Learning the native-like codons with a 5'UTR and secondary RNA structure aided species-informed transformer model

Qiuyue Hu[1,2], Xiaolin Tian[3], Yuanning Li[4], Jintao Meng[1], Shen Wang[5], Jingjing Guo[6], Liangzhen Zheng[7,5,*], and Yanjie Wei[1,*]

[1]Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, 1008 Xueyuan Ave, Shenzhen, China
[2]University of Chinese Academy of Sciences, Beijing, China
[3]Institute of Marine Science and Technology, Shandong University, Qingdao, China
[4]Laboratory for Marine Biology and Biotechnology, Qingdao Marine Science and Technology Center, Qingdao, China
[5]Shanghai Zelixir Biotech Co. Ltd., 298 Xiangke Road, Shanghai, 201210, China
[6]Centre in Artificial Intelligence Driven Drug Discovery, Faculty of Applied Sciences, Macao Polytechnic University, Macao SAR, China
[7]Shenzhen Zelixir Biotech Co. Ltd, Hengtaiyu Building, Fenghuang Street, Guangming District, Shenzhen, China
[*]Corresponding author: Liangzhen Zheng and Yanjie Wei, zhenglz@zelixir.com and yj.wei@siat.ac.cn

Abstract

Efficient protein expression across heterologous hosts remains a major challenge in synthetic biology, largely due to species-specific differences in codon usage and regulatory sequence context. A key difficulty lies in reconstructing the codon landscape of the target expression system within a foreign host with a native-like codon preference. To address this, we present TransCodon, a Transformer-based deep learning model that leverages both 5' untranslated regions (5'UTRs) and coding sequences (CDS), along with explicit species identifiers and RNA secondary structure information, to learn nuanced codon usage patterns across diverse organisms. By incorporating multisource genomic data and modeling sequence dependencies in a masked language modeling paradigm, TransCodon effectively captures both local and global determinants of codon preference. Our experiments demonstrate that integrating species-level information during training significantly improves the model's ability to predict optimal synonymous codons for enhanced cross-species protein expression. TransCodon consistently outperforms existing codon optimization tools in multiple evaluation metrics. More importantly it identifies native-like codons with less divergence from natural sequences compared to other methods. For a high abundant protein, TransCodon could capture more low-frequency codons which are often omitted by other deep learning-based method. The results thus indicate that TransCodon as a robust codon language model has the potential for designing genes with high translational efficiency in target hosts.

1 Introduction

A DNA sequence consists of four nucleotides—adenine (A), thymine (T), guanine (G), and cytosine (C). Triplets of non-overlapping nucleotides, known as codons, encode specific amino acids in a protein sequence. Multiple codons can encode the same amino acid; these are referred to as synonymous codons and are a fundamental basis for the degeneracy of the genetic code. However, the usage frequency of these synonymous codons varies significantly across species, influenced by factors such as intracellular tRNA abundance, translational dynamics, protein folding regulation, and evolutionary constraints[1, 2]. This phenomenon, referred to as codon usage bias, is a species-specific characteristic that plays a critical role in determining the efficiency of gene expression.

Codon optimization, the process of redesigning DNA sequences to align synonymous codon usage with the preferences of a target host organism, is a crucial step in heterologous protein expression. With the advent of affordable de novo DNA synthesis and advances in protein engineering[3–7], the demand for codon optimization has surged. However, the combinatorial complexity of synonymous codon arrangements makes exhaustive search approaches impractical. Traditional codon optimization methods often rely on selecting the most frequent codons or employ heuristic rules that fail to con-

sider broader biological context[8]. Strategies focused on optimizing the Codon Adaptation Index (CAI)[9] and favoring the most common codons have shown some success in specific cases[10, 11]. However, such approaches do not guarantee effective expression and can lead to unintended consequences, including excessive cellular resource consumption, protein misfolding, ribosomal stalling, and even host toxicity[12]. Codon harmonization methods attempt to mimic natural codon usage patterns to reconstruct the codon landscape in heterologous hosts[13, 14]. Yet, their reliance on heuristic rules restricts their ability to capture deeper biological relationships, often resulting in a superficial reconstruction of codon landscapes that neglects important factors such as translational dynamics and mRNA stability.

Furthermore, many existing approaches neglect the functional role of untranslated regulatory regions, particularly the 5' untranslated region (5' UTR) of mRNA. Located upstream of the start codon, the 5' UTR plays a crucial role in regulating protein expression levels[8, 15–17]. It influences mRNA stability, ribosome binding efficiency, and translation initiation. The structural features of the 5' UTR, such as secondary structure, the sequence context around the start codon, and the ribosome binding site (RBS), are closely linked to ribosomal accessibility and overall translational efficiency. Failing to consider the 5' UTR during codon optimization may result in inadequate modeling of translational regulation mechanisms. Additionally, the RNA secondary structure within the coding sequence (CDS) significantly affects translation[18]. Local secondary structures modulate ribosome elongation and pausing, thereby impacting protein folding and function. Strong secondary structures can cause ribosome stalling and reduce translation efficiency. Therefore, effective codon optimization should integrate both 5' UTR and CDS structural features to better capture the dynamics of gene expression.

Recent advances in deep learning have introduced powerful alternatives for codon optimization. Neural network models have shown strong capabilities in learning complex sequence dependencies and capturing the latent "language" of codon usage[15, 19–21]. Nevertheless, current deep learning-based approaches face several limitations: they are often trained on small-scale Datasets, tailored to a single host species (e.g., Escherichia coli), and lack the flexibility required for broader adoption in synthetic biology. Additionally, in the context of cross-species protein expression, many models focus excessively on optimizing the Codon Adaptation Index (CAI), which further deviates from natural codon usage patterns[20, 21]. CodonTransformer[21], a codon optimization model based on the BERT architecture, incorporates species-specific information and is trained on approximately one million sequences. However, it does not consider the regulatory effects of the 5' UTR, and its use of the BigBird model tends to produce sequences with overly high CAI values, potentially underestimating the importance of rare codons. These limitations hinder its generalizability and biological relevance.

In this study, we present TransCodon, a novel species-aware codon optimization model based on the BERT (Bidirectional Encoder Representations from Transformers) architecture. Our model integrates multiple sequence components, including the 5' untranslated region (5' UTR) and coding sequence (CDS), and explicitly incorporates species-specific information and RNA secondary structure features to capture codon usage preferences across diverse organisms. By leveraging the sequence modeling capabilities of the Transformer architecture and utilizing a large-scale dataset comprising 5.5 million gene sequences from 1436 species, TransCodon effectively identifies both global and local patterns of codon usage, enabling a more accurate reconstruction of the codon landscape. This facilitates robust optimization of heterologous genes to enhance protein expression. Experimental results demonstrate that TransCodon consistently outperforms traditional methods and recent machine learning-based approaches across multiple evaluation metrics. Further performance improvements can be achieved through fine-tuning, validating its effectiveness and broad applicability in synthetic biology.

2 Results

2.1 A Transformer-based framework for cross-species codon optimization

Due to the inherently sequential nature of genomic data, recurrent neural networks (RNNs) and Transformer-based architectures have demonstrated remarkable efficacy in capturing the complex intrinsic features of biological sequences[22–26], thereby making them highly suitable for codon optimization tasks[20, 21, 27, 28]. In prior research, codon optimization has often been formulated as either a sequence labeling or machine translation problem. Both approaches typically use the target protein sequence as input and leverage bidirectional encoder models to generate context-aware embeddings for each amino acid, employing architectures such as bidirectional long short-term memory networks (Bi-LSTM) or bidirectional Transformer encoders[29, 30].

Given the large-scale training data used in this study, we employ a more expressive bidirectional Transformer encoder architecture, trained using a masked language modeling (MLM) approach, which has been successfully applied for pretraining protein amino acid sequences[22, 23]. A key challenge lies in defining the mapping from amino acids to codons. CodonTransformer[21] uses a composite vocabulary where each token represents an amino acid-codon pair, such as A-GCC to denote alanine encoded by the codon GCC. In contrast, our study opts for a finer-grained vocabulary based solely

on nucleotides (A, T, C, and G), aligning with the design of most DNA/RNA language models[31, 32]. This choice is based on the observation that synonymous codons(different codons encoding the same amino acid) often differ by only one or two nucleotides. Therefore, instead of representing an amino acid like alanine as a generic token, such as A-UNK (an alanine residue with an unspecified codon), as in CodonTransformer, our method enables partial decoding (e.g., GC<UNK>), preserving richer sequence-level information (Figure 1a). During training, a fraction of nucleotide tokens are randomly masked with a special <mask> symbol. During inference, nucleotides that can be confidently determined are explicitly predicted, while only the uncertain ones are represented with <UNK>. This strategy enables the model to generate optimized DNA sequences conditioned on the target protein sequence with improved precision and biological fidelity. The vocabulary granularity ablation analysis is provided in Supporting Table S1 of the supplementary materials.

