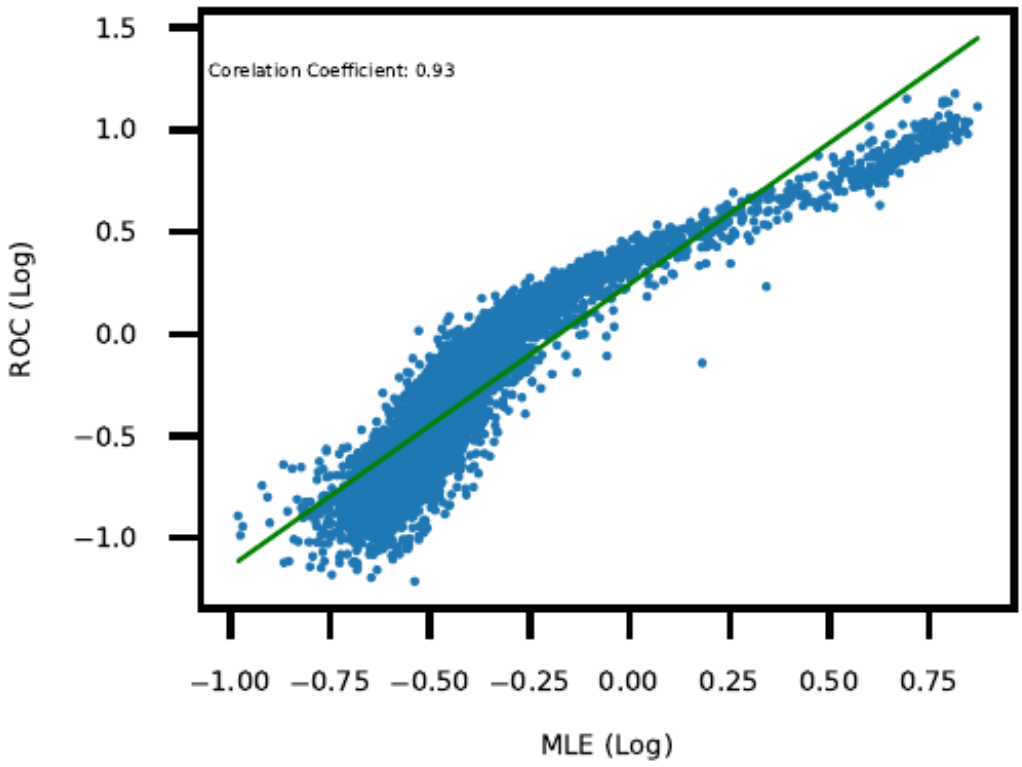

Figure 3.1: Correlation between MLE Φ and ROC-SEMPPR Φ As shown there is a 0.93 Pearson correlation between these two measures, which indicates that our new MLE estimation framework closely corresponds with the calculations from the original ROC-SEMPPR.

Table 3.1: Comparison of three metrics for different yeast data A comparison between our three considered metrics using previously published yeast mRNA abundances. Based on the Pearson correlation between predictions and empirical gene expression data, all three methods perform similarly in yeast.

	MLE-Phi	tAI	CAI
Arava 2003	0.637	0.621	0.643
Sun 2012	0.602	0.600	0.560
Nagalakshimi 2008	0.521	0.532	0.500
Holstege 1998	0.763	0.710	0.718
Causton 2001	0.688	0.676	0.657

correlation for all organisms (Table IV), Φ is not always significantly better than tAI and CAI (Z score, $p < 0.01$).

3.0.5 Looking at the effects of other factors affecting expression

Based on the Selection-Mutation-Drift model, more complex organisms with smaller effective population sizes should be more tolerant of drift and therefore are expected to be less affected by selection pressure. For example, in the original tAI paper, the authors estimated the selection pressure on different organisms. Although yeast, considered above, has strong estimated pressure (0.77-0.82), this pressure is only 0.24 in the model organism *Arabidopsis thaliana* and almost non-existent in humans (0.03) as shown in Table 3.2.

Because the more inclusive MLE-Φ model should perform better than CAI and tAI for more complex organisms, we next decided to compare different metrics for organisms under less selection pressure. Although MLE-Φ has the highest overall correlation for all organisms (Table IV), Φ is not always significantly better than tAI and CAI (Z score, $p < 0.01$).

As described previously, measurement (and therefore assessment) is more difficult for genes with lower overall expression ((15) (68)). It is also possible that a given gene may have different expression levels under different conditions/cell types in multicellular organisms. To address this issue, "Housekeeping" genes have historically been used, which are genes involved in basic cell maintenance that are expected to maintain consistent expression levels irrespective of tissue type, developmental stage, or external signals. Although there are also a few genes such as 16S, tus, rpoD, glyA, dnaB, gyrA, pykA/F, pfkA/B, mdoG and arcA that are widely used, it is difficult to obtain these specific values for the organisms we are studying (27).

To overcome this issue we extend a previously published method from 2007 (27) that used RT-PCR-based abundance estimates to rank genes. By picking genes on

Table 3.2: Estimation of Selection Pressure in Several Eukaryotes

Organism	**Common Name**	**Estimated Selection Pressure**
Arabidopsis thaliana	Thale Cress	0.24
Caneohardibtis elegans	Round Worm	0.45
Drosophila melanogaster	Fruit Fly	0.31
Homo saipens	Human	0.03
Plasmodium falciparum	Malaria Parasite	0.17
Saccharomyces serevisiae	Baker's Yeast	0.77
Schizosaccharomyces pombe	Fission Yeast	0.82

Table 3.3: Correlation-based comparison of the three considered metrics using the top 5% of highly expressed genes and empirical expression data
Fisher's R-Z transform is used to compute the Z score

	Yeast	Roundworm	Fruitfly	Arabidopsis
Same Size	518	2190	1393	2756
MLE-Phi	0.765	0.606	0.546	0.302
tAI	0.696	0.579	0.424	0.257
CAI	0.726	0.580	0.309	0.106
Z Score (Phi,tAI)	2.39	1.38	4.22	1.81
Z Score (Phi, CAI)	1.41	1.33	7.73	7.62

the top of the generated rankings, our selections would likely be "housekeeping" gene candidates and, more importantly, for this analysis, have more stable expression levels. Here, rather than using RT-PCR RNA abundance data we create rankings based on RNA-seq expression data from each organism and analyze the top 5% of the highly expressed genes based on these data. As shown in table 3.3, this approach generates candidates that are less noisy when compared to considering all protein-coding genes. After reconsidering the Pearson correlation coefficients between the considered metrics and prior empirical measurements, our Φ framework still consistently outperforms other methods for all organisms tested.

We also analyzed the difference between correlation coefficients using Fisher's R-Z transform. As shown above, we observe consistently positive Z scores with most comparisons having a corresponding p-value less than 0.05. This further confirms our hypothesis that, by weighting in the effect of mutation bias, Φ-estimation is more comprehensive and therefore a more accurate estimate fo organisms where selection pressure is not the dominant driver of codon usage bias.

3.0.6 Looking deeper into mutation bias

We have shown above that our MLE estimate of Φ has better accuracy than other traditional codon usage metrics when mutation bias impacts gene expression level estimates.

To confirm that mutation bias is responsible for the observed differences between model predictions, we created rankings for each coding gene in *D. melanogaster* using Φ, tAI, and the prior empirical measurements. We then sorted the genes by the ranking distance differences between tAI and Φ relative to the empirical measurement data. As we move from genes with the highest prediction differences to the lowest, we observe a clear shift in GC content (see Figure 3.2). This further confirms that mutation bias plays an important role in the computational prediction of gene expression, especially for multi-cellular organisms such as *Drosophila*.

