BiRNA-BERT Allows Efficient RNA Language Modeling with Adaptive Tokenization

Md Toki Tahmid[1*], Haz Sameen Shahgir[2*†], Sazan Mahbub[3],
Yue Dong[2], and Md. Shamsuzzoha Bayzid[1]

[1]Bangladesh University of Engineering and Technology
[2]University of California Riverside
[3]Carnegie Mellon University

July 2, 2024

Abstract

Recent advancements in Transformer-based models have spurred interest in their use for biological sequence analysis. However, adapting models like BERT is challenging due to sequence length, often requiring truncation for proteomics and genomics tasks. Additionally, advanced tokenization and relative positional encoding techniques for long contexts in NLP are often not directly transferable to DNA/RNA sequences, which require nucleotide or character-level encodings for tasks such as 3D torsion angle prediction. To tackle these challenges, we propose an adaptive dual tokenization scheme for bioinformatics that utilizes both nucleotide-level (NUC) and efficient BPE tokenizations. Building on the dual tokenization, we introduce BiRNA-BERT, a 117M parameter Transformer encoder pretrained with our proposed tokenization on 36 million coding and non-coding RNA sequences. BiRNA-BERT achieves state-of-the-art results in long-sequence downstream tasks and achieves a performance comparable to 6× larger models in short-sequence tasks with 27× less pre-training compute. In addition, our empirical experiments and ablation studies demonstrate that NUC is often preferable over BPE for bioinformatics tasks, given sufficient VRAM availability. This further highlights the advantage of BiRNA-BERT, which can dynamically adjust its tokenization strategy based on sequence length, utilizing NUC for shorter sequences and switching to BPE for longer ones, thereby eliminating the need for truncation. [1]

1 Introduction

The introduction of encoder-only transformer models has revolutionized Natural Language Processing (NLP) by significantly improving our ability to extract deep semantic representations from text which can be used for a wide range of downstream tasks [13, 24]. This success has inspired researchers to apply the pretraining-finetuning paradigm to a wide range of topics beyond NLP, including biological sequence modeling with remarkable success [6, 14, 20, 27, 51]. However, transferring improvements from natural language processing (NLP) to biological sequences is not always straightforward. A key challenge arises from the need to model extremely

*Contributed equally to this work
†Corresponding author: hshah057@ucr.edu
[1]The code and model weights are available at https://github.com/buetnlpbio/BiRNA-BERT