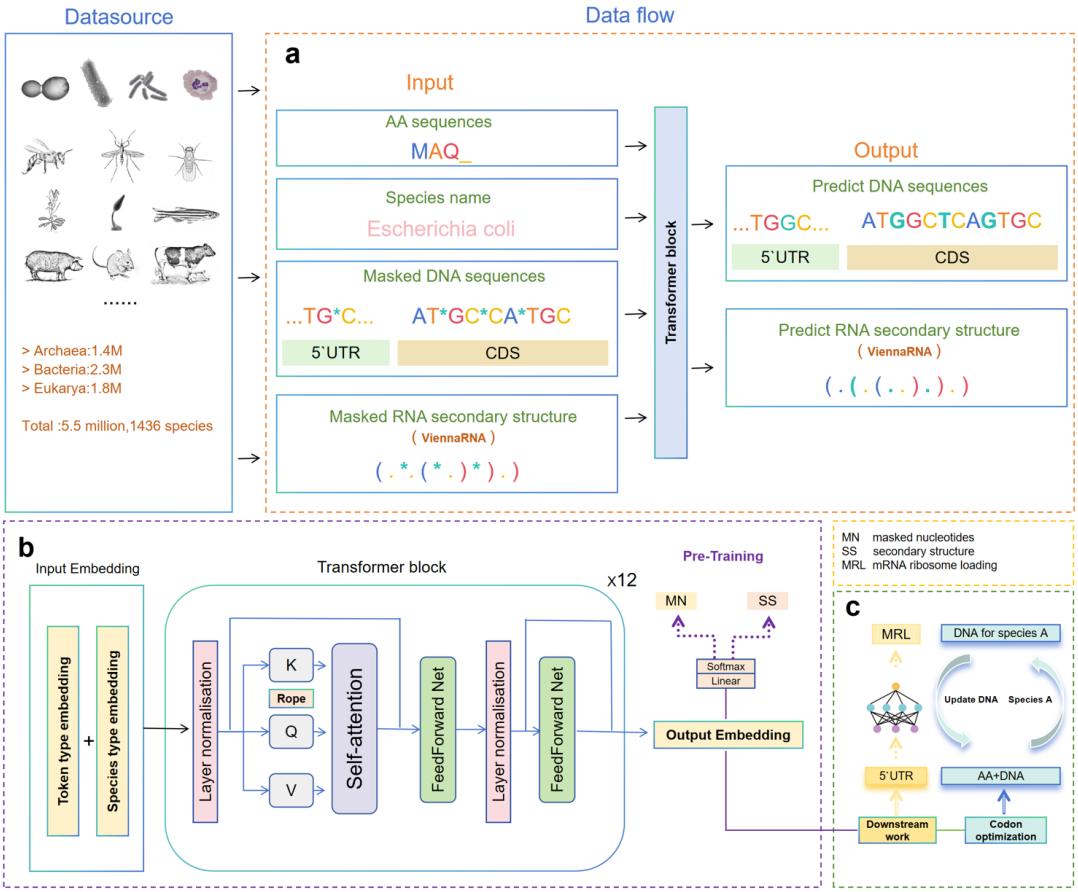

Figure 1: Overview of the design and applications of the codon optimization model, TransCodon. The model was trained on approximately 5.5 million samples from 1,436 species. **a. Data flow**: The model takes as input an amino acid sequence, species name, a masked DNA sequence (including both the 5' untranslated region and the coding region), and a masked RNA secondary structure predicted by ViennaRNA. These inputs are processed through multiple Transformer layers to produce the recovered DNA sequence and RNA secondary structure. **b. Model architecture**: TransCodon adopts an encoder-only Transformer architecture. The input embeddings include token type embeddings and species identity embeddings. The training objectives comprise masked nucleotide prediction (MN) and secondary structure prediction (SS). **c. Applications**: TransCodon can be applied to downstream tasks such as 5'UTR-related MRL prediction. Additionally, it can be used for cross-species codon optimization: given a target species and an amino acid sequence, the model generates a random DNA sequence and iteratively refines it to produce an optimized DNA sequence specific to the target species.

Furthermore, to achieve species-aware optimization, we adopt a strategy inspired by the token type embedding mechanism used in Transformer architectures. Specifically, we incorporate a species embedding into the learnable token embeddings, mapping each species to a unique token type identifier. This design enables the model to explicitly capture species-specific codon usage preferences. During inference, it allows users to direct DNA optimization toward a specified target species by selecting the corresponding token type (Figure 1a). During training, RNA secondary structure information is represented in dot-bracket notation and used as a supervised training label (Figure 1a). This supervised task,

similar to masked token recovery, is integrated with unsupervised masked language modeling to enable semi-supervised training. The model architecture is detailed in Figure 1a.

Our pretrained model, named TransCodon, was trained on four datasets comprising approximately 5.5 million gene sequences from 1436 species, including bacteria (41.38%), archaea (25.88%), and eukaryotes (32.75%) (Figure 1). This diverse dataset ensures robust cross-species generalization. Additionally, we constructed a holdout test dataset composed of eight representative species from across the tree of life. This test dataset was used for downstream evaluation and fine-tuning with the top 10% of high-expression genes ranked by CSI[33]. Details regarding the model architecture, training protocols, fine-tuning, and dataset preprocessing can be found in the Methods section.

2.2 TransCodon generates DNA sequences closely resembling native sequences

To assess the similarity between DNA sequences generated by the model and their corresponding native sequences across different species, we evaluated the generated sequences using several metrics, including codon recovery rate, codon similarity index (CSI)[33], codon adaptation index (CAI)[9], GC content[34], codon frequency distribution (CFD)[35], minimum free energy (MFE)[36], and dynamic time warping (DTW)[37]. Definitions of these metrics are provided in the Methods section. For a fair comparison, the test set was selected from a reserved subset of eight representative species within Dataset1 (the training set used for CodonTransformer, as described in the Methods section). These species were also included in the fine-tuning dataset for CodonTransformer. Importantly, the training and test sets were duplicated using CD-HIT[38] with a 40% sequence identity threshold, while CodonTransformer's pretraining utilized the full dataset without duplicates.

2.2.1 Codon recovery rate

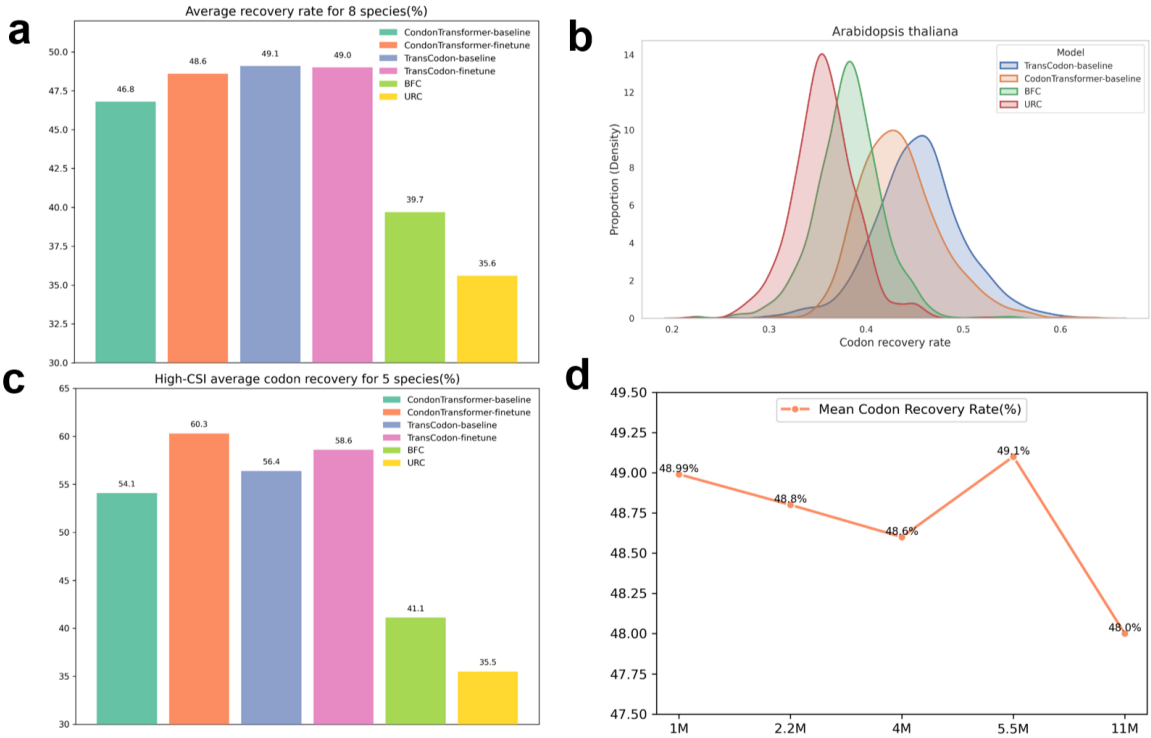

Figure 2: **a.** The average codon recovery rate across 8 species, with species details provided in the Methods section. **b.** Distribution of codon recovery rates for *Arabidopsis thaliana* as a representative species. **c.** Codon recovery rate for high-CSI sequences across 5 species (*Arabidopsis thaliana*, *Escherichia coli*, *Homo sapiens*, *Mus musculus*, *and Saccharomyces cerevisiae*). **d.** Impact of varying training dataset sizes on codon recovery rate, with the test set consisting of the 8 retained species from Dataset1. The training sets are as follows: 1M (Dataset1), 2.2M (Dataset1+Dataset2), 4M (Dataset1+Dataset2+Dataset3), 5.5M (Dataset1+Dataset2+Dataset4), and 11M (Dataset1+Dataset2) with an additional 8.8M bacterial data.

Codon recovery rate is the most direct metric for assessing the similarity between generated and native sequences. As shown in Figure 2a, TransCodon outperforms uniform random choice (URC), background frequency choice (BFC), and

CodonTransformer in this regard. The TransCodon-baseline model achieves an average codon recovery rate of 49.0% on the held-out test set, demonstrating its ability to preserve the inherent structure and functional features of the target sequences. Moreover, fine-tuning did not yield significant improvements over the base model, suggesting that the pretraining phase had already captured the codon usage patterns effectively. In contrast, after fine-tuning, CodonTransformer exhibited a slight improvement, reaching a codon recovery rate of 48.6%, which is comparable to that of the TransCodon-baseline model.

Following the experiments of CondonTransformer,we also analyzed the codon recovery rate of high-CSI sequences in five species. Fifty sequences per species (the top 10% in terms of CSI and included in the fine-tuning dataset) were selected for testing. As shown in Figure 2c, the TransCodon-baseline still outperforms the CodonTransformer-baseline, while CodonTransformer showed the best performance after fine-tuning, slightly surpassing TransCodon.

Although the test data originates from Dataset1, we assessed the model's performance by progressively expanding the training set. As shown in Figure 2d, the average codon recovery rate remained stable across eight species as the training data increased from 1M to approximately 5.5M, indicating robust performance and effective assimilation of codon distribution information from diverse species. However, when the training data expanded to 11M, the codon recovery rate decreased, suggesting that 11M may exceed the effective modeling capacity of the current model parameters.

2.2.2 CSI and CFD

In previous methods, when evaluating CSI/CAI and CFD metrics, it was generally assumed that higher CSI/CAI values and lower CFD values indicate better expression levels[9, 39]. While this assumption may reflect a model's ability to capture codon usage frequencies, it also risks replicating the shortcomings of approaches that merely select the most frequent codons—potentially leading to issues such as improper protein folding. We argue that the optimal outcome is for the generated sequences to closely match the CSI/CAI and CFD distributions of natural sequences, particularly those of naturally highly expressed sequences.