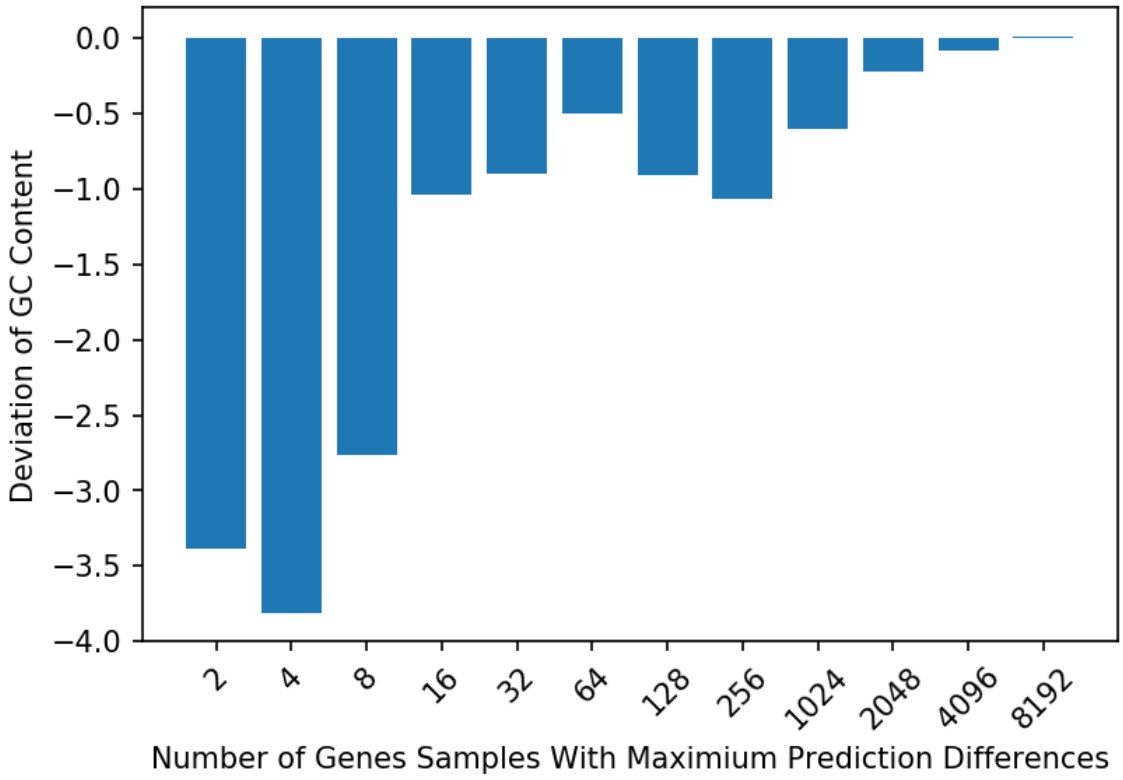

Figure 3.2: Shift of GC content across genes with different levels of prediction differences between using Φ and tAI The x-axis represents the number of genes with the highest prediction differences between Φ and tAI, samples with a smaller size contain genes with more prediction differences, while the y-axis represents the deviation from sample mean of GC content to the population mean calculated from all 11,196 coding genes in *Drosophila*). The observed GC bias decreases as we sample less different predictions between Φ and tAI.

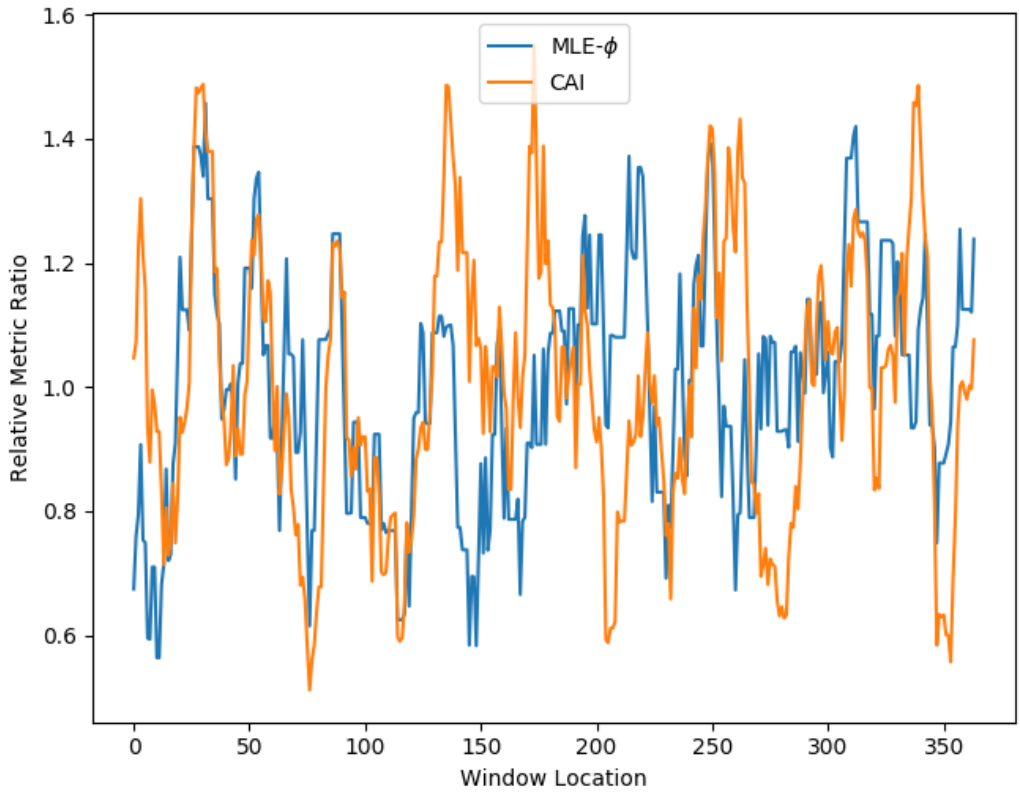

Figure 3.3: Relative MLE-Φ and CAI Window Estimation MLE-Φ and CAI for k=10 codon windows for the ACT1 gene in yeast; values along the x-axis mark the start codon position of the window, values on the y axis represent the ratio between window metric estimate and whole gene metric estimate. This illustration indicates that although ACT1 is a "housekeeping" gene with consistent global gene expression estimates (difference in ranking $< 1\%$) using different methods, there is visible disagreement in the more local translation rate estimates using these approaches.

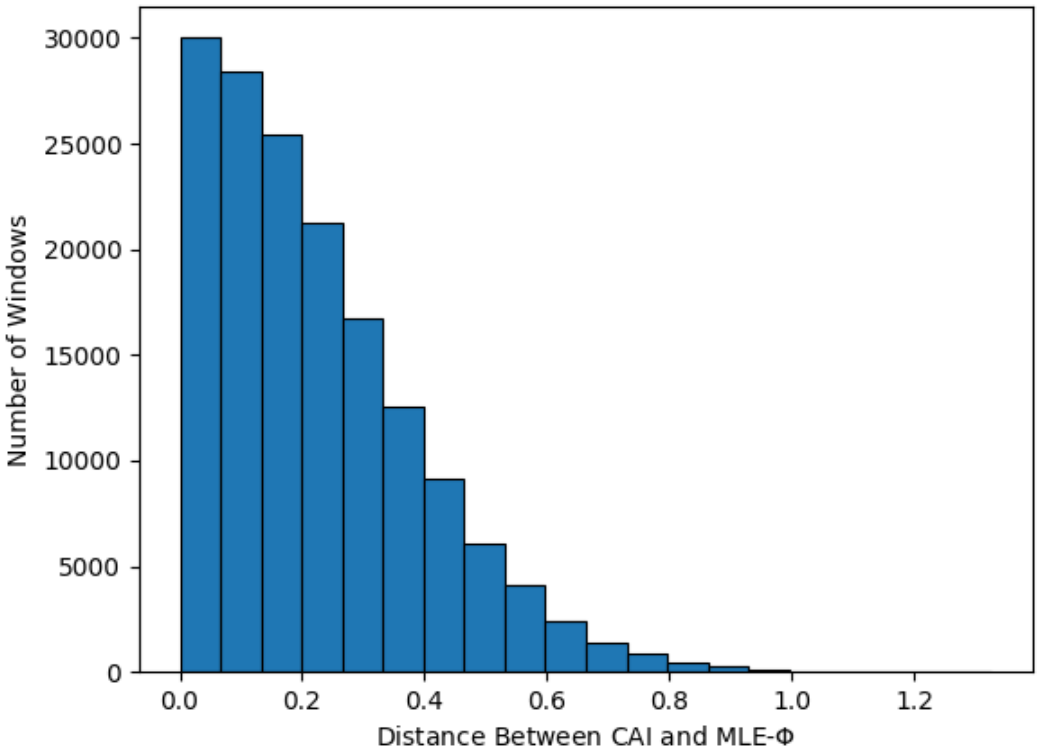

Figure 3.4: Distribution of Window Measurements by CAI and MLE-Φ Figure shows distributions of the distance between CAI and MLE-Φ, x label shows the distance if the relative metric ratio between MLE-Φ and CAI, values along y-axis represent the number of windows (window size of 10 codons) with respective measurement distance.

3.0.7 Local vs Global estimates

To ascertain how mutation bias affects so-called "translation tempo", or the rate by which a ribosome transcribes a specific region, we compared local measurements of MLE-Φ and CAI using a window-based analysis (similar to our prior work in (18)), created a ranking of CAI and Φ-based gene expression estimates, and selected a total of 300 genes (5% of roughly 6000 coding genes in yeast) with the least difference in overall expression level predictions.