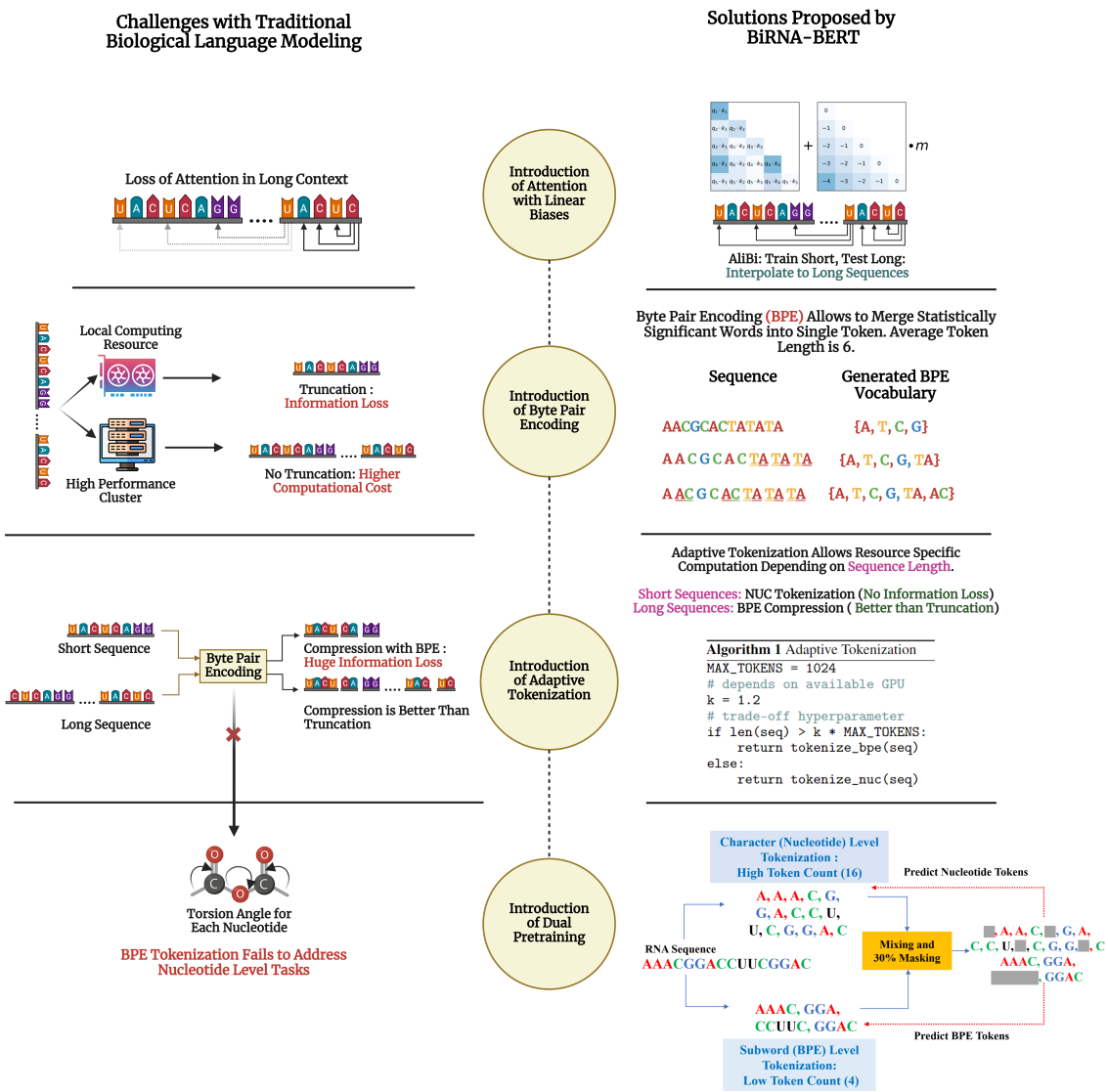

Figure 1: Overall architecture of BiRNA-BERT. The issue of loss of attention in long sequences is resolved by using AliBi positional encoding in BiRNA-BERT. In low compute resources, where large sequences can not be fitted into the memory, we do not have actually any alternate other that truncating the sequences, even if we use AliBi encoding. To resolve this issue BiRNA-BERT uses Byte Pair Encoding(BPE) tokenization which allows to merge statistically significant residues into a single token, allowing longer context length support within limited compute resource. Even though BPE compression works better than seqeunce truncation in long sequences, BPE has absolute information loss which results in poor performance in short sequences. Moreover, by performing BPE tokenization we lose the residue level information and can not perform tasks like secondary structure prediction and tortion angle prediction. To overcome this issue BiRNA-BERT proposes dual tokenization and pretraining. If sequcnes are small, we perform nuc tokenization, if long we perform BPE tokenization instead of truncation which allows a generalized solution to the issues with biological language modeling.

long sequences for certain tasks, as biological sequences can range from tens to millions of nucleotides or amino acids. The standard transformer encoder architecture (e.g., BERT [13]) uses positional embeddings that limit the maximum input length to usually 512 or 1024, requiring truncation or inefficient workarounds for biological sequences.

In addition, despite recent advancements in methodological and architectural research efforts to scale up the length for BERT [7,17,28] by addressing the limited context window [29,40] or with more efficient tokenization methods [16,36], they have not yet been successfully adopted by bioinformatics without sacrificing performance or sample efficiency. The limited number of approaches to adopting new positional encoding and improved pretraining developed in NLP often fall short in biological sequencing. For example, Nucleotide Transformers (NT) [11] replace nucleotide-level tokenization (NUC) with non-overlapping k-mers, where k-mers represent subtokens of fixed length k, enabling the model to handle sequences that are k times longer while sacrificing granularity. This limitation is acknowledged in DNABERT-2 [51], where they demonstrate the poor sample efficiency of k-mer tokenization and propose adapting BPE tokenization for DNA sequences. However, their approaches still suffer from granularity loss by combining common subsequences into one token, and the original paper does not provide a controlled comparison between NUC and BPE. Moreover, in domains like RNA or protein sequences where secondary structure prediction or granular alignment tasks are important, simply using BPE tokenization would prevent the trained foundation model from generating nucleotide- or amino acid-level predictions, which are essential for these additional tasks.