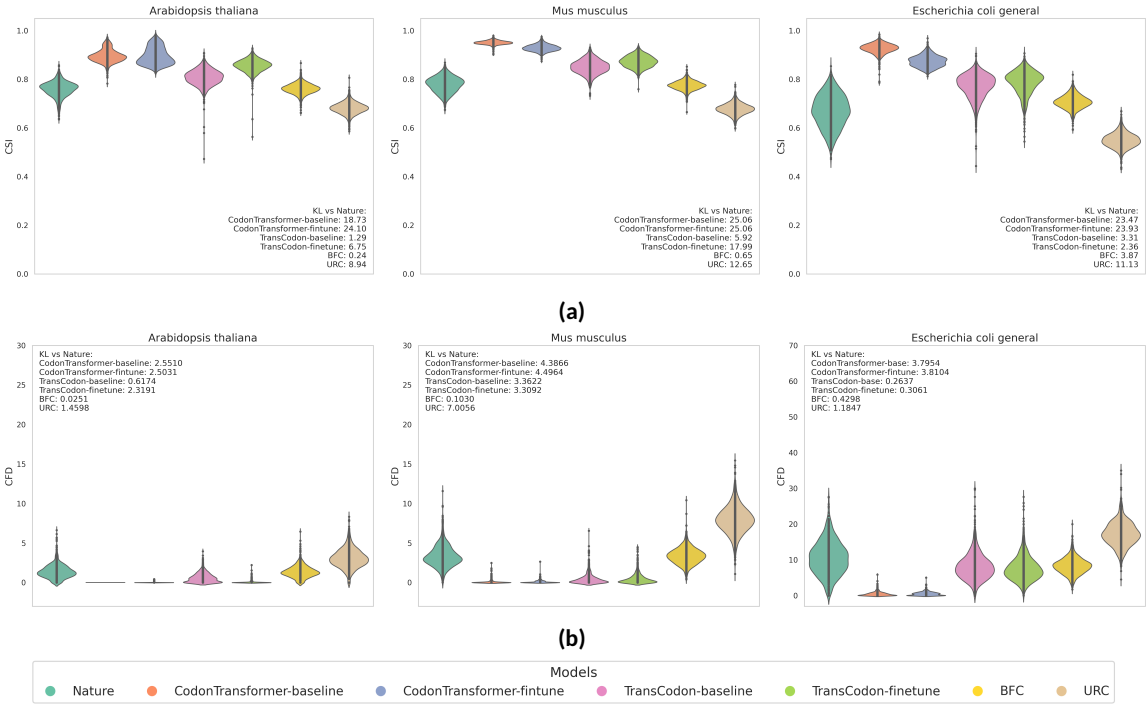

Figure 3: TransCodon-generated sequences closely resemble natural sequences. Using *Arabidopsis thaliana*, *Mus musculus*, and *Escherichia coli* as representative species (full details for all 8 species are provided in the supporting materials). **(a).** In terms of CSI, both TransCodon-baseline and BFC exhibit the closest similarity to natural sequences. For *E. coli*, the KL divergence between TransCodon-baseline and natural sequences is 3.31, significantly lower than CodonTransformer's 23.47. **(b).** Similarly, in CFD analysis, TransCodon-baseline and BFC most closely mimic natural distributions. Notably, fine-tuning with high-CSI genes typically increases TransCodon's KL divergence— for example, from 0.6174 to 2.3191 in *A. thaliana*.

Using *Arabidopsis thaliana*, *Mus musculus*, and *Escherichia coli general* as representative species (results for other species are provided in the supplementary materials), we compared several models based on the CSI and CFD. The Kullback-Leibler (KL)[40] divergence is a key measure of how one probability distribution diverges from a second, reference distribution—the smaller the value, the closer the distributions are. As shown in Figure 3, the BFC method, which is based purely on global codon frequencies, yields average CSI and CFD values that are closer to the natural distribution. For instance, in *A. thaliana*, its KL divergences are 0.24 and 0.0251 for CSI and CFD, respectively. However, this method lacks deep modeling capabilities, relying purely on simple probability heuristics. In contrast, TransCodon demonstrated superior alignment with natural sequences across both metrics, achieving KL divergences of 1.29 (CSI) and 0.6174 (CFD) in *A. thaliana*. By contrast, CodonTransformer exhibited systematic biases: its CSI values were disproportionately high, while its CFD scores deviated markedly from biological reality. The baseline model showed extreme divergence from natural data—with KL scores of 18.73 (CSI) and 2.551 (CFD) in *A. thaliana*.Notably, fine-tuning TransCodon with high-CSI sequences elevates the CSI of predicted sequences. As expected, this also increases the KL divergences between predicted and natural distributions for both CSI and CFD metrics.

2.2.3 GC and MFE

GC content varies significantly across species and even among different genes within the same species[41, 42]. These variations reflect the inherent diversity of genetic sequences. Therefore, when generating sequences, it is crucial to account for this diversity rather than enforcing a fixed GC ratio. Based on this, we contend that the GC content of generated sequences should closely match the GC levels of the corresponding natural sequences, rather than adhering to a fixed range.

Accordingly, in evaluating the structural properties of generated sequences, we prioritize the consistency of minimum free energy (MFE) values with those of natural sequences, rather than merely favoring lower or higher MFE values.

Using *Drosophila melanogaster*, *Mus musculus*, and *Escherichia coli general* as a representative example (results for other species are provided in the supplementary materials), we compare the GC content and MFE deviation between generated and natural sequences. A smaller deviation indicates greater similarity to the natural sequence. As shown in Figure 4a,b, TransCodon demonstrates superior performance in both metrics. For instance, in *Drosophila melanogaster*, the mean differences in GC content and MFE were 2.66 and 32.42, respectively, indicating that the generated sequences closely resemble natural sequences in both GC ratio and the mRNA structural stability. In contrast, CodonTransformer exhibited the largest discrepancies in these metrics (6.71 and 65.88), suggesting a suboptimal alignment with natural patterns. Notably, TransCodon's results were remarkably close to those of BFC (a frequency-based method), further validating that TransCodon has learned the codon frequency and natural probability distributions of the entire species, rather than merely memorizing the most frequent codons.

2.2.4 %MinMax and DTW

The %MinMax[43] metric captures local fluctuations in codon usage, offering detailed insights into patterns that influence translation efficiency and mRNA stability. However, existing methods often neglect these localized dynamics. Figure 4c illustrates two %MinMax profiles, with their similarity assessed using the Dynamic Time Warping (DTW) algorithm, which calculates the minimum alignment distance. Ideally, the %MinMax profile of generated sequences should closely match that of the corresponding natural sequences, preserving both global codon usage trends and local sequence features.

As shown in Figure 4d, comparative analysis of different models reveals that the %MinMax curves of sequences generated by TransCodon exhibit smaller DTW distances compared to natural sequences. The average DTW distance of TransCodon across 8 species is 0.158, significantly lower than the 0.341 and 0.411 values observed for CodonTransformer. Notably, in the original CodonTransformer paper, the DTW calculation was divided by 2. Additionally, its high DTW value is closely tied to the very high mean of the generated MinMax sequences, which is also associated with the excessively high CAI of its sequences. TransCodon, on the other hand, shows a smaller DTW distance of 0.146 when compared to the BFC method. As expected, fine-tuning TransCodon results in a slight increase in DTW distance, likely due to the use of high-CSI genes for fine-tuning. Although the BFC method demonstrates the smallest DTW distance to natural sequences, it relies solely on probabilistic random selection. In contrast, TransCodon-generated sequences align more closely with natural distributions and exhibit higher codon recovery rates.

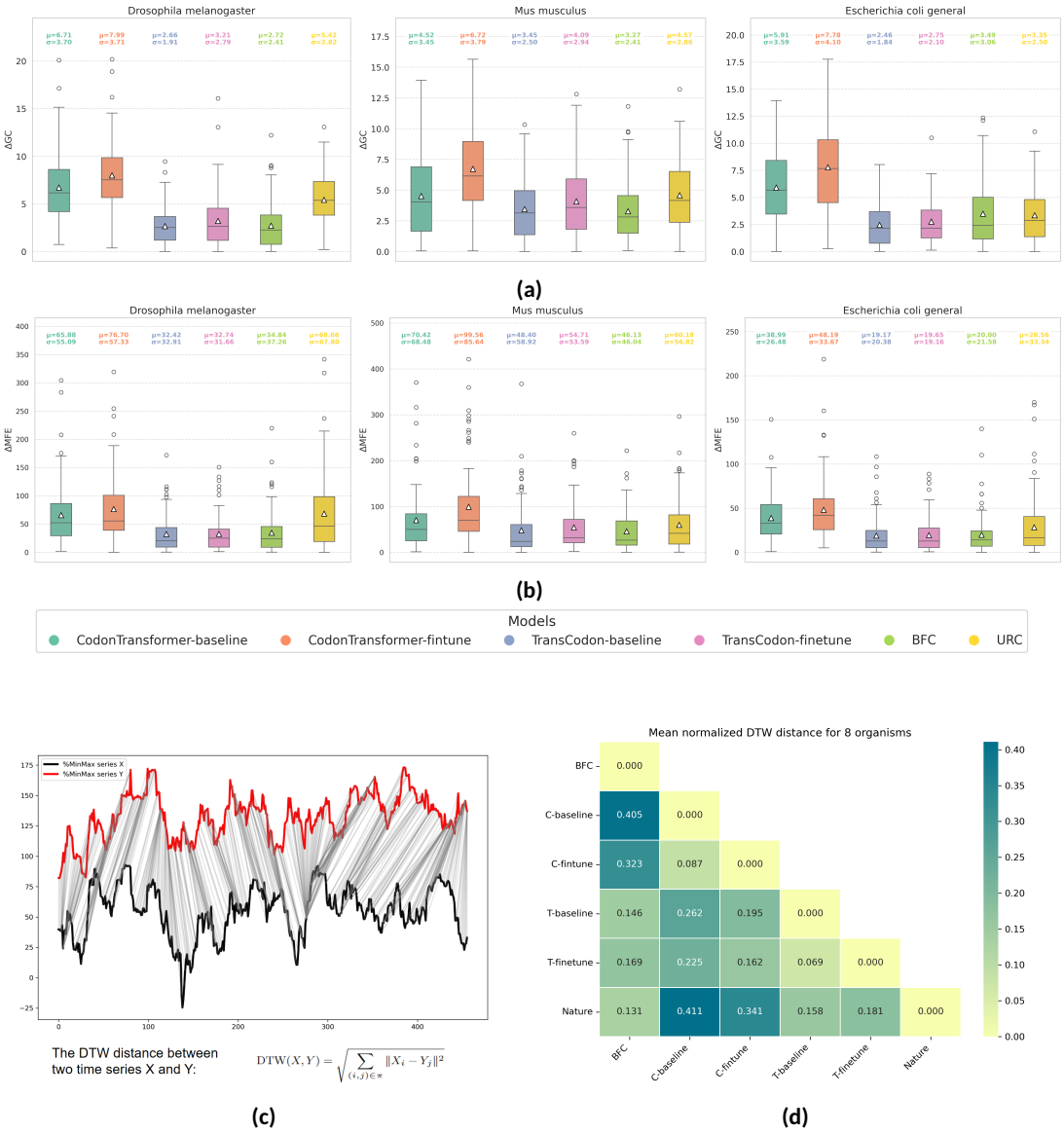

Figure 4: TransCodon-generated sequences closely resemble natural sequences. Using *Drosophila melanogaster*, *Mus musculus*, and *Escherichia coli* as representative species: **(a).** In terms of GC content, both TransCodon-baseline and BFC exhibited the closest alignment with natural sequences. For example, in *E. coli*, the average GC content difference between TransCodon-baseline and natural sequences was 2.46, significantly lower than the 5.91 observed in CodonTransformer. **(b).** Similarly, regarding Minimum Free Energy (MFE), both TransCodon-baseline and BFC closely approximated the MFE values of natural sequences. Notably, after fine-tuning with high-CSI genes, the MFE difference for CodonTransformer increased significantly, whereas TransCodon showed minimal changes. **(c).** Schematic diagram of %MinMax for the gene *yahG (E. coli)* between natural and TransCodon-generated sequences. The DTW algorithm calculates the minimum distance between two %MinMax sequences by finding matching positions (see "Methods" for details). **(d).** The average DTW values across 8 retained test species. C-baseline and T-baseline denote CodonTransformer-baseline and TransCodon-baseline, respectively. Results show that sequences generated by the BFC method and TransCodon have the smallest DTW distances to natural sequences, with values of 0.131 and 0.158, respectively.