For an example gene, we present ACT1(YFL039C) which is a housekeeping gene ranked in the top 5% of highly expressed by MLE-Φ, CAI, and all other methods considered here. As expected, gene-wide predictions of the expression level of ACT1 are very similar; however, we see clear variations in certain local regions (see Figure 3.3).

This result indicates that local protein translation rate estimates between models can vary, even when global gene expression level predictions are similar. The rationale is CAI and MLE-Φ are global estimates that converge to the same level but do not indicate how fast/slow specific regions are translated. To illustrate this more clearly, we computed the distance between MLE-Φ and CAI for codon windows in a total of 300 (roughly 5%) of the genes described above (Figure 3.4). Although most genes have similar estimates using CAI and MLE-Phi, there is a number like ACT1 that differ substantially. This is a major contribution of this work since there was no local/window-based version of ROC-SEMPPR Φ prior to our developing MLE-Φ as reported here and therefore such differences between global and local estimates has not yet been reported to the best of our knowledge.

3.0.8 Discussion

Gene expression is a topic of great interest in biology, and there are a wide range of approaches to model it(10)(77). For example, prior work has applied probabilistic and machine learning approaches based on microarray data and typically achieved

a prediction accuracy between 73% to 79% in yeast (10). Similar performance is achieved using codon usage bias-based estimates such as CAI and tAI. We extend and improve upon ROC-SEMPPR to develop a new MLE-Φ framework, which allows estimating expression using any arbitrary interval. This allows using codon usage bias to better understand other areas of biological interest such as protein synthesis rates and co-translational protein folding.

Estimation of Φ also provides a more comprehensive interpretation of codon and incorporates mutation bias estimates ΔM from ROC-SEMPPR. We confirm that mutation bias plays an important role in shaping observed codon usage bias. By only selecting the top 5% genes that are highly expressed, which is the exact method that underlies CAI and TAI-based estimates, we observe that our new method MLE-Φ is always better. This suggests that incorporating mutation bias into the expression model better predicts the precise expression level of a gene, even in highly expressed genes that are expected to have codon usage dominated by selection. This discovery is most important for more complex organisms like *D. melanogaster*, Arabidopsis and humans.

Significantly, we provide for the first time a framework that can use selection and mutation-based parameters for more localized windows. Prior work, including ours ((18)), have shown that rare codons are evolutionary conserved and some likely help proteins fold by slowing down the ribosome translation complex, a phenomenon called "co-translational folding" in the literature. We show that a number of genes in yeast have the same global estimate but differ greatly in a more local window-based analysis (see Figures 3.3 and 3.4). We are currently using our new MLE-Φ framework with ongoing experimental validation of preferred codon usage models for genes known to co-translationally fold (see (57) for details). This will provide biological support that MLE-Φ, which incorporates selection and mutational bias to better predict overall gene expression, also is better able to estimate the "tempo" of the ribosome and aid in downstream protein-focused research.

Chapter 4

From Sequence to Expression, Using Deep Learning Models to Decipher Relations Between Sequences and Gene Expression

4.1 Dream Challenge 2022 - Predicting promoter sequences using millions of random promoter sequences

4.1.1 Challenge Description

"Decoding how gene expression is regulated is critical to understanding disease. Regulatory DNA is decoded by the cell in a process termed "cis-regulatory logic", where proteins called Transcription Factors (TFs) bind to specific DNA sequences within the genome and work together to produce as output a level of gene expression for downstream adjacent genes. This process is exceedingly complex to model as a large number of parameters is needed to fully describe the process.

Having the ability to understand cis-regulatory logic in the human genome is an important goal and would provide insight into the origins of many diseases. However, learning models from human data is challenging due to limitations in the diversity of sequences present within the human genome (e.g. extensive repetitive DNA), the vast number of cell types that differ in how they interpret regulatory DNA, limited reporter assay data, and substantial technical biases present in many omic methods. To overcome these issues, we have recently created high-throughput measurements of the cis-regulatory activity of millions of randomly generated promoters in the single-cell organism Yeast (de Boer et al. 2020). Here, the expression level generated by each promoter sequence is measured via a fluorescent reporter gene regulated by a promoter (Sharon et al. 2012). The set of randomly generated promoter sequences is so large that it rivals the complexity of the entire human genome, which gives us unprecedented power to learn the many parameters required to understand gene regulation (see Rationale). Because both human and Yeast cis-regulatory logic uses similar principles, we hope that the model architectures learned on yeast data can inform how to create models for the human genome. " (14)

4.1.2 Overall Approach

For this challenge, we use a basic BERT transformer (23), training the labeled training in a 2-step process: whole data training and subset fine-tuning. We first train the model with a designed regression trainer using all 6.7 million samples provided in the training data, then based on the bins of expression tiers, sample equally from 18 expression bins to create a close to uniform distribution subset of data, and use this subset to further tune the model to allow it to overcome it's tendency to bias toward the middle expression region that contains significantly more samples than low and high expression regions. Based on our evaluation metric, this step effectively increased the performance of the model, especially for prediction accuracy on low and high-expression genes.

4.1.3 Data Usage

In this challenge, we used the data mostly as it is, in the fine-tuning process, we sampled a subset from the entire set, as described in below training process, we use a 99-1 random train test split for both training and fine-tuning. The training dataset contains all 6.7 million samples provided in the challenge, then based on the bins of expression tiers, we sample k promoter sequences randomly from each expression bin (bins 1-18) to create a close to uniform distribution subset of data, by looking at the distribution of the data and testing different k values, we used k=25000. Other values of k we have tested include 3000, 15000, 50000, and 1000000. Both files used in the two-step training process have been included in GitHub.

4.1.4 Model

We use a basic transformer model (72), and specifically, a basic configuration of BERT (Bidirectional Encoder Representations from Transformers) (23). In addition to achieving state-of-the-art performance on natural language processing tasks, BERT has been successfully applied to prediction tasks on biological sequences (37), (16). This BERT base model uses 12 layers of transformer blocks with a hidden size of 768 and the number of self-attention heads as 12 and, in total, around 110M trainable parameters.

4.1.5 Training Procedure

Our training procedure is a two-step process based on provided labeled sequence training data. During both pieces of training, we treat each individual nucleotide as a word and split the entire promoter sequence, typically composed of at most 110 bases, into a sentence with at most 110 words, separated by spaces. Because some sequences are larger, we use 128 as our max input size, and any sequence with less than 128 words (nucleotides) is padded with "[PAD]" tokens to ensure uniform input

sizes. All training procedures and models were implemented in Python, our training environment involves PyTorch (50) and Hugging-face (75).

Tokenizer

We first train a word piece tokenizer (61) customized to the input, due to the special nature of inputs converted from SNA sequences, vocabularies are guaranteed to be ['A','G','G','T','N'], while adding five special tokens commonly included in transformer models: "[PAD]", "[UNK]", "[CLS]", "[SEP]", "[MASK]", which results in a fixed vocabulary size of 10 tokens. In future training only two of the special tokens – the mask token "[MASK]" and the pad token "[PAD]" – are used in further training; the other special tokens were included during initial development to support alternative model formulations.

Training and Fine-Tuning

We first train a BERT model from scratch, using only the entire data provided for the contest (approximately 6.7 million sequences), with a designed regression model, using an MSE (mean squared error) loss function and AdamW (40) optimization function. After training the model with all the samples and their expression labels, we further train the model with the fine-tuned subset. This subset used for fine-tuning is sampled equally from 18 expression bins of the full dataset to create a close-to-uniform distribution subset. The two-step training process shares the same infrastructure except that the fine-tuned step uses a Huber loss function instead of an MSE loss function, the detailed parameter for the layers and parameters are included in below table:

4.1.6 Result and Discussion

Despite limitations in hardware and timeline, our approach exceeded the published baseline model and made it to the final leader-board 45 teams at rank 33, similar

Table 4.1: Transformer Architecture and Model Configurations for Dream NLP Challenge

Model Architectures	BertForSequence Classification	**Train Epoches**	3
Hidden Layer Activation Function	GELU	**Per Device Batch Size**	36
Hidden Layers	12	**Initial Learning Rate**	1e-5
Hidden Layer Size	768	**Optimizer**	adamW
Dropout_prob	0.1	**Loss Function**	MSE/Huber
Max Position Embeddings	512		

deep learning approaches with different variants were adopted by the top teams, which further shows neutral network can be to successfully capture the patterns from sequence to gene expression, it is also worth noticing that, contrary to the increased performance attention mechanisms usual bring to natural language models, teams who introduced attention mechanism into their model on this dataset generally performed worse, which suggests that contextualized information of DNA sequences is either not related to their gene expressions, or the designed models were not capable of capturing the context information to make better predictions. To further this, we designed a similar context-based transformer and examined the degree of importance of contextualized information in genetic sequences when they are used to make predictions about gene expressions. We also developed a model that uses codon usage as the basic input unit instead of a single nucleotide, namely Codon-Bert.