Given the abovementioned limitations, we propose **Bi**-tokenization **RNA BERT** (BiRNA-BERT) a Transformer encoder model for RNA sequences pretrained on both NUC and BPE tokens of the same RNA sequences simultaneously. BiRNA-BERT uses Attention with Linear Biases (ALiBi) which allows the context window to be extended without retraining and can dynamically choose between NUC and BPE tokenization based on the input sequence length. For shorter sequences, it utilizes NUC to capture fine-grained patterns, and for longer sequences, it switches to more efficient BPE tokenization to reduce memory requirements without truncating the input. This dynamic tokenization and context length expansion allows BiRNA-BERT to enable downstream tasks with arbitrarily long sequences and set state-of-the-art results on 4 out of 6 tasks of miRNA-lncRNA interaction dataset [46]. Along with sequence-level tasks, the dual tokenization approach allows us to conduct nucleotide-level analyses, such as RNA 3D torsion angle prediction [39], as additional concurrent downstream tasks.

We conduct further downstream ablation experiments on both RNA and DNA data to demonstrate that NUC tokens yield better downstream task performance than BPE when enough resources are available to handle the short sequences. In addition, the joint-pretraining strategy does not compromise nucleotide-level performance compared to models trained on only NUC tokens.

In summary, BiRNA-BERT demonstrates the effectiveness and importance of utilizing proper tokenization tailored to the unique demands of bioinformatics tasks. Pre-training models leveraging our proposed tokenization can achieve significant performance gains across a range of biological sequence analysis tasks. Our main contributions can therefore be summarized as follows:

1. We present dual tokenization pretraining -an effective approach that extends the effective context window of biological foundation models with efficient tokenization while retaining the ability to generate character-level embeddings for the first time.

2. Using dual tokenization and ALiBi, we train BiRNA-BERT which achieves absolute state-of-the-art results on long-sequence tasks and is comparable to 6× larger models on short-sequence and nucleotide-level tasks while being trained with 27× less pretraining compute.

BiRNA-BERT can dynamically adjust the tokenization algorithm based on the sequence length and available compute.

3. We provide a rigorous mathematical analysis from information theoretic perspective about the absolute information loss of token compression with byte pair encoding for the first time. With that, we show that NUC outperforms BPE on tasks where sequences are short enough to fit into GPU memory and the trade-off between sequence truncation and information compression works in favour of the later when long sequences are in consideration.

4. We validate the adaptability of our proposed scheme to other biological language models beyond RNA sequences by extending the idea of dual tokenization with AliBi encoding to DNA sequences too with BiDNA-BERT which shows very competitive performance to the DNA foundation models, even being 66x less compute heavy.

2 Related Work

The application of NLP techniques to biological research has seen renewed interest following the success of AlphaFold [21] in predicting protein structures. While the majority of research efforts have focused on protein sequences with considerable success, recent interest has begun to shift towards modeling DNA and RNA sequences. Bioinformatics researchers aim to develop *foundation models*—models pre-trained on vast amounts of unlabelled biological data that can be fine-tuned for a range of downstream tasks where labeled data is limited. Depending on the downstream tasks, different variants of the Transformer architecture such as encoder-only BERT [13], encoder-decoder T5 [30], and decoder-only GPT [4] have been applied to Bioinformatics.

Protein Foundation Models [32] proposed ESM-1B, a RoBERTa-based model pre-trained on 250 million protein sequences which achieved state-of-the-art results on the secondary structure and tertiary contact prediction of proteins. Similar BERT-style models include ProtBERT [3], TAPE [31], ESM-2 [23], and OmegaPLM [47]. Generative models such as GPT have also been adapted to generate protein sequences with specific biophysical properties such as ProtGPT2 [15], ProGen [25], and xTrimoPGLM [5].

DNA Foundation Models Notable foundation models for DNA include DNABERT [20], DNABERT-2 [51], and Nucleotide Transformer [11], all of which are BERT-style encoder-only models largely following the pretrain-then-finetune paradigm. DNABERT is a 110M BERT-based model pretrained on human-genome datasets. Nucleotide transformer (NT) scales up both pretraining data and model size by training on DNA sequences from 850 species including humans and training multiple models from 500M to 2.5B parameters. DNABERT-2 improved upon DNABERT by training on the same dataset as NT [10] and was the first to use BPE tokenization for biological sequence modeling [16, 36]. DNABERT-2 is comparable to 2.5B NT in terms of downstream task performance, but despite having 21× fewer parameters and approximately 92× less GPU time in pre-training [51].