In summary, the results across multiple evaluation metrics demonstrate that TransCodon effectively captures the natural codon distribution of different species and generates sequences that closely resemble natural DNA.

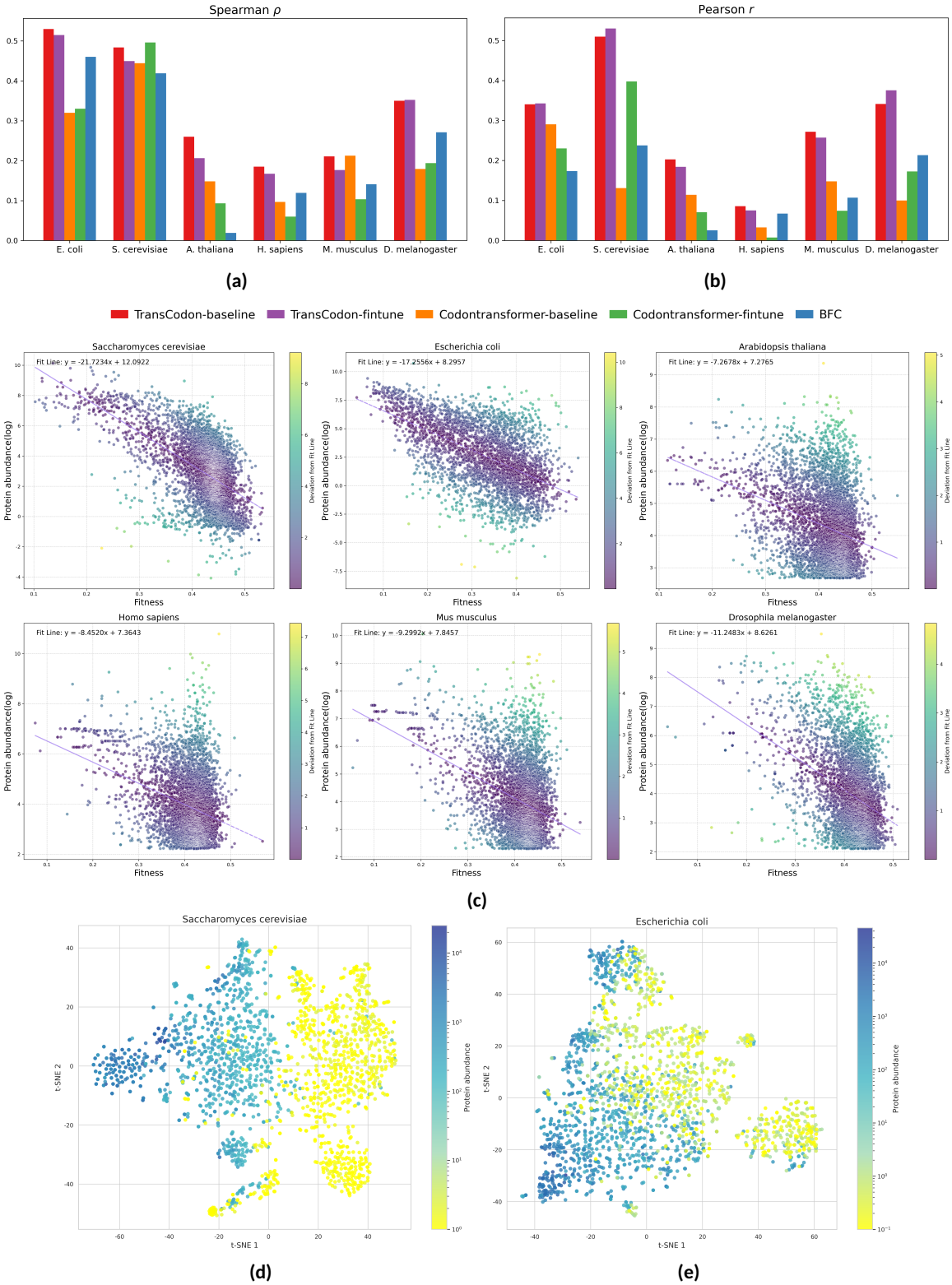

Figure 5: The correlation between model-predicted fitness score and protein abundance of different genes from *Escherichia coli*, *Saccharomyces cerevisiae*, *Arabidopsis thaliana*, *Homo sapiens*, *Mus musculus*, and *Drosophila melanogaster*. **(a,b).** The correlation between protein abundance and model fitness across six species using various methods, including TransCodon, CodonTransformer, URC, and BFC. **(c).** The relationship between protein abundance and model fitness across six species. The y-axis represents the log-transformed protein abundance, and the x-axis denotes the model's fitness score. The color bar indicates the deviation of different points from the fit line: the greater the deviation, the lighter the color. Notably, *Escherichia coli* and *Saccharomyces cerevisiae* exhibit a distinct trend, where lower fitness scores are associated with higher protein abundance. **(d,e).** Dimensionality reduction of embeddings for the top 1,000 and bottom 1,000 protein abundance sequences for *Escherichia coli* and *Saccharomyces cerevisiae* using TransCodon.

2.3 TransCodon exhibits robust effectiveness for protein abundance correlation in zero-shot scenarios

Beyond its ability to design sequences for heterologous expression systems, TransCodon also shows a certain level of correlation with experimentally determined protein abundance[44]. Previous studies have demonstrated that mRNA levels are strongly correlated with protein abundance, whereas the Codon Adaptation Index (CAI) shows a weaker correlation with protein abundance[39, 45]. In this context, the fitness score of a sequence evaluated by TransCodon reflects the likelihood of observing the corresponding DNA (or mRNA) sequence given the protein sequence. Therefore, it can be used as an index for predicting protein abundance.

We retrieved protein expression data for various genes from species including *Escherichia coli*, *Saccharomyces cerevisiae*, *Arabidopsis thaliana*,*Mus musculus*,*Drosophila melanogaster*, and *Homo sapiens* using the PaxDb database[44]. By using the native DNA sequences as input, we evaluated the correlation between protein abundance and the fitness score computed by different models. Details of the fitness score calculation can be found in the Methods section. As shown in Figure 5a,b, TransCodon consistently achieves the highest Spearman and Pearson correlation coefficients across all test species in the zero-shot setting, indicating that the model has effectively captured deep and biologically meaningful patterns.

The high correlation observed in prokaryotes such as *E. coli* can be attributed to the relatively simple regulatory mechanisms[46], where cis- and trans-acting elements exert certain influence on gene expression. As a result, codon usage remains a more dominant factor, making expression levels more predictable based solely on sequence features.

In contrast, gene expression in complex eukaryotic organisms is influenced by a wide range of regulatory mechanisms[47], including enhancers, silencers, chromatin accessibility, and post-transcriptional regulation. These layers of regulation introduce considerable biological variability, which can diminish the direct correlation between codon-level features and protein abundance — particularly in Pearson correlation, which is sensitive to linear relationships. Notably, our model continues to perform well on *Saccharomyces cerevisiae*, a relatively simple unicellular eukaryote. This result further underscores TransCodon's robustness in capturing codon-related fitness across diverse biological contexts.

Furthermore, we examined the relationship between protein abundance and model fitness scores across different sequences within each species. The y-axis represents the log-transformed protein abundance, as determined by mass spectrometry, while the x-axis corresponds to the model's fitness score. As illustrated in Figure 5c, all six species exhibit a consistent trend, where lower fitness scores correlate with higher protein abundance. This pattern is especially pronounced in *Escherichia coli* and *Saccharomyces cerevisiae*, and aligns with the computed Spearman correlation coefficients.

Specifically, we applied t-SNE to reduce the dimensionality of the embeddings for the top 1,000 and bottom 1,000 protein abundance sequences. As shown in Figure 5d,e, the embeddings corresponding to high and low protein abundance show clear separation for both *Saccharomyces cerevisiae* and *Escherichia coli* (see Supplementary Materials for results from other species). This observation further supports the model's ability to capture meaningful biological information.

2.4 5' UTR-Based downstream prediction of mean ribosome load (MRL)

Ribosome load refers to the number of ribosomes actively engaged in translating a given mRNA molecule at a specific time [48]. The Mean Ribosome Load (MRL) is the average number of ribosomes translating a particular mRNA and serves as a critical indicator of translational efficiency. MRL is influenced by various factors, including the sequence of the 5' untranslated region (5' UTR) and the secondary structure of the mRNA. UTR-LM[15], a BERT-based pre-trained model specifically trained on 5' UTR sequences, has demonstrated superior performance in predicting mean ribosome load (MRL) compared to other methods, such as Optimus[49], FramePool[50], MTTrans[51], RNABERT[52], and RNA-FM[31]. Our model, TransCodon, also integrates 5' UTR data during training on Dataset2 and Dataset3, prompting us to evaluate its effectiveness for 5' UTR-related downstream tasks.

We utilized the $\mathtt{U1}$, $\mathtt{U2}$, $\psi 1$, and $\psi 2$ libraries provided by UTR-LM—synthetic sub-libraries containing 50-nucleotide $5'$ UTR sequences—for Mean Ribosome Load (MRL) prediction. The data were split using a rank-based strategy[15], with each library comprising approximately 260,000 sequences for training and around 20,000 sequences for testing. As shown in Figure 1c, the final output embedding is obtained after passing through multiple Transformer layers. An MRL prediction head, implemented as a simple multilayer perceptron (MLP) with one hidden layer, is added on top of the TransCodon-baseline model. Following the UTR-LM approach[15], we perform full-parameter fine-tuning for MRL prediction.