4.2 CodonBERT: A Novel Approach to Understanding and Utilizing Codon Usage Patterns

4.2.1 Abstract

Codon usage bias refers to the uneven use of synonymous codons. For decades the Codon Adaption Index (CAI) has been the gold standard for modeling codon usage bias, especially with respect to predicting a gene's overall expression level. However, CAI is based on the assumption that a small training dataset – usually genes with higher expression – are the best data for determining "preferred" within a species. Although this assumption has largely held true for uni-cellular species such as yeast and E.coli, CAI has been unable to serve as a viable method for modeling codon usage in more complex organisms. Here we propose a novel deep learning framework to predict gene expression using a natural language processing scheme, namely CodonBERT. We show that our model is capable of making substantially better predictions of gene expression for a diverse collection of model organisms, especially for ones

where it is less obvious how to choose highly expressed genes for a CAI-based model. Our main contribution of this work is utilizing codon position information, which is often overlooked by other methods, always produces a better model of codon usage. Further, our more sophisticated framework that considers all genes (low, high, and all genes in-between) does a much better job estimating gene expression of the intermediate genes than prior approaches.

4.2.2 Introduction

Codon usage bias, which refers to using synonymous codons that encode for the same amino acid with different frequencies, has long been studied. Previous studies have shown that synonymous codon usage is likely shaped in part by positive selection pressure (e.g., (2)), and highly expressed genes generally show higher synonymous codon usage biases ((3), (11)). The observed positive correlation between gene expression levels and estimated codon usage bias is generally attributed to selection for translational efficiency ((64), (1)). Mutation (and thus random drift) is an alternative and less interesting force that also can shape codon usage biases (see (62) for a review).

In 1986, Sharp proposed the Codon Adaption Index (CAI) (64) under the assumption that, since optimal codons are more likely to be found in the highly expressed genes given their stronger selection pressure, codons that are "preferred" (higher synonymous biases) in a known collection of highly expressed genes are more likely to be transitionally efficient. This method proceeds as follows: calculate the observed frequency of each codon divided by its frequency expected under the assumption of equal usage, which is also called relative synonymous codon usage (RSCU). The CAI of a gene is simply the geometric mean of RSCUs for each codon in a gene divided by gene length. Genes with more optimal codons (higher RSCUs) are likely highly expressed genes, and therefore CAI is a relatively accurate measure of gene expression in the absence of mutation/random drift.

It follows that CAI can be a very effective method of measuring codon usage bias in single-celled microorganisms with short generation times such as yeast and *E. coli*. However, CAI can be less effective when it comes to multi-cellular species with weaker selection pressure ((59), (34)). Further, CAI tends to identify genes with higher expression levels much better than lowly expressed genes that are more susceptible to mutation and drift. Over the years, many methods based on CAI have been proposed ((59), (34), (36)), but most of these methods only show a small improvement over the original CAI methods with respect to predicting overall gene expression.

More recently, studies have shown that rare codon "clusters" are functionally important for protein activity and gene expression ((53), (19), (20), (7)). Because CAI is simply a geometric mean of RSCUs, the relative positions of codons are not factored in. Experimentally derived ribosome footprint data have shown that different stages in the protein translation process, such as initiation and elongation of distinct regions of genes, can have different impacts on protein synthesis ((56)). This suggests that the same codon in different regions of a gene can have a different impact on protein synthesis based in its "neighborhood" and therefore could impact overall gene expression in a more localized fashion.

To better understand potential codon usage patterns and their impact on observed gene expression, we propose a novel deep-learning framework to predict gene expression using a natural language processing scheme, namely Codon-BERT. We show that, compared to traditional codon usage bias models like CAI, our model is capable of making significantly better predictions of gene expression in a diverse collection of model organisms, especially for genes that are moderately expressed. Our median improvement is nearly 54% relative to CAI, and is as high as 1600% on our mammalian data, suggesting an approach looking at only the top X% of genes may lead to a poor model, especially for large, multi-cellular organisms considered here.

4.2.3 Data

To test our model's performance for expression level prediction, we collected expression measurements from six model organisms: *Saccharomyces cerevisiae* (Baker's yeast), *Escherichia coli*, *Caenorhabditis elegans* (Roundworm), *Drosophila melanogaster* (Common fruit fly), *Arabidopsis thaliana* (Thale cress), and *Mus musculus* (House mouse). Data sources for each of these species are included in Table 4.2.

For each species, only baseline control experiments are used (hence ignoring any other included experiments that involved alternate conditions). The median expression value is then used for each gene across all available experiments.

4.2.4 Method

The field of natural language processing (NLP) has seen the advent of a new family of neural networks known as transformers. Transformers (72) are in direct contrast to classical recurrent neural network techniques as they avoid autoregressive, sequential processing, utilizing a self-attention mechanism (8). Self-attention mechanisms learn contextual information about each input token, learning how each token is related to other tokens within the sequence. In 2018, BERT (Bidirectional Encoder Representations from Transformers) (23) was shown to outperform preexisting models across a large diversity of NLP benchmark problems. When compared to its predecessors, the major innovation of BERT comes from its inclusion of "bidirectionality," meaning a model learns a word and its context from left to right as well as from right to left, leading to a significant improvement in how researchers can model languages.

In computational biology, transformers, and specifically BERT, have seen success across a wide variety of applications for biological sequences. DNABERT (37) applied a modified BERT model to solve tasks such as identifying promoters, splice sites, and transcription factor binding sites. ProteinBERT (16) applied BERT

Table 4.2: Genbank accessions for all empirical expression data used in this study.

Species	**Accession Number**
Saccharomyces cerevisiae (strain S288c)	E-MTAB-8626
Escherichia coli (strain K12)	GSE1121
Caenorhabditis elegans	E-MTAB-2812
Drosophila melanogaster	E-GEOD-18068
Arabidopsis thaliana	E-MTAB-4202
Mus musculus	E-GEOD-52564

and a pretraining procedure to ultimately train a model on diverse protein-related tasks. Genetic sequences are not so different from natural languages, and a coding sequence, which usually ranges between 100-200 codons in length, can be treated as a sentence in a language with a vocabulary of 64, one for each of the possible codons. We hypothesized that NLP models—like BERT—could help discover patterns in codon usage because there are both left-to-right patterns (translational efficiency, see Introduction) as well as right-to-leftright to left (much less understood ribosomal pausing/co-translational folding patterns). For example, we have previously shown that less common codons (also called rare codons) that likely facilitate co-translational folding occur in similar regions within evolutionary-related proteins (19). Further, empirical analysis of ribosome footprint data – which is the best in vivo way to measure ribosome translation efficiency – can be modeled well using fixed-sized windows as small as 10bp (76). To the best of our knowledge, this study is the first attempt to use transformer-based NLP models to further decipher codon usage bias patterns, especially within the diverse set of model organisms considered here.

4.2.5 Pre-processing

The input to our CodonBERT model is gene sequences along with their labeled empirical expression values. Gene sequences are converted into vectors through the use of a tokenizer, which is a common technique in NLP that replaces each word in a sentence with an encoded token (from the vocabulary) that the model can understand as input. We used the base BERT (word piece) tokenizer from (23). We split the input sequence into 3-base-pair subsequences, which equate perfectly to codons with the gene and each constitutes a separate token. As is standard practice in training transformers, we allow for a padding token represented by "[PAD]"; this is used to fill the sequence to the maximum allowed length. One more token, "[UNK]" is used to represent any unknown tokens, e.g., errors in the sequencing input data that fail to determine the exact nucleotide.

Once the sequences are tokenized, samples are then randomly split into 80% training data and 20% testing data, the latter of which is unseen to the model during training. We also use the same training set for a CAI-based framework that calculates codon frequencies using only the most highly expressed genes (top 5%).

4.2.6 Model Architecture

Our model is implemented with Huggingface (75), a large open-source library of models commonly used in NLP. Importantly, we use an identical model to the base BERT model introduced in (23). The model uses 12 attention heads and 12 hidden layers in the encoding layers, allowing for a deep and large model that can learn complex patterns in the input sequences. After hidden layers and attention mechanisms, our configured BERT model uses GELU activation functions (33). All the additional tuned hyperparameters and specifics for the configuration of the model can be seen in Table 4.3.