RNA Foundation Models [6] proposed RNA-FM, a 100 million parameters model pre-trained on 23.7 million unannotated ncRNAs from the RNAcentral database [8] using the standard masked language modeling task and nucleotide-level tokenization (NUC). Uni-RNA [45] was pre-trained on a dataset of 1 billion RNA sequences for a spectrum of structural and

functional downstream tasks. [50] proposed a Multiple sequence alignment (MSA)-based RNA language model (RNA-MSM), a BERT-style language model that utilizes a set of homologous sequences per forward propagation and produced attention maps and embeddings that have direct correlations to RNA secondary structure and solvent accessibility without supervised training. For downstream tasks, RNA-MSM finds homologous sequences of the input sequence from a stored database, computes the MSA matrix using RNAcmap [49], and uses information from the MSA as input to the model. However, obtaining the MSAs is a time-consuming procedure, for example, it takes RNAcmap on average 9 hours to obtain an MSA for one RNA sequence of length 60. [27] introduced RiNALMo, a 650M parameter model that incorporates algorithm and architecture improvements in NLP models, namely Rotary Position Encoding (RoPE) [40], FlashAttention2 [12], and SwiGLU [37], and is pre-trained on 36 million ncRNA sequences from RNAcentral [8], NT, Rfam [18], and Ensembl databases [9]. RiNALMo achieves state-of-the-art performance on a variety of downstream tasks such as splice-site and mean ribosome loading and exhibits stronger generalization than previous thermodynamics-based and deep learning models on secondary structure prediction.

All the aforementioned biological foundation models either use character-level (nucleotide-level) tokenization or BPE tokenization, each with its trade-offs. Approaches using nucleotide-level tokenization inevitably face long tokenized sequences, while approaches using BPE, such as DNABERT-2, do not cater to tasks requiring nucleotide-level granularity. In genomics and proteomics, we have a number of tasks with long sequences, for instance, the mRNA sequence database on NCBI [35] has an average length of 4,175 nucleotides. Additionally, tasks like secondary structure prediction [42] and torsion angle prediction [39] often require generating features at the nucleotide level. Hence, it's crucial to manage longer sequences without compromising nucleotide-level analysis, where our proposed dual tokenization balances this trade-off effectively.

3 Methods

In this section, we first describe the applicability and methodology of using BPE tokenization for BiRNA-BERT (Subsection 3.1). Then we describe the motivation behind using relative positional encoding (ALiBi) for extrapolating the trained model to longer sequences in downstream tasks (Section 3.2). In 3.3, we discuss the dual tokenization pretraining approach and its motivation. Finally, we describe the pretraining configurations and datasets used in Section 3.5.

3.1 Tokenization Strategies for Biological Foundational Models

Nucleotide-level tokenization (NUC) has long been the dominant form of tokenization in biological sequence modeling [6,27], partially due to the need for NUC embeddings in secondary tasks for RNA and protein. An alternative form of tokenization is k-mer tokenization, where segments of k nucleotides are considered as one token. The motivation for k-mer tokenization is to effectively capture the biological significance of RNA sequences, as certain k-mers, such as 3-mers (codons), directly correspond to amino acids, the building blocks of proteins. DNABERT [20] uses overlapping k-mers while models such as Nucleotide Transformers [11] use non-overlapping k-mers. Recently, DNABERT-2 [51] has validated the superiority of BPE tokenization on DNA over NUC and k-mer tokenization. DNA has no major downstream task that requires NUC embeddings and as a result, BPE tokenization was well suited for the task.

Nucleotide-level Tokenization Nucleotide-level tokenization involves representing each nucleotide in a sequence as an individual token. For example, given the RNA sequence:

$$\text{Sequence} = \text{AUGGCUACUGCAUGCUAGUCA} \tag{1}$$

This sequence would be tokenized into:

$$\text{Tokens} = [\text{A, U, G, G, C, U, A, C, U, G, C, A, U, G, C, U, A, G, U, C, A}] \tag{2}$$

Byte-Pair Encoding (BPE) Tokenization Byte-Pair Encoding (BPE) [16,36] is a subword tokenization technique that iteratively merges the most frequent pairs of bytes or characters to create new tokens, thereby reducing the vocabulary size to a fixed number.