Experimental results shown in Figure 6 demonstrate that, using the same simple MLP prediction head, TransCodon outperforms UTR-LM[15] across Spearman and Pearson correlation metrics, as well as RMSE and MAE. Furthermore, it significantly surpasses RNA-BERT[52] and RNA-FM[31] in performance. These results indicate that TransCodon excels not

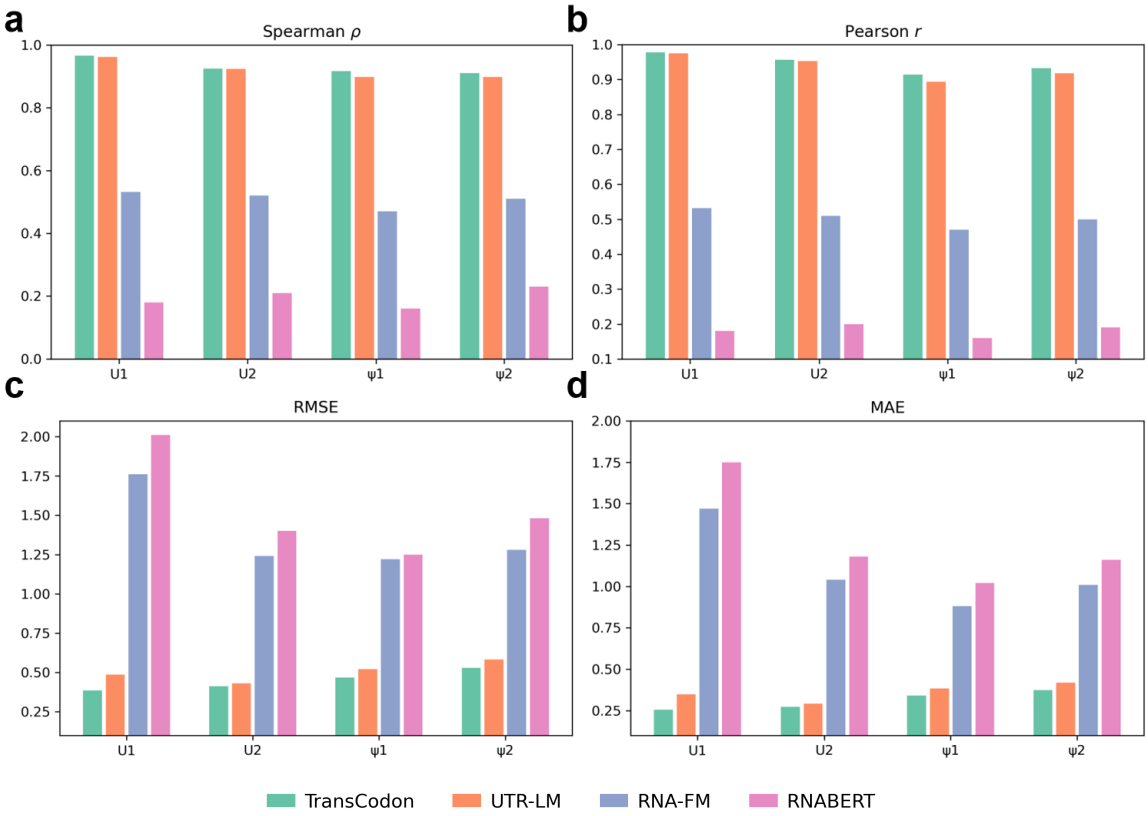

Figure 6: **a, b, c, d.** Performance comparison of MRL prediction across different methods, evaluated using Spearman and Pearson correlation coefficients, RMSE, and MAE. The results for *RNABERT*, *RNA-FM*, and *UTR-LM* are taken directly from the original UTR-LM publication.

only in codon optimization but also in downstream tasks related to the 5' UTR.

3 Discusstion

3.1 Ablation study

To gain deeper insights into the factors influencing model performance,we conducted a series of ablation studies to evaluate the effects of training data volume, RNA secondary structure, and the role of the 5'UTR region. The results also warrant further investigation.

In the development of large models, both training data size and model parameters are critical for enhancing performance. Since TransCodon incorporates species labels, the DNA data for each species is limited, and expanding the training set introduces new codon distributions from additional species. This raises the question of how these new species affect the codon landscape of existing species. With a fixed test set, increasing the training data volume results in minimal changes to performance metrics, yet the correlation between model fitness and protein abundance is significantly influenced by the type of training data. From Figure S1, we observe that for eukaryotes such as *D. melanogaster*, *H. sapiens*, and *M. musculus*, the correlation decreased as training data increased from 1M to 2.2M. This is likely due to the inclusion of archaea (Dataset2), whose codon usage deviates from that of eukaryotes. However, when the data size reached 5.5M (including fungal data), the correlation peaked. For microorganisms like *E. coli* and *S. cerevisiae*, adding archaea and bacterial datasets improved correlation, but a slight decrease was observed with the inclusion of fungal data. This suggests that model performance is influenced by both the scale and the composition of the training data. Expanding the training set from 1M to 5.5M enhances the model's ability to capture codon usage preferences across species, with species diversity playing a crucial role in model generalization. Whether increasing model parameters to accommodate larger, more complex datasets will further enhance performance remains an area for future exploration.

Additionally, incorporating 5' UTR information during pretraining proves to be effective; however, omitting it during infer-

ence negatively impacts performance. While the inclusion of RNA secondary structure information did not significantly affect codon recovery, it revealed a notable pattern in the correlation between fitness and protein abundance, suggesting its potential for deeper biological insights. Detailed results of all ablation experiments are provided in the supplementary materials.

3.2 Learning the low-frequency codons in a highly expressed protein

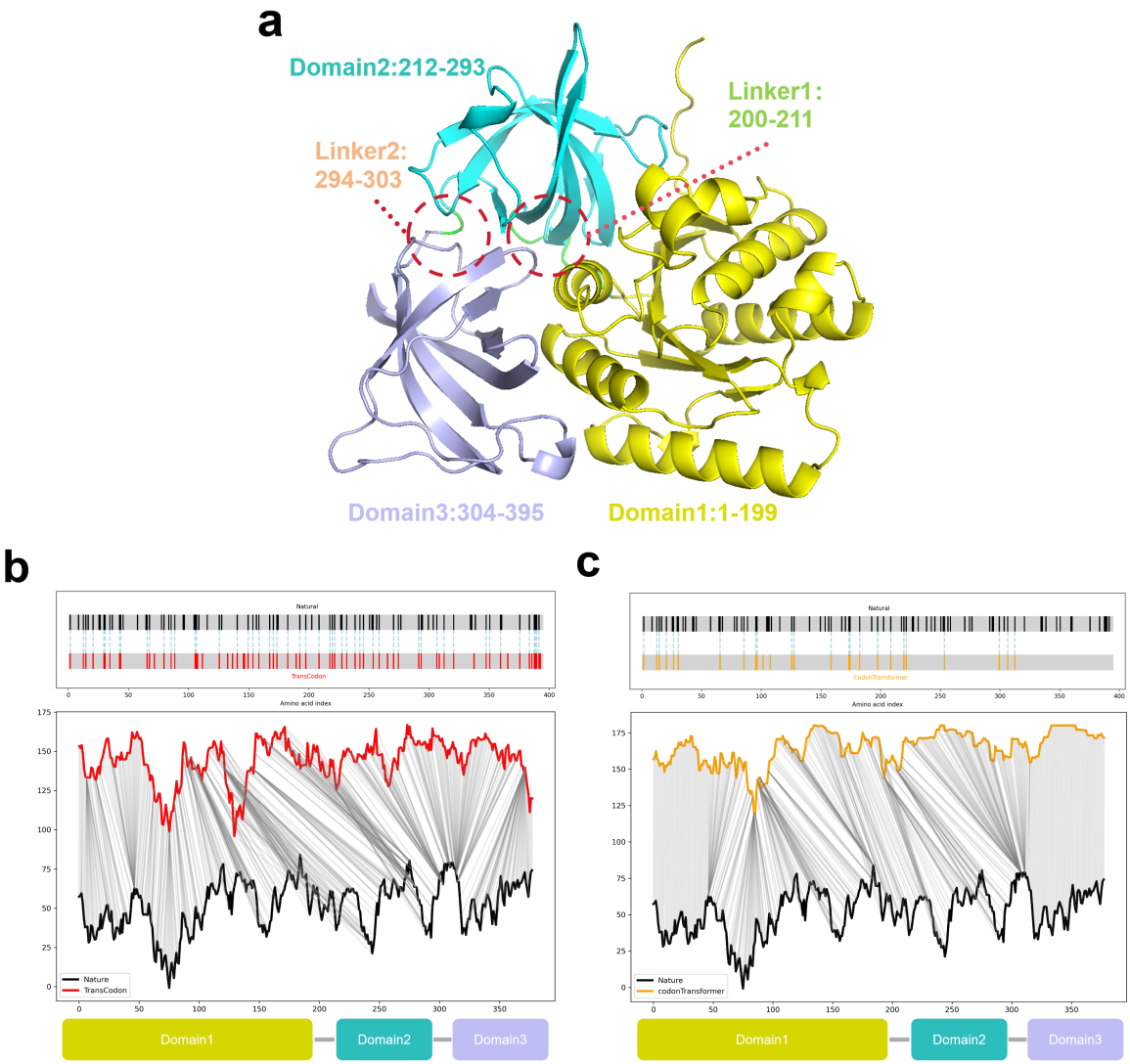

Figure 7: TransCodon captures the distribution of rare codons in inter-domain regions. **a.** The 3D structure of the *tufA* protein, which consists of three structural domains and two linker regions. The numbers indicate the amino acid sequence positions. **b,c.** In these panels, black represents the natural sequence, red represents the TransCodon-generated sequence, and orange represents the CodonTransformer-generated sequence. The top panel displays the distribution of low-frequency codons, with blue dashed lines marking the matches. The middle panel shows the %MinMax profiles, and the bottom panel provides a summary of the structural domains.

Many existing codon optimization tools primarily focus on maximizing the Codon Adaptation Index (CAI)[20, 21], which may inadvertently lead to issues such as protein misfolding. For multi-domain proteins, some studies[14, 53, 54] suggest that regions between the N-terminal and downstream domains are often enriched in low-frequency codons, potentially slowing translation to facilitate proper folding. However, this is not a universal principle.