4.2.7 Model Training and Evaluation

We train the model for 20 epochs on each dataset using a mean-squared error (MSE) loss function. The model is trained with an AdamW optimization function (41) and the base learning rate scheduler at which the initial learning rate is set to 1×10^{-5}. For regularization, we utilize dropout (67) on both hidden encoder layers and attention. Both dropout rates are set to 0.1, and the dropout mechanism is disabled during testing. Another method of regularization is imposed through LayerNorm (6) layers, through which we use an ϵ value of 1×10^{-12}. Other training configurations are summarized in Table 4.3. Finally, we use the Spearman's rank correlation coefficient between model predictions and empirical expression values to assess each individual model's performance.

Code and data used to train the model, and instructions on running the model are hosted on github at github.com/zlu-volyote/CodonBert.

Table 4.3: Transformer Architecture and Model Configurations for CodonBert

Model Architectures	BertForSequence Classification	**Train Epochs**	20
Hidden Layer Activation Function	GELU	**Per Device Batch Size**	2
Hidden Layers	12	**Hidden Layer Size**	768
Attention Heads	12	**Attention Dropout Rate**	0.1
Initial Learning Rate	1e-5	**Optimizer**	AdamW
Dropout Rate	0.1	**Loss Function**	MSE
Max Position Embeddings	512	**LayerNorm** ϵ	1e-12

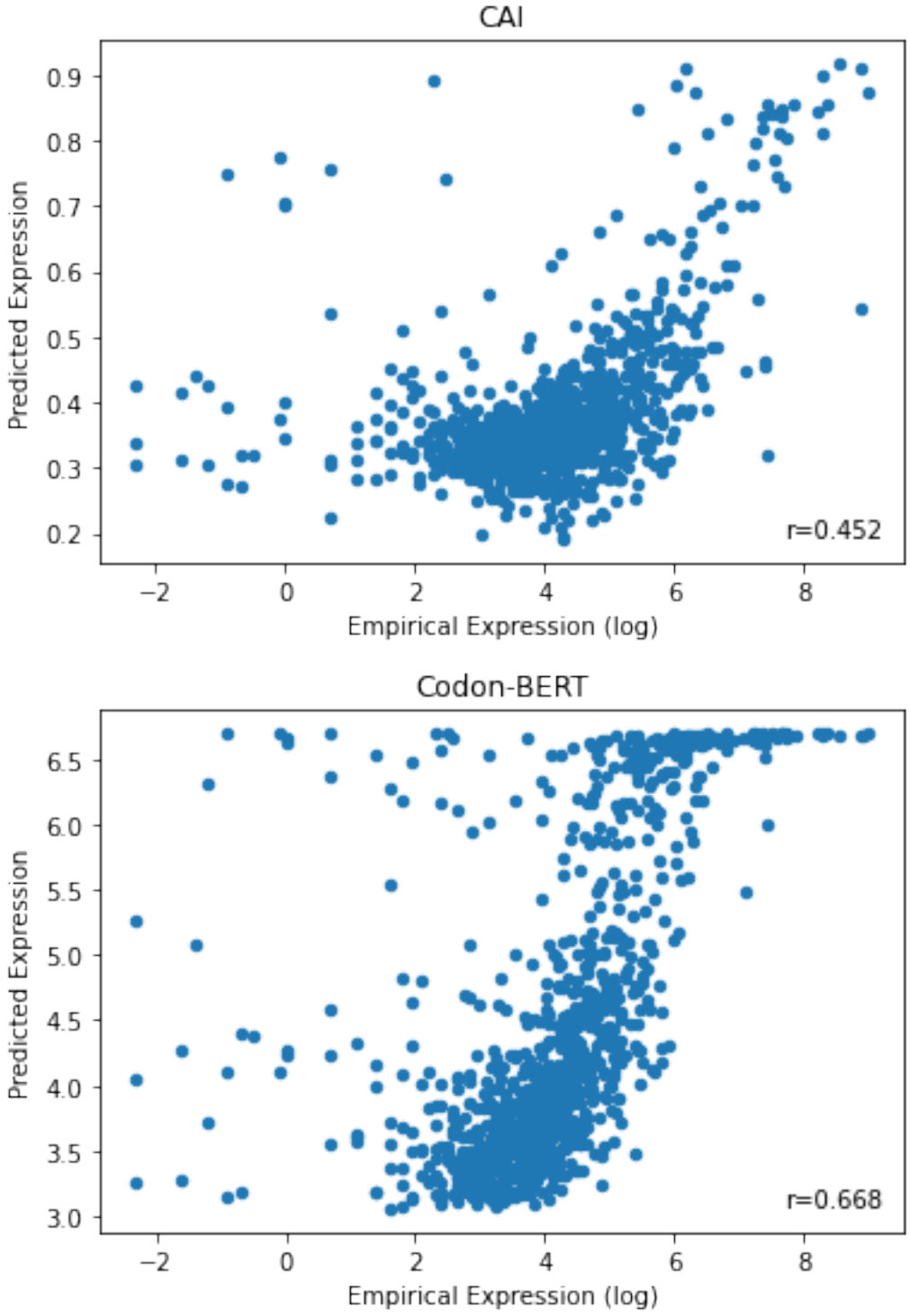

Figure 4.1: Scatter plot of *Saccharomyces cerevisiae* S288c gene expression predictions made by CAI/Codon-BERT with respect to log-transformed empirical expression measurements. We use Spearman r for ranked correlation (see Methods).

4.2.8 Results

We first considered *Saccharomyces cerevisiae* (yeast) since we and others have shown a strong correlation between observed codon usage biases and overall gene expression (42), (76), (64), (3), (62). This dataset, therefore, is expected to have one of the highest – if not the highest – CAI-based gene expression correlation. As shown in Figure 4.1, our predictions of gene expression levels in *Saccharomyces cerevisiae* S288c generated by our model (r=0.668) has a stronger correlation than CAI (r=0.452).

We further zoom into different tiers of expression as illustrated in Figure 4.2. For genes with lower or higher expression levels, both methods perform either relatively poorly (r=0.22 vs r=0.235) or relatively well (r=0.731 vs r=0.636), respectively, consistent with prior studies in *Saccharomyces cerevisiae* (64), (42). Significantly, for the majority of genes that are neither significantly highly expressed nor lowly expressed (i.e. moderately expressed), our model significantly outperforms CAI (r=0.539 vs r=0.258).

We then perform the same benchmark for five other datasets from different model organisms, and the summary table of Spearman ranked correlations between the models and empirical expression data are shown in Table 4.4. In strong support of our underlying hypothesis that NLP can better determine cryptic codon usage patterns than a more aggregate-based method, our model significantly outperforms CAI across all datasets considered, especially for more complex organisms such as *Arabidopsis thaliana* and *Mus musculus*. In these large multi-cellular organisms (a plant and animal) our model yields a reasonable performance while CAI is barely correlated with observed gene expression data generated from these species.

4.2.9 Discussion

As mentioned earlier in the Introduction, many alternative approaches compare their results (often correlations) relative to CAI using genes from either yeast S288c or *E. coli* K12, which were both considered here as well. We therefore performed a

Table 4.4: Model benchmark results across selected model organisms. Consistent with prior results (e.g., (42)), expression data-based CAI works poorly on Arabidopsis and mouse while codonBERT performs relatively much better based on a more comprehensive (but codon diversified) training dataset.

Species	**CodonBert**	**CAI**	**Percent Improvement**
Escherichia coli	0.615	0.494	24.5
Saccharomyces cerevisiae	0.668	0.452	47.8
Caenorhabditis elegans	0.684	0.428	59.8
Drosophila melanogaster	0.523	0.392	33.4
Arabidopsis thaliana	0.522	0.157	223.5
Mus musculus	0.427	0.025	1600

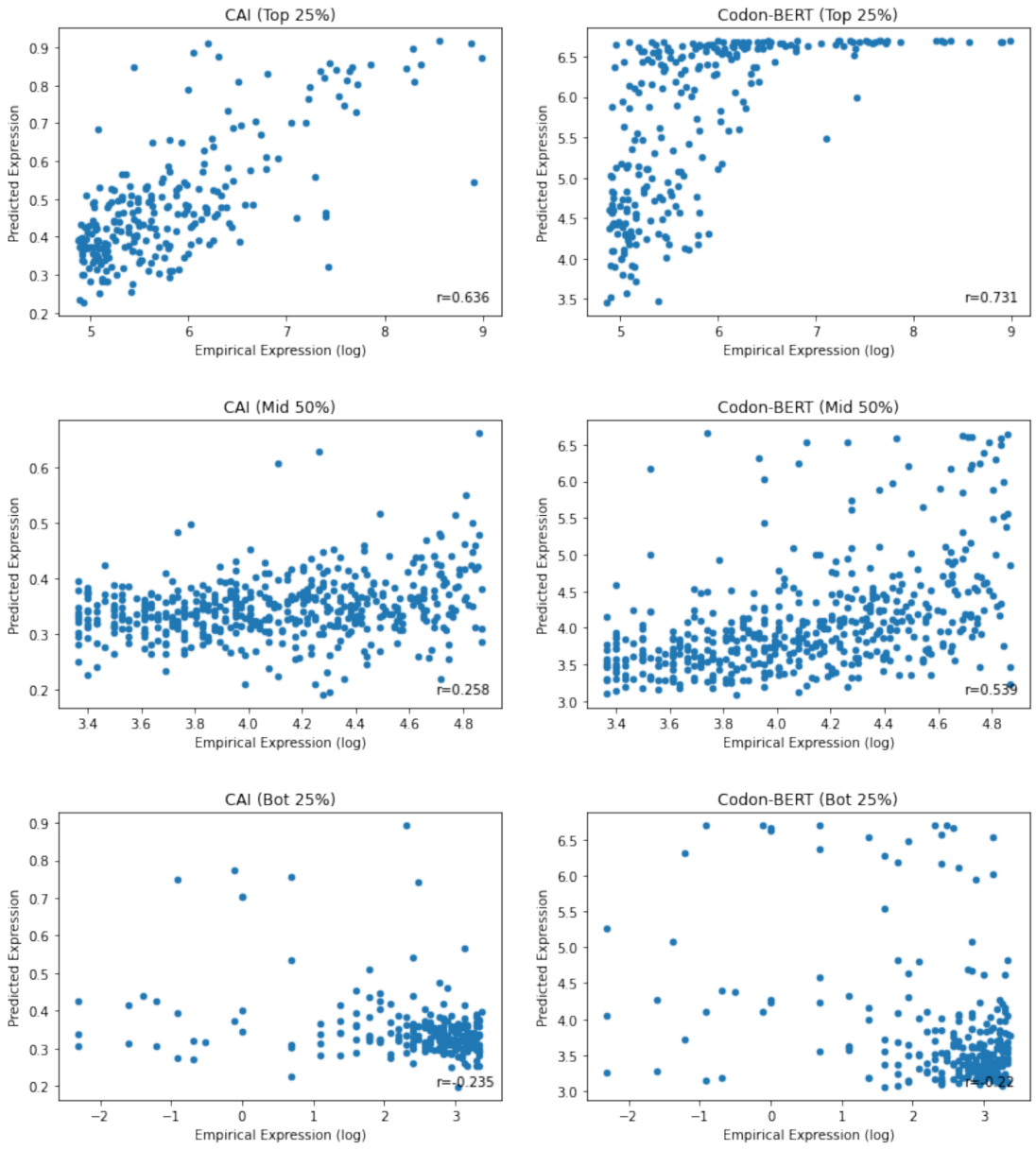

Figure 4.2: Scatter plot of model predictions with empirical expression data across different ranges of expression.

literature search to compare our model performance with prior gene-based approaches for predicting overall gene expression.

Relative Codon Usage Bias (RCBS) is an improved version of CAI based on a bespoke set of 123 highly expressed genes from yeast genome and achieved roughly a 3% gain in performance (an increased of Pearson correlation from 0.5987 to 0.6163 (22)) and, when performing the same analysis with another biologically-informed set of 45 highly expressed genes from *E. coli*, they achieved roughly a 15% gain in performance (a final Pearson correlation of 0.7082 (58)). We note that CodonBERT already has the same correlation as the initial highly curated set of genes from *E. coli* did (roughly 0.615) and outperforms RCBS and their highly curated set of genes for yeast gene prediction by 8.4%.

Renana Sabi and Tamir Tuller proposed stAI as an improved version of tAI (59)(60), which is often preferred by protein biochemists given that it more directly models protein elongation by leveraging information about the overall tRNA pool in an organism. Although this method is not broadly applicable since biological experiments are required to parameterize the model, it provides an alternative, biologically-informed view of modeling gene expression.

Further, Sabi and Tuller extensively evaluated tAI and stAI using empirical protein abundance data, which is only indirectly measured based on sequencing-based gene expression analysis. They showed that tAI has a Spearman correlation of r=0.5032 with empirical gene expression measurements in *E. coli*, r=0.3328 in *Arabidopsis thaliana*, r=0.0919 in *C. elegans* , r=0.4878 in *D. melanogaster*, and 0.6915 in *S. cerevisiae*. stAI has a spearman correlation of 0.5493, 0.3762, 0.0956, 0.5001, and 0.5802, respectively. CodonBERT does as well or better across all datasets with the exception of *C. elegans* and Arabidopsis where our model has a correlation of 0.684 and 0.522 vs. their best models at 0.0956 and 0,3762, respectively. We conclude that CodonBERT is comparable and often better than even tRNA-pool-based predictions.

We are aware that one of the largest drawbacks of applying deep learning approaches to biology applications is the lack of interpretability. In this specific application, we train our model based on codon vectors along with their labeled expression values, and the neural network adjusts the weights between layers based on a designated objective function to minimize loss. As typical, we then optimize the hyper-parameters to reach a good fit and present a model that significantly outperforms existing codon-based methods without any clear biological insight on what is driving the observed increase in performance.

Our objective in designing a preliminary model using NLP-transformers, BERT in particular, is to leverage positional information of codons that largely has been ignored in prior modeling of codon usage bias. For example, CAI is a geometric mean of the RSCU value for each codon, so the primary consideration is "Does this gene have more preferred/optimal codons than expected?" In contrast, BERT is capable of using the positional information of words (here codons) by introducing contextualized attention. Although this can result in a 17X improved correlation in analyzing codon usage in a mammalian organism (mouse; see Table III), it largely is comparable to CAI-based results on yeast and *E. coli* when CAI is trained on less noisy input data (see (22) and above). Therefore, to more fairly evaluate whether positional information of codons can lead to performance gains in predicting overall gene expression, we randomly shuffle the codons in the original sequences, keep the expression label unchanged, and then use the shuffled codon sequences to train a new model. Doing this shuffling should effectively remove almost all local positional information of codon "neighborhoods" while keeping other parameters of the model the same, e.g., each gene has the same tokens just in a different order.

As shown in Figure 4.3, model performance uniformly drops for all datasets after the input codons for each coding sequence are randomly shuffled. We can also see that performance drop for certain species such as *Mus musculus* and *Arabidopsis thaliana* is much more significant than it is in *Saccharomyces cerevisiae* and *Escherichia coli*, which suggests that positional information of codons are more important when it

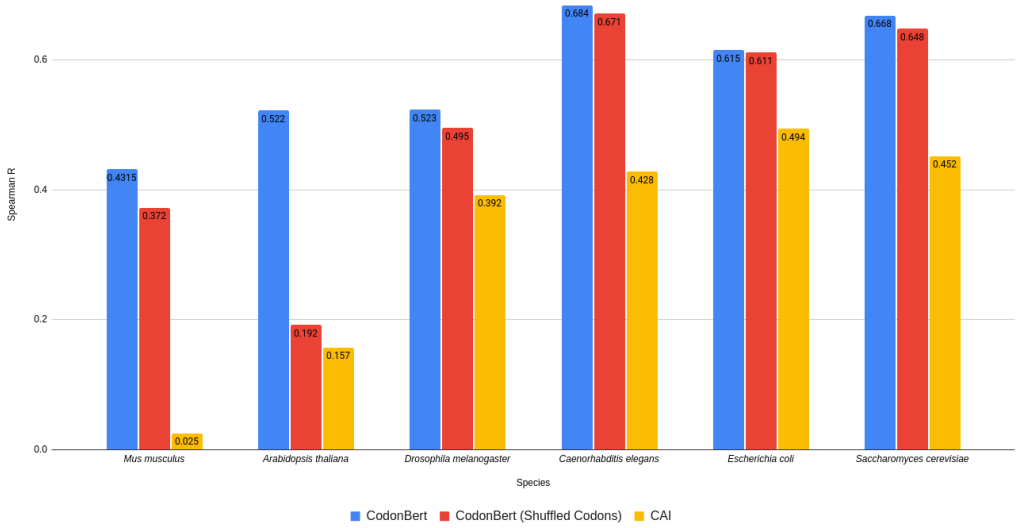

Figure 4.3: Performance of CodonBERT After Shuffling Input Codons

comes to predicting overall gene expression in some of the large, multi-cellular species since *C. elegans* has roughly 1,000 cells (4). The effect of positional information is most pronounced in Arabidopsis (2.7X better performance), which is the only plant considered. This either may be a result of an under-resolved model—not uncommon for deep learning NLP–or less likely some unique codon pattern that can better inform overall gene prediction in plants due to some biological process. We leave this to future work.