To explore this phenomenon, we examined the first highly expressed multi-domain protein (*tufA*, Elongation Factor Tu) in Escherichia coli [44]. *tufA* is involved in the GTP-dependent binding of aminoacyl-tRNA to the A-site of ribosomes and plays a crucial role in bacterial growth as well as in the cell's response to nutrient deprivation [55]. As shown in Figure 7a,

this gene encodes a protein with three domains and two inter-domain linkers. Defining low-frequency codons as those with a usage frequency below 0.3, we identified 85 low-frequency codons out of 395 total codons in the *tufA* coding sequence, yielding a proportion of 0.2152. In the two inter-domain regions, 5 out of 22 codons were low-frequency, corresponding to a proportion of 0.227, which is comparable to the overall frequency.

What we aim to investigate is whether TransCodon, given only the amino acid sequence as input, can effectively learn which regions should incorporate low-frequency codons to slow down translation and promote stable protein folding. The TransCodon-generated DNA sequence achieves a codon-level recovery rate of 72.91% compared to the natural CDS *tufA* sequence. As shown in Figure 7b, the top panel illustrates the distribution of low-frequency codons in both the natural and TransCodon-generated sequences, the middle panel presents the %MinMax profiles of the *tufA* gene, and the bottom panel summarizes the composition of each structural domain. Notably, TransCodon achieves a 63% match rate with the natural sequence in terms of low-frequency codon positions, significantly outperforming CodonTransformer, which only achieves 28%. When examining the %MinMax sequence plot in the mid-range (indices 0–100), TransCodon demonstrates strong alignment with the natural sequence's %MinMax pattern. Between indices 100–300, although direct matches are absent, TransCodon shows a slight lag compared to the natural sequence, yet retains a close morphological resemblance. In contrast, CodonTransformer's alignment becomes progressively harder to discern after index 100, which is consistent with its lower matching rate for low-frequency codons. This suggests that TransCodon effectively identifies regions where low-frequency codons should be incorporated, implying that it has implicitly learned structural and folding-related biological insights.

4 Conclusion

TransCodon establishes a transformative paradigm for codon optimization through its species-informed transformer architecture that integrates 5'UTR regulatory contexts, coding sequences, and RNA structural features. By leveraging multi-organism genomic patterns across many species, the model achieves unprecedented biological fidelity—generating sequences with 49.0% codon recovery while maintaining native-like distributions of codon adaptation (KL divergence = 1.29 for CSI) and low-frequency codon positioning (63% retention in structural domain interfaces). Crucially, TransCodon resolves the historical trade-off between expression optimization and sequence divergence: it simultaneously enhances protein abundance correlation (Spearman ρ = 0.82 in E. coli) and preserves translational dynamics critical for proper folding. The framework's versatility is further demonstrated through state-of-the-art performance in 5'UTR-mediated ribosome load prediction, showcasing its capacity to model end-to-end translational regulation. With openly available model weights and optimization pipelines, TransCodon provides a foundational tool for synthetic biology that bridges species-specific codon landscapes while enabling future extensions into chromatin-aware eukaryotic optimization and de novo protein design workflows.

5 Methods

5.1 Datasets

The pretraining data are compiled from four distinct datasets. Dataset 1, sourced from CodonTransformer, consists of approximately 1 million protein-coding sequences from 164 species, with bacterial, archaeal, and eukaryotic sequences representing 56.1%, 2.5%, and 41.4%, respectively. Datasets 2–4 include sequences retrieved from the NCBI database[56], encompassing approximately 1.4 million archaeal sequences (Dataset 2), 1.8 million bacterial sequences (Dataset 3), and 1.2 million eukaryotic sequences (Dataset 4). Importantly, Datasets 2 and 3 also contain 5' untranslated regions (5' UTRs), defined as the 100 nucleotides upstream of the start codon.

To rigorously assess both the model's expressiveness and generalization ability, and to ensure a fair comparison with CodonTransformer, we constructed a holdout test set comprising 8 representative species from Dataset 1: *Escherichia coli O157:H7 str. Sakai*, *Escherichia coli str. K-12 substr. MG1655*, *Homo sapiens*, *Arabidopsis thaliana*, *Saccharomyces cerevisiae*, *Escherichia coli (general)*, *Drosophila melanogaster*, and *Mus musculus*. All eight species are among the 15 species used for fine-tuning in CodonTransformer. To construct the test set, we applied CD-HIT[38] clustering at a 40% amino acid sequence similarity threshold and randomly selected 15% of the clusters from each species (up to a maximum of 1,000 sequences). Following filtering and preprocessing, the final held-out dataset consists of approximately 6,400 sequences.

The fine-tuning dataset was derived from the training data of the aforementioned 8 species, specifically selecting the top 10% of genes with the highest Codon Similarity Index (CSI) scores within each species. All preprocessing steps ensured the removal of sequence similarity between the training and test datasets. Consequently, the final fine-tuning dataset comprises approximately 18,000 sequences.

Table 1: Dataset Summary

Dataset	Species	Volume	With 5′UTR	Source
Dataset1[1]	164	1,001,197	No	CodonTransformer
Dataset2	538	1,407,346	100bp	Archaea
Dataset3	584	1,728,836	100bp	Bacteria
Dataset4	154	1,398,475	No	Eukarya

[1] Dataset1 contains Bacteria (561,625), Eukarya (414,445), and Archaea (25,127).

5.2 Model details

5.2.1 Model architecture

The model architecture is inspired by the ESM family of language models and consists of three primary components: a learnable embedding layer, a stack of Transformer encoder layers, and a language modeling head. The vocabulary comprises 10 tokens, including the four nucleotides (A, T, C, G), along with special tokens to represent sequence start (`<cls>`), end (`<eos>`), padding (`<pad>`), unknown nucleotides (`<unk>`), masking (`<mask>`), and a separator (`<sep>`) to distinguish between the 5'UTR and coding regions. Input token sequences of length T are first embedded into a 768-dimensional latent space, resulting in a $[T, 768]$ representation. This embedding is then processed through a Transformer encoder stack. Each self-attention layer employs multi-head scaled dot-product attention, where queries (Q), keys (K), and values (V) are projected from the inputs via learnable linear transformations. Our model uses 12 attention heads with a hidden size of 64 per head ($d_k = 64$), ensuring that each head captures distinct sequence features. To facilitate sequential modeling, rotary positional embeddings are applied instead of absolute position embeddings. Each Transformer block also includes residual connections, a feedforward neural network with a hidden dimension of 3072, and layer normalization. Furthermore, we employ pre-layer normalization and disable dropout to enhance training stability. After passing through 12 Transformer layers, the model outputs token representations of shape $[T, 768]$. These representations are then fed into the language modeling head, which consists of a feedforward network, a layer normalization step, and a matrix multiplication with the transposed embedding matrix. The head projects the 768-dimensional features to the vocabulary size, producing logits that, after applying the softmax function, yield uncalibrated probability distributions over the tokens. The model is trained using a masked language modeling (MLM) objective. For each sequence x, a subset of positions M is randomly sampled for masking, and the model is trained to minimize the negative log-likelihood of the true tokens at masked positions, given the masked input sequence.

5.2.2 Model input

The model takes both DNA sequences and their corresponding species IDs as input. To enable species-aware optimization, we incorporate a token type embedding mechanism inspired by the Transformer architecture, where each token embedding is combined with a learnable species embedding.

5.3 Training Objectives and Model Training

The training process of our model consists of two main stages: **pretraining** and **fine-tuning**, both utilizing a dynamic masking strategy. In each training batch, 25% of input tokens are randomly selected for masking: 80% of these are replaced with a special `<mask>` token, 10% are substituted with a random nucleotide, and the remaining 10% remain unchanged. The model supports sequences of up to 2048 tokens. For sequences exceeding this limit, random cropping is applied within each batch, and all sequences are padded to match the length of the longest sequence in the batch.

Masked Nucleotide (MN) Prediction In the pretraining phase, we adopt a masked language modeling (MLM) strategy. Specifically, 25% of the nucleotide tokens are randomly masked, and the model is trained to predict the masked tokens based on the surrounding context. This approach facilitates the learning of contextual nucleotide representations.

Let $x = (x_1, x_2, \ldots, x_T)$ denote a nucleotide sequence of length T, and let $M \subset \{1, 2, \ldots, T\}$ be the set of randomly selected masked indices, with $|M| = 0.25T$. The training objective for this task, referred to as masked nucleotide (MN) prediction, is:

$$\mathcal{L}_{\text{MN}} = E_{x \sim \mathcal{D}} E_M \left[\sum_{i \in M} -\log p(x_i \mid x_{\setminus M}) \right] \tag{1}$$

Here, $x_{\setminus M}$ denotes the unmasked tokens, and $p(x_i \mid x_{\setminus M})$ is the predicted probability of the true nucleotide at position i.

Secondary Structure (SS) Prediction To integrate RNA structural features, we introduce a supervised learning task that leverages secondary structure (SS) annotations for the coding region, computed by ViennaRNA[36]. The SS is represented in dot-bracket notation: paired nucleotides are denoted by `'('` and `')'`, while unpaired nucleotides are denoted by `'.'`.

Using the same masking strategy as above, each nucleotide masked x_i is associated with a corresponding SS label s_i, and the model is trained to predict this label. The SS prediction loss is defined as:

$$\mathcal{L}_{\text{SS}} = E_{x\sim\mathcal{D}} E_M \left[\sum_{i\in M} -\log p(s_i \mid x_{\setminus M}) \right] \tag{2}$$

where $s_i \in \{(,), .\}$ represents the SS label at position i.

Total Objective The overall training loss combines the MN and SS objectives with a weighting coefficient λ:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{MN}} + \lambda \mathcal{L}_{\text{SS}} \tag{3}$$

where λ balances the contribution of the SS task, with a default value of 0.4.

Training Details Pretraining was performed on four NVIDIA V100 GPUs for 5 epochs, requiring approximately 10 days. The AdamW optimizer was used, with a batch size of 3 and a gradient accumulation step of 6. We reserved 0.3% of the training data for validation. The learning rate was linearly warmed up from 0 to 2e-4 over the first 10% of steps and then decayed to 0 following a cosine schedule.

Fine-tuning was performed using training data from 8 representative species, maintaining the same hyperparameters as pretraining. This phase utilized two NVIDIA V100 GPUs and ran for 10 epochs.

Model inference During inference, the nucleotides corresponding to the known amino acid sequence are explicitly provided, while unknown regions are marked with the `<unk>` token. For example, for the amino acid sequence "MKD", the input DNA sequence might be ATGAA`<unk>`GA`<unk>`. This approach, compared to using only the amino acid sequence, supplies richer contextual information, allowing the model to more effectively generate optimized DNA sequences. Alternatively, a complete and valid DNA sequence can be directly input for further optimization.