Interestingly, we note that even after shuffling codons prior to training our model, codonBERT still outperforms CAI for each dataset, including Arabidopsis shuffled and especially our mouse-derived dataset (see Figure 4.2). We attribute this performance difference to our NLP-based model's ability to better grasp codon usage patterns across different expression tiers, without explicitly needing a subset (aka top 5%) of the training.

4.2.10 Conclusion

Here, we propose a deep learning-inspired codon-based expression prediction model that outperforms currently prominent methods such as CAI, tAI, stAI, and RCBS. We also show that position information of codons can play at least a minor role in the gene translation process and always has a better overall performance relative to shuffling the tokens for each gene considered. Interestingly, in contrast to prior work we see relatively better performance on complex multi-cellular organisms (mouse and Arabidopsis) relative to traditional approaches.

Bibliography

[1] (2001). Gene expression and molecular evolution. *Current Opinion in Genetics Development*, 11(6):660–666. 45

[2] Akashi, H. (1995). Inferring weak selection from patterns of polymorphism and divergence at 'silent' sites in Drosophila DNA. *Genetics*, 139(2):1067–1076. 45

[3] Akashi, H. (2003). Translational selection and yeast proteome evolution. *Genetics*, 164(4):1291–1303. 45, 53

[4] Alberts, B., Bray, D., Lewis, J., Raff, M., Roberts, K., and Watson, J. (2002). *Molecular Biology of the Cell*. Garland, 4th edition. 59

[5] Athey, J., Alexaki, A., Osipova, E., Rostovtsev, A., Santana-Quintero, L. V., Katneni, U., Simonyan, V., and Kimchi-Sarfaty, C. (2017). A new and updated resource for codon usage tables. *BMC Bioinformatics*, 18(1). 3

[6] Ba, J. L., Kiros, J. R., and Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*. 50

[7] Babjac, A., Li, J., and Emrich, S. (2021). Fine-grained synonymous codon usage patterns and their potential role in functional protein production. In *2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 2187–2193. 46

[8] Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. *International Conference on Learning Representations*. 47

[9] Bartell, S. M. (2006). Biomarkers, bioindicators, and ecological risk assessment—a brief review and evaluation. *Environmental Bioindicators*, 1(1):60–73. 10

[10] Beer, M. A. and Tavazoie, S. (2004). Predicting gene expression from sequence. *Cell*, 117(2):185–198. 37, 38

[11] Behura, S. K. and Whitfield, C. W. (2010). Correlated expression patterns of microRNA genes with age-dependent behavioural changes in honeybee. *Insect Molecular Biology*, 19(4):431–439. 45

[12] Bernabe, B. P., Cunningham, J. L., Tussing-Humphreys, L., Carroll, I., Meltzer-Brody, S., Maki, P. M., Gilbert, J. A., and Kimmel, M. (2020). Predicting microbiomes through a deep latent space. *Bioinformatics*. 9

[13] Bernardo-Cravo, A. P., Schmeller, D. S., Chatzinotas, A., Vredenburg, V. T., and Loyau, A. (2020). Environmental factors and host microbiomes shape host–pathogen dynamics. *Trends in Parasitology*, 36(7):616–633. 10

[14] Bionetworks, S. (2022). Predicting gene expression using millions of random promoter sequences. 40

[15] Bloom, J. S., Khan, Z., Kruglyak, L., Singh, M., and Caudy, A. A. (2009). Measuring differential gene expression by short read sequencing: quantitative comparison to 2-channel gene expression microarrays. *BMC Genomics*, 10(1):221. 27, 30

[16] Brandes, N., Ofer, D., Peleg, Y., Rappoport, N., and Linial, M. (2022). ProteinBERT: a universal deep-learning model of protein sequence and function. *Bioinformatics*, 38(8):2102–2110. 41, 47

[17] Callahan, B. J., McMurdie, P. J., and Holmes, S. P. (2017). Exact sequence variants should replace operational taxonomic units in marker-gene data analysis. *The ISME Journal*, 11(12):2639–2643. 6, 12

[18] Chaney, J. L., Steele, A., Carmichael, R., Rodriguez, A., Specht, A. T., Ngo, K., Li, J., Emrich, S., and Clark, P. L. (2017a). Widespread position-specific conservation of synonymous rare codons within coding sequences. *PLOS Computational Biology*, 13(5). 7, 37, 38

[19] Chaney, J. L., Steele, A., Carmichael, R., Rodriguez, A., Specht, A. T., Ngo, K., Li, J., Emrich, S., and Clark, P. L. (2017b). Widespread position-specific conservation of synonymous rare codons within coding sequences. *PLOS Computational Biology*, 13(5):1–19. 46, 49

[20] Chartier, M., Gaudreault, F., and Najmanovich, R. (2012). Large-scale analysis of conserved rare codon clusters suggests an involvement in co-translational molecular recognition events. *Bioinformatics*, 28(11):1438–1445. 46

[21] Chen, J., King, E., Deek, R., Wei, Z., Yu, Y., Grill, D., and Ballman, K. (2017). An omnibus test for differential distribution analysis of microbiome sequencing data. *Bioinformatics*, 34(4):643–651. 16

[22] Das, S., Roymondal, U., and Sahoo, S. (2009). Analyzing gene expression from relative codon usage bias in yeast genome: a statistical significance and biological relevance. *Gene*, 443(1-2):121–131. 56, 57

[23] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*. 40, 41, 47, 49, 50

[24] Dunne, J. A., Saleska, S. R., Fischer, M. L., and Harte, J. (2004). Integrating experimental and gradient methods in ecological climate change research. *Ecology*, 85(4):904–916. 11

[25] Edgar, R. C. (2010). Search and clustering orders of magnitude faster than blast. *Bioinformatics*, 26(19):2460–2461. 12

[26] Effrosynidis, D. and Arampatzis, A. (2021). An evaluation of feature selection methods for environmental data. *Ecological Informatics*, 61:101224. 13, 17

[27] Eisenberg, E. and Levanon, E. Y. (2014). Corrigendum to: Human housekeeping genes, revisited. *Trends in Genetics*, 30(3):119–120. 30

[28] Ellison, A. M., Bank, M. S., Clinton, B. D., Colburn, E. A., Elliott, K., Ford, C. R., Foster, D. R., Kloeppel, B. D., Knoepp, J. D., Lovett, G. M., Mohan, J., Orwig, D. A., Rodenhouse, N. L., Sobczak, W. V., Stinson, K. A., Stone, J. K., Swan, C. M., Thompson, J., Von Holle, B., and Webster, J. R. (2005). Loss of foundation species: consequences for the structure and dynamics of forested ecosystems. *Frontiers in Ecology and the Environment*, 3(9):479–486. 10

[29] Emmert-Streib, F., Yang, Z., Feng, H., Tripathi, S., and Dehmer, M. (2020). An introductory review of deep learning for prediction models with big data. *Frontiers in Artificial Intelligence*, 3. 13

[30] Ghannam, R. B. and Techtmann, S. M. (2021). Machine learning applications in microbial ecology, human microbiome studies, and environmental monitoring. *Computational and Structural Biotechnology Journal*, 19:1092–1107. 9

[31] Gilchrist, M. A., Chen, W.-C., Shah, P., Landerer, C. L., and Zaretzki, R. (2015). Estimating gene expression and codon-specific translational efficiencies, mutation biases, and selection coefficients from genomic data alone ‡. *Genome Biology and Evolution*, 7(6):1559–1579. 25, 26

[32] Gouy, M. and Gautier, C. (1982). Codon usage in bacteria: correlation with gene expressivity. *Nucleic Acids Research*, 10(22):7055–7074. 27

[33] Hendrycks, D. and Gimpel, K. (2016). Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*. 50

[34] Henry, I. and Sharp, P. M. (2006). Predicting gene expression level from codon usage Bias. *Molecular Biology and Evolution*, 24(1):10–12. 46

[35] Hershberg, R. and Petrov, D. A. (2008). Selection on codon bias. *Annual Review of Genetics*, 42(1):287–299. 27

[36] Jansen, R., Bussemaker, H. J., and Gerstein, M. (2003). Revisiting the codon adaptation index from a whole-genome perspective: analyzing the relationship between gene expression and codon occurrence in yeast using a variety of models. *Nucleic Acids Research*, 31(8):2242–2251. 46