5.4 Model evaluation

To comprehensively evaluate the quality of the generated sequences, we utilized a variety of assessment metrics, primarily adapted from those employed in CodonTransformer[21]. Each metric focuses on a distinct aspect of codon usage or sequence characteristics. These evaluations are designed to assess the extent to which the generated DNA sequences align with natural sequences, considering both their biological relevance and structural properties.

Specifically, we evaluated:

- **Codon Recovery Rate**: Measures the percentage of exact codon matches between the generated and natural DNA sequences. A higher recovery rate indicates better preservation of native sequence structure and codon usage. For a natural DNA sequence A and a generated sequence B, the recovery rate is calculated as:

$$\text{Recovery rate}(A, B) = \frac{1}{L} \sum_{i=1}^{L} \delta(a_i, b_i) \times 100 \tag{4}$$

  where L is the length of the sequences, a_i and b_i are the codons at position i in sequences A and B, respectively, and δ is the Kronecker delta function, which equals 1 if $a_i = b_i$ and 0 otherwise.

- **Codon Similarity Index (CSI)**: The Codon Similarity Index (CSI)[33], derived from the Codon Adaptation Index (CAI)[9] , quantifies the similarity in codon usage between a sequence and the codon usage table of an organism, without relying on any highly expressed genes. CSI provides a softer and potentially more biologically appropriate estimator of codon usage for a given organism.

The relative adaptiveness of a codon w_{ij} is calculated as the ratio of its frequency x_{ij} to that of the most frequently used synonymous codon x_i^{max} for the same amino acid:

$$w_{ij} = \frac{x_{ij}}{x_i^{\text{max}}} \tag{5}$$

The CSI for a gene is then computed as the geometric mean of these relative adaptiveness values over all codons in a gene sequence of length L. A CSI value of 1 indicates that the sequence exclusively uses the most frequently used codons. Here, w_k denotes the value of w_{ij} corresponding to the current position k.

$$\text{CSI} = \exp\left(\frac{1}{L}\sum_{k=1}^{L} \ln w_k\right) \tag{6}$$

- **Codon Frequency Distribution (CFD)**: The Codon Frequency Distribution (CFD)[35] indicates the frequency of rare codons selected by the model (with a frequency < 0.3), which tends to vary across species. A weight w_{ij} is assigned to each codon as:

$$w_{ij} = \begin{cases} 1 & \text{if } \frac{x_{ij}}{x_i^{\text{max}}} < 0.3 \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

The CFD is then computed as the mean of these weights across all L codons in the sequence:

$$\text{CFD} = \frac{1}{L}\sum_{k=1}^{L} w_k \tag{8}$$

- **GC Content (%GC)**: GC content[34] is a fundamental metric for measuring the proportion of guanine (G) and cytosine (C) in a DNA sequence. Different species and genes typically exhibit specific GC content ranges, which are closely related to gene stability and expression levels. It measures the proportion of guanine (G) and cytosine (C) bases in a DNA sequence.

$$\text{GC} = \frac{G + C}{A + T + G + C} \tag{9}$$

- **%MinMax**: The %MinMax[43] evaluates the balance between high-frequency and low-frequency codons within a sliding window along the sequence. This metric captures local fluctuations in codon usage, offering insights into the fine-grained distribution patterns that can influence translational efficiency and mRNA stability. For a window size w (typically $w = 18$), the %MinMax value at window i is computed as:

$$\%\text{MinMax}_i = \frac{x_i^{\text{max}} - x_i^{\text{min}}}{x_i^{\text{max}} + x_i^{\text{min}}} \times 100 \tag{10}$$

where x_i^{max} and x_i^{min} are the maximum and minimum codon usage frequencies within the i-th window, respectively. The overall %MinMax for the gene is an array of such values:

$$\%\text{MinMax} = [\%\text{MinMax}_1, \%\text{MinMax}_2, \ldots, \%\text{MinMax}_n] \tag{11}$$

where n is the total number of windows in the sequence.

- **Dynamic Time Warping (DTW)**: Dynamic Time Warping (DTW)[37] is an algorithm designed to measure the similarity or alignment between two time series, particularly excelling in scenarios involving local distortions or shifts. In this study, we employ DTW to quantify the distance between the MinMax profiles of generated and native sequences, providing a more detailed evaluation of codon usage patterns across the entire sequence.The DTW distance between two time series X and Y is defined as:

$$\text{DTW}(X, Y) = \sqrt{\sum_{(i,j)\in\pi} \|X_i - Y_j\|^2} \tag{12}$$

where X and Y are the %MinMax vectors, π is the optimal alignment path, and $\|X_i - Y_j\|$ denotes the Euclidean distance between aligned elements. DTW provides a robust quantitative assessment of how closely a generated sequence's codon usage pattern follows the native profile. A smaller DTW distance indicates that the local dynamics of codon usage are more faithfully captured.

- **Minimum Free Energy (MFE) of RNA Structures**: Minimum Free Energy (MFE) reflects the stability of secondary structures (such as hairpin structures) formed by nucleic acid sequences at thermodynamic equilibrium. Lower energy generally indicates a more stable structure. Both overly stable and overly relaxed secondary structures can affect the transcription and translation efficiency of mRNA. RNA secondary structure prediction was performed using the $\text{fold}()$ function from the ViennaRNA package[36], which computes the minimum free energy (MFE) of the resulting structures.

The code for calculating the Codon Recovery Rate, CSI, CFD, GC Content, %MinMax, DTW, and MFE is included in the source code. By incorporating these diverse metrics, we establish a comprehensive and multidimensional evaluation framework. This ensures that the model is not solely optimized for a single criterion (e.g., high CAI) but is also capable of capturing the subtle variations and biologically relevant features inherent in natural DNA sequences.

5.5 Model fitness score

To assess how well a coding DNA sequence aligns with the model's learned codon preferences, we define a fitness score at the codon level. Given a codon sequence $c = (c_1, c_2, \ldots, c_N)$ of length N (where each c_i represents a triplet codon), the model assigns a conditional probability $p(c_i \mid c_{\setminus i})$ to each codon based on its surrounding context.

The fitness score $F(c)$ is computed as the average log-likelihood of the codons under the model:

$$F(c) = \frac{1}{N} \sum_{i=1}^{N} \log p(c_i \mid c_{\setminus i}) \tag{13}$$

6 Acknowledgment

This work was partly supported by the National Key Research and Development Program of China under Grant No. 2024YFA0919702 (Y.W.) and 2023YFA0915500 (S.W. and L.Z.), National Science Foundation of China under grant no. 62272449 (Y.W.) and 12426303 (Y.W.), the Shenzhen Basic Research Fund under grant no KQTD20200820113106007, ZDSYS20220422103800001. We would also like to thank the funding support by the Key Laboratory of Quantitative Synthetic Biology, Chinese Academy of Sciences under grant no. CKL075. This research is also supported by the Science and Technology Development Fund of Macau (0004/2025/RIA1 to J.G.).

7 Data availability

The dataset used in this study consists of pretraining data, fine-tuning data, and a held-out test set. All processed sequences and related annotation files are publicly available at https://drive.google.com/drive/folders. The pretrained model weights can be downloaded from https://huggingface.co/guyuehuo/TransCodon.

8 Code availability

The source code used to train and evaluate the *TransCodon* model is publicly available at https://github.com/guyuehuo/TransCodon. The repository contains scripts for data preprocessing, model training, inference, and evaluation, along with detailed instructions for reproducing the results presented in this paper.

References

[1] Sebastian Pechmann and Judith Frydman. Evolutionary conservation of codon optimality reveals hidden signatures of cotranslational folding. *Nature structural & molecular biology*, 20(2):237–243, 2013.

[2] Eduardo PC Rocha. Codon usage bias from trna's point of view: redundancy, specialization, and efficient decoding for translation optimization. *Genome research*, 14(11):2279–2286, 2004.

[3] John B Ingraham, Max Baranov, Zak Costello, Karl W Barber, Wujie Wang, Ahmed Ismail, Vincent Frappier, Dana M Lord, Christopher Ng-Thow-Hing, Erik R Van Vlack, et al. Illuminating protein space with a programmable generative model. *Nature*, 623(7989):1070–1078, 2023.

[4] Ali Madani, Ben Krause, Eric R Greene, Subu Subramanian, Benjamin P Mohr, James M Holton, Jose Luis Olmos Jr, Caiming Xiong, Zachary Z Sun, Richard Socher, et al. Large language models generate functional protein sequences across diverse families. *Nature biotechnology*, 41(8):1099–1106, 2023.

[5] F. Hu, Y. Peng, S. Chang, X. Luo, Y. Yuan, X. Zhu, Y. Xu, K. Du, Y. Chen, et al. Vimentin binds to a novel tumor suppressor protein, gspt1-238aa, encoded by circgspt1 with a selective encoding priority to halt autophagy in gastric carcinoma. *Cancer Letters*, 545:215826, 2024. doi: 10.1016/j.canlet.2024.215826.

[6] H. Zhang, Q. Huang, Z. Bei, Y. Wei, and C.A. Floudas. Comsat: Residue contact prediction of transmembrane proteins based on support vector machines and mixed integer linear programming. *Proteins: Structure, Function, and Bioinformatics*, 84(3):332–348, 2016. doi: 10.1002/prot.24988.

[7] J. Meng, P. Chen, M. Wahib, M. Yang, L. Zheng, Y. Wei, S. Feng, and W. Liu. Boosting the predictive performance with aqueous solubility dataset curation. *Scientific Data*, 9(1):71, 2022. doi: 10.1038/s41597-022-01137-z.

[8] Evelina Angov, Patricia M Legler, and Ryan M Mease. Adjustment of codon usage frequencies by codon harmonization improves protein expression and folding. *Heterologous Gene Expression in E. coli: Methods and Protocols*, pages 1–13, 2010.

[9] Paul M Sharp and Wen-Hsiung Li. The codon adaptation index-a measure of directional synonymous codon usage bias, and its potential applications. *Nucleic acids research*, 15(3):1281–1295, 1987.

[10] Claes Gustafsson, Jeremy Minshull, Sridhar Govindarajan, Jon Ness, Alan Villalobos, and Mark Welch. Engineering genes for predictable protein expression. *Protein expression and purification*, 83(1):37–46, 2012.

[11] Barbara Maertens, Anne Spriestersbach, Uritza von Groll, Udo Roth, Jan Kubicek, Michael Gerrits, Marcus Graf, Michael Liss, Daniela Daubert, Ralf Wagner, et al. Gene optimization mechanisms: a multi-gene study reveals a high success rate of full-length human proteins expressed in escherichia coli. *Protein Science*, 19(7):1312–1326, 2010.

[12] Tessa EF Quax, Nico J Claassens, Dieter Söll, and John van der Oost. Codon bias as a means to fine-tune gene expression. *Molecular cell*, 59(2):149–161, 2015.

[13] Nico J Claassens, Melvin F Siliakus, Sebastiaan K Spaans, Sjoerd CA Creutzburg, Bart Nijsse, Peter J Schaap, Tessa EF Quax, and John Van Der Oost. Improving heterologous membrane protein production in escherichia coli by combining transcriptional tuning and codon usage algorithms. *PLoS One*, 12(9):e0184355, 2017.

[14] Evelina Angov, Collette J Hillier, Randall L Kincaid, and Jeffrey A Lyon. Heterologous protein expression is enhanced by harmonizing the codon usage frequencies of the target gene with those of the expression host. *PloS one*, 3(5): e2189, 2008.

[15] Yanyi Chu, Dan Yu, Yupeng Li, Kaixuan Huang, Yue Shen, Le Cong, Jason Zhang, and Mengdi Wang. A 5 utr language model for decoding untranslated regions of mrna and function predictions. *Nature Machine Intelligence*, 6(4):449–460, 2024.

[16] Zhichao Miao, Antonin Tidu, Gilbert Eriani, and Franck Martin. Secondary structure of the sars-cov-2 5'-utr. *RNA biology*, 18(4):447–456, 2021.

[17] Patricia R Araujo, Kihoon Yoon, Daijin Ko, Andrew D Smith, Mei Qiao, Uthra Suresh, Suzanne C Burns, and Luiz OF Penalva. Before it gets started: regulating translation at the 5 utr. *International Journal of Genomics*, 2012(1):475731, 2012.

[18] Anton A Komar. A pause for thought along the co-translational folding pathway. *Trends in biochemical sciences*, 34 (1):16–24, 2009.

[19] Zilin Ren, Lili Jiang, Yaxin Di, Dufei Zhang, Jianli Gong, Jianting Gong, Qiwei Jiang, Zhiguo Fu, Pingping Sun, Bo Zhou, et al. Codonbert: a bert-based architecture tailored for codon optimization using the cross-attention mechanism. *Bioinformatics*, 40(7):btae330, 2024.

[20] Rishab Jain, Aditya Jain, Elizabeth Mauro, Kevin LeShane, and Douglas Densmore. Icor: improving codon optimization with recurrent neural networks. *BMC bioinformatics*, 24(1):132, 2023.

[21] Adibvafa Fallahpour, Vincent Gureghian, Guillaume J Filion, Ariel B Lindner, and Amir Pandi. Codontransformer: a multispecies codon optimizer using context-aware neural networks. *Nature Communications*, 16(1):3205, 2025.

[22] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

[23] Carlos Outeiral and Charlotte M Deane. Codon language embeddings provide strong signals for use in protein engineering. *Nature Machine Intelligence*, 6(2):170–179, 2024.

[24] Žiga Avsec, Natasha Latysheva, Jun Cheng, Guido Novati, Kyle R Taylor, Tom Ward, Clare Bycroft, Lauren Nicolaisen, Eirini Arvaniti, Joshua Pan, et al. Alphagenome: advancing regulatory variant effect prediction with a unified dna sequence model. *bioRxiv*, pages 2025–06, 2025.

[25] Tao Shen, Zhihang Hu, Siqi Sun, Di Liu, Felix Wong, Jiuming Wang, Jiayang Chen, Yixuan Wang, Liang Hong, Jin Xiao, et al. Accurate rna 3d structure prediction using a language model-based deep learning approach. *Nature Methods*, pages 1–12, 2024.

[26] Yekaterina Shulgina, Marena I Trinidad, Conner J Langeberg, Hunter Nisonoff, Seyone Chithrananda, Petr Skopintsev, Amos J Nissley, Jaymin Patel, Ron S Boger, Honglue Shi, et al. Rna language models predict mutations that improve rna function. *Nature Communications*, 15(1):1–17, 2024.

[27] Dennis R Goulet, Yongqi Yan, Palak Agrawal, Andrew B Waight, Amanda Nga-sze Mak, and Yi Zhu. Codon optimization using a recurrent neural network. *Journal of Computational Biology*, 30(1):70–81, 2023.

[28] Shashank Pathak and Guohui Lin. Pre-trained protein language model for codon optimization. *bioRxiv*, pages 2024–12, 2024.

[29] Zena A Kadhuim and Samaher Al-Janabi. Codon-mrna prediction using deep optimal neurocomputing technique (dlstm-dsn-woa) and multivariate analysis. *Results in Engineering*, 17:100847, 2023.

[30] David K Yang, Samuel L Goldman, Eli Weinstein, and Debora Marks. Generative models for codon prediction and optimization. *Machine Learning in Computational Biology*, 2019.

[31] Jiayang Chen, Zhihang Hu, Siqi Sun, Qingxiong Tan, Yixuan Wang, Qinze Yu, Licheng Zong, Liang Hong, Jin Xiao, Tao Shen, et al. Interpretable rna foundation model from unannotated data for highly accurate rna structure and function predictions. *arXiv preprint arXiv:2204.00300*, 2022.

[32] Haopeng Yu, Heng Yang, Wenqing Sun, Zongyun Yan, Xiaofei Yang, Huakun Zhang, Yiliang Ding, and Ke Li. An interpretable rna foundation model for exploring functional rna motifs in plants. *Nature Machine Intelligence*, 6(12): 1616–1625, 2024.

[33] Niv Sabath, Andreas Wagner, and David Karlin. Evolution of viral proteins originated de novo by overprinting. *Molecular biology and evolution*, 29(12):3767–3780, 2012.

[34] Giorgio Bernardi. Isochores and the evolutionary genomics of vertebrates. *Gene*, 241(1):3–17, 2000.

[35] Manolo Gouy and Christian Gautier. Codon usage in bacteria: correlation with gene expressivity. *Nucleic acids research*, 10(22):7055–7074, 1982.

[36] Ronny Lorenz, Stephan H Bernhart, Christian Höner zu Siederdissen, Hakim Tafer, Christoph Flamm, Peter F Stadler, and Ivo L Hofacker. Viennarna package 2.0. *Algorithms for molecular biology*, 6:1–14, 2011.

[37] Hiroaki Sakoe and Seibi Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal processing*, 26(1):43–49, 2003.

[38] Limin Fu, Beifang Niu, Zhengwei Zhu, Sitao Wu, and Weizhong Li. Cd-hit: accelerated for clustering the next-generation sequencing data. *Bioinformatics*, 28(23):3150–3152, 2012.

[39] Dov Greenbaum, Christopher Colangelo, Kenneth Williams, and Mark Gerstein. Comparing protein abundance and mrna expression levels on a genomic scale. *Genome biology*, 4:1–8, 2003.

[40] Harvy Clyde Carver, A O'TOOLE, and T Raiford. *The annals of mathematical statistics*. Edwards Bros., 1930.

[41] J R Lobry. Asymmetric substitution patterns in the two dna strands of bacteria. *Molecular Biology and Evolution*, 13 (5):660–665, 05 1996. ISSN 0737-4038. doi: 10.1093/oxfordjournals.molbev.a025626. URL https://doi.org/10.1093/oxfordjournals.molbev.a025626.

[42] Michael Lynch and John S Conery. The origins of genome complexity. *science*, 302(5649):1401–1404, 2003.

[43] Thomas F Clarke IV and Patricia L Clark. Rare codons cluster. *PloS one*, 3(10):e3412, 2008.

[44] Mingcong Wang, Christina J Herrmann, Milan Simonovic, Damian Szklarczyk, and Christian von Mering. Version 4.0 of paxdb: protein abundance data, integrated across model organisms, tissues, and cell-lines. *Proteomics*, 15(18): 3163–3168, 2015.

[45] Stefan Klumpp, Jiajia Dong, and Terence Hwa. On ribosome load, codon bias and protein abundance. *PloS one*, 7 (11):e48542, 2012.

[46] Dmitry A Rodionov. Comparative genomic reconstruction of transcriptional regulatory networks in bacteria. *Chemical reviews*, 107(8):3467–3497, 2007.

[47] Bryan J Venters and B Franklin Pugh. How eukaryotic genes are transcribed. *Critical reviews in biochemistry and molecular biology*, 44(2-3):117–141, 2009.

[48] Boxuan Simen Zhao, Ian A Roundtree, and Chuan He. Post-transcriptional gene regulation by mrna modifications. *Nature reviews Molecular cell biology*, 18(1):31–42, 2017.

[49] Paul J. Sample, Botao Wang, David W. Reid, Vladimir Presnyak, Ivan J. McFadyen, Daniel R. Morris, and Georg Seelig. Human 5 utr design and variant effect prediction from a massively parallel translation assay. *Nature Biotechnology*, 37(7):803–809, 2019.

[50] Alexander Karollus, Žiga Avsec, and Julien Gagneur. Predicting mean ribosome load for 5'utr of any length using deep learning. *PLoS Computational Biology*, 17(5):e1008982, 2021.

[51] Weizhong Zheng, Xinyu Lin, Yong Zhang, Qi Zhang, Zhiya Qiu, Lei Chen, and Xin Gao. Discovery of regulatory motifs in 5 untranslated regions using interpretable multi-task learning models. *Cell Systems*, 14(12):1103–1112, 2023.

[52] Manato Akiyama and Yasubumi Sakakibara. Informative rna base embedding for rna structural alignment and clustering by deep representation learning. *NAR Genomics and Bioinformatics*, 4(1):lqac012, 2022.

[53] Yi Liu. A code within the genetic code: codon usage regulates co-translational protein folding. *Cell Communication and Signaling*, 18(1):145, 2020. doi: 10.1186/s12964-020-00642-1.

[54] Shi Zhang, Emanuel Goldman, and Geoffrey Zubay. Clustering of low usage codons and ribosome movement. *Journal of Theoretical Biology*, 170(4):339–354, 1994. doi: 10.1006/jtbi.1994.1181.

[55] Yokota Takashi, Sugisaki Hiroyuki, Takanami Mitsuru, and Kaziro Yoshito. The nucleotide sequence of the cloned tufa gene of escherichia coli. *Gene*, 12(1-2):25–31, 1980.

[56] Nuala A O'Leary, Matthew W Wright, Jeremy R Brister, Sandra Ciufo, Diane Haddad, Richard McVeigh, Bhavna Rajput, Ben Robbertse, Sahoo, Dhruba K and, et al. Reference sequence (refseq) database at ncbi: current status, taxonomic expansion, and functional annotation. *Nucleic Acids Research*, 44(D1):D733–D745, 2016. doi: 10.1093/nar/gkv1188.