[37] Ji, Y., Zhou, Z., Liu, H., and Davuluri, R. V. (2021). DNABERT: pre-trained Bidirectional Encoder Representations from Transformers model for DNA-language in genome. *Bioinformatics*, 37(15):2112–2120. 41, 47

[38] Kivlin, S. N., Mann, M. A., Lynn, J. S., Kazenel, M. R., Taylor, D. L., and Rudgers, J. A. (2022). Grass species identity shapes communities of root and leaf fungi more than elevation. *ISME Communications*, 2(1). 11

[39] Lloyd-Price, J., Mahurkar, A., Rahnavard, G., Crabtree, J., Orvis, J., Hall, A. B., Brady, A., Creasy, H. H., McCracken, C., Giglio, M. G., and et al. (2017). Strains, functions and dynamics in the expanded human microbiome project. *Nature*, 550(7674):61–66. 12

[40] Loshchilov, I. and Hutter, F. (2019a). Decoupled weight decay regularization. In *International Conference on Learning Representations*. 42

[41] Loshchilov, I. and Hutter, F. (2019b). Decoupled weight decay regularization. *International Conference on Learning Representations*. 50

[42] Lu, Z., Gilchrist, M., and Emrich, S. (2020). Analysis of mutation bias in shaping codon usage bias and its association with gene expression across species. *EPiC Series in Computing*. ix, 53, 54

[43] Marcos-Zambrano, L. J., Karaduzovic-Hadziabdic, K., Loncar Turukalo, T., Przymus, P., Trajkovik, V., Aasmets, O., Berland, M., Gruca, A., Hasic, J., Hron, K., Klammsteiner, T., Kolev, M., Lahti, L., Lopes, M. B., Moreno, V., Naskinova, I., Org, E., Paciência, I., Papoutsoglou, G., Shigdel, R., Stres, B., Vilne, B., Yousef, M., Zdravevski, E., Tsamardinos, I., Carrillo de Santa Pau, E., Claesson, M. J., Moreno-Indias, I., and Truu, J. (2021). Applications of machine learning in human microbiome studies: A review on feature selection, biomarker identification, disease prediction and treatment. *Frontiers in Microbiology*, 12:313. 9

[NIH] NIH. The cost of sequencing a human genome. x, 8

[45] Nurk, S., Koren, S., Rhie, A., Rautiainen, M., Bzikadze, A. V., Mikheenko, A., Vollger, M. R., Altemose, N., Uralsky, L., Gershman, A., and et al. (2022). The complete sequence of a human genome. *Science*, 376(6588):44–53. 1

[46] O, F., J, H., X, B., Itzkowitz, S., I, P., and A., B. (2015). Improved OTU-picking using long-read 16s rRNA gene amplicon sequencing and generic hierarchical clustering. *Microbiome*. 13

[47] Oh, M. and Zhang, L. (2020). Deepmicro: Deep representation learning for disease prediction based on microbiome data. *Scientific Reports*, 10(1). 12

[48] Pasolli, E., Truong, D. T., Malik, F., Waldron, L., and Segata, N. (2016a). Machine learning meta-analysis of large metagenomic datasets: Tools and biological insights. *PLOS Computational Biology*, 12(7). 12

[49] Pasolli, E., Truong, D. T., Malik, F., Waldron, L., and Segata, N. (2016b). Machine learning meta-analysis of large metagenomic datasets: Tools and biological insights. *PLOS Computational Biology*, 12:1–26. 17

[50] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and

Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R., editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc. 42

[51] Paulson, J., Stine, O., Bravo, H., and Pop, M. (2014). Differential abundance analysis for microbial marker-gene surveys. *Nature Methods*. 15

[52] Pepin, N. and Losleben, M. (2002). Climate change in the colorado rocky mountains: Free air versus surface temperature trends. *International Journal of Climatology*, 22(3):311–329. 11

[53] Perach, M., Zafrir, Z., Tuller, T., and Lewinson, O. (2021). Identification of conserved slow codons that are important for protein expression and function. *RNA Biology*, 18(12):2296–2307. 46

[54] Piryonesi, S. M. and El-Diraby, T. E. (2020). Data analytics in asset management: Cost-effective prediction of the pavement condition index. *Journal of Infrastructure Systems*, 26(1):04019036. 11

[55] Reis, M. D. (2004). Solving the riddle of codon usage preferences: a test for translational selection. *Nucleic Acids Research*, 32(17):5036—-5044. 24, 25

[56] Riba, A., Nanni, N. D., Mittal, N., Arhné, E., Schmidt, A., and Zavolan, M. (2019). Protein synthesis rates and ribosome occupancies reveal determinants of translation elongation rates. *Proceedings of the National Academy of Sciences*, 116(30):15023–15032. 46

[57] Rodriguez, A., Wright, G., Emrich, S., and Clark, P. L. (2017). codon usage and its impact on protein folding. *Protein Science*, 27(1):356–362. 38

[58] Roymondal, U., Das, S., and Sahoo, S. (2009). Predicting Gene Expression Level from Relative Codon Usage Bias: An Application to Escherichia coli Genome. *DNA Research*, 16(1):13–30. 56

[59] Sabi, R. and Tuller, T. (2014a). Modelling the efficiency of codon–trna interactions based on codon usage bias. *DNA Research*, 21(5):511–526. 46, 56

[60] Sabi, R. and Tuller, T. (2014b). Modelling the Efficiency of Codon–tRNA Interactions Based on Codon Usage Bias. *DNA Research*, 21(5):511–526. 56

[61] Schuster, M. and Nakajima, K. (2012). Japanese and korean voice search. In *2012 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5149–5152. IEEE. 42

[62] Shah, P. and Gilchrist, M. A. (2011). Explaining complex codon usage patterns with selection for translational efficiency, mutation bias, and genetic drift. *Proceedings of the National Academy of Sciences*, 108(25):10231–10236. 45, 53

[63] Sharp, P. M. and Li, W.-H. (1987a). The codon adaptation index-a measure of directional synonymous codon usage bias, and its potential applications. *Nucleic Acids Research*, 15(3):1281–1295. 24

[64] Sharp, P. M. and Li, W.-H. (1987b). The codon adaptation index-a measure of directional synonymous codon usage bias, and its potential applications. *Nucleic Acids Research*, 15(3):1281–1295. 45, 53

[65] Sherstinsky, A. (2020). Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nonlinear Phenomena*, 404:132306. 13

[66] Siddig, A. A., Ellison, A. M., Ochs, A., Villar-Leeman, C., and Lau, M. K. (2016). How do ecologists select and use indicator species to monitor ecological change? Insights from 14 years of publication in ecological indicators. *Ecological Indicators*, 60:223–230. 10

[67] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958. 50

[68] Tan, P. K. (2003). Evaluation of gene expression measurements from commercial microarray platforms. *Nucleic Acids Research*, 31(19):5676–5684. 27, 30

[69] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288. 17

[70] Tong, Q., Cui, L.-Y., Hu, Z.-F., Du, X.-P., Abid, H. M., and Wang, H.-B. (2020). Environmental and host factors shaping the gut microbiota diversity of brown frog *Rana dybowskii*. *Science of The Total Environment*, 741:140142. 10

[71] Trotta, E. (2013). Selection on codon bias in yeast: a transcriptional hypothesis. *Nucleic Acids Research*, 41(20):9382–9395. 27

[72] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30. 41, 47

[73] Wallace, E. W., Airoldi, E. M., and Drummond, D. A. (2013). Estimating selection on synonymous codon usage from noisy experimental data. *Molecular Biology and Evolution*, 30(6):1438–1453. 25

[74] WATSON, J. D. and CRICK, F. H. (1953). Molecular structure of nucleic acids: A structure for deoxyribose nucleic acid. *Nature*, 171(4356):737–738. iv, 1

[75] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al. (2020). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45. 42, 50

[76] Wright, G., Rodriguez, A., Li, J., Clark, P. L., Milenković, T., and Emrich, S. J. (2020). Analysis of computational codon usage models and their association with translationally slow codons. *PlOS ONE*, 15(4):1–15. 49, 53

[77] Yuan, Y., Guo, L., Shen, L., and Liu, J. S. (2007). Predicting gene expression from sequence: A reexamination. *PLoS Computational Biology*, 3(11). 37

Vita

Hi, I'm Zhixiu Lu, a researcher interested in machine learning, deep learning, life science applciations and bioinformatics. I'm currently studying in the Computer Science Department in University of Tennessee at Knoxville under the supervision of Dr. Scott Emrich.