BPE Tokenization in RNA Sequence Modeling In the context of RNA sequences, the variability and complexity of the nucleotide sequences pose a challenge for traditional tokenization methods. Fixed k-mer-based tokenization can result in an excessively large and sparse vocabulary of size $4^k + 5$ [48], as it captures all possible k-mers regardless of their biological significance or frequency. BPE tokenization, on the other hand, leverages the statistical frequency of sub-sequences to create a more compact and meaningful vocabulary. By iteratively merging the most frequent pairs of sub-sequences, BPE ensures that commonly occurring patterns are represented by single tokens, while less frequent patterns are broken down into smaller units. This process is particularly beneficial for RNA sequences, where certain motifs and regions (e.g., hairpin loops, binding sites) occur frequently and are biologically significant. DNABERT-2 [51] tested the effect of the BPE vocabulary size on downstream tasks and empirically showed that a vocabulary size of 4096 is optimal. Following DNABERT-2, we use a vocabulary size of 4096 for the BPE tokenization.

For example, starting with the same RNA sequence:

$$\text{Sequence} = \text{AUGGCUACUGCAUGCUAGUCA} \tag{3}$$

Assume our initial vocabulary includes individual nucleotides: {A, U, G, C}. The BPE algorithm might merge frequent pairs like "AU" and "GC" in the following steps:

1. Identify the most frequent pair of characters. Let's say "AU" appears most frequently.
2. Merge "AU" into a single token. Update the sequence:

$$\text{Sequence} = \text{AUGGCUACUGCAUGCUAGUCA} \tag{4}$$

3. Identify the next most frequent pair, e.g., "GC". Merge "GC" into a single token. Update the sequence:

$$\text{Sequence} = \text{AUGGCUACUGCAUGCUAGUCA} \tag{5}$$

4. Continue this process until the predefined vocabulary size is reached.

The final tokenized sequence might look like:

$$\text{Tokens} = [\text{AU, GGC, UAC, UGC, AUG, CUA, GUC, A}] \tag{6}$$

3.2 Positional Encoding in the Transformer Architecture

Since the attention mechanism is permutation-invariant, positional information must be explicitly added. There are two main strategies for encoding positional information - fixed and relative. In fixed positional encoding schemes such as sinusoidal [43] or learned embeddings [13], the positional information is a vector function of the position index within some predefined context length. Since the positional information is explicit in the form of a vector, these methods cannot extrapolate to context lengths beyond those seen in pretraining. Previous biological foundation models [6, 20] have mostly used fixed positional embeddings following the original BERT implementation. As a result, long sequences had to be truncated (usually to 1024 tokens) on downstream tasks. In relative positional encoding, the positional information is a function of the distance between two tokens. Popular algorithms for relative positional embeddings are T5 Bias [30], Rotary Positional Embedding (RoPE) [41], and Attention with Linear Biases (ALiBi) [29]. Recently, RiNALMo [27] utilized Rotary Positional Embedding (RoPE) but still truncated sequences to 1022 nucleotides since extending the context window using RoPE requires additional training to preserve performance. DNABERT-2 [51] was the first biological foundation model to use ALiBi, which when combined with BPE, allowed them to process sequences of up to 10000 nucleotides, significantly outperforming recurrent models such as HyenaDNA [26] on long-sequence tasks.

Attention with Linear Biases (ALiBi) In contrast to complex methods such as T5 Bias and RoPE which are hard for models to extrapolate without continued pretraining, ALiBi simply reduces the attention score between two tokens by a scaler function of their distance.

Let $\mathbf{Q} \in \mathbb{R}^{L\times d}$, $\mathbf{K} \in \mathbb{R}^{L\times d}$, and $\mathbf{V} \in \mathbb{R}^{L\times d}$ represent the query, key, and value matrices, where L is the sequence length and d is the hidden dimension. ALiBi modifies the attention scores by adding a bias term, a function of the positional distance between the query and key elements. Specifically, for positions i and j, the linear bias term b_{ij} is defined as $b_{ij} = \alpha \cdot |i-j|$, where α is a hyperparameter. This results in modified attention scores $\mathbf{A}_{ij} = \frac{\mathbf{Q}_i \mathbf{K}_j^\top}{\sqrt{d}} - b_{ij}$. Since ALiBi explicitly penalized the attention scores between distance tokens, the α values must be carefully chosen to preserve long-context capabilities. The authors of ALiBi use different α values for different attention heads, uniformly sampled from the geometric progression $\frac{1}{2}, \frac{1}{2^2}, ..., \frac{1}{2^8}$. Intuitively, some attention heads specialize in aggregating local information while others preserve long-context information.

3.3 Pretraining with Dual Tokenization

The primary motivation for training a single model with dual tokenization is memory efficiency. Although having separate BPE and NUC models is viable, we argue that a single model training on both is memory-efficient since only one set of weights needs to be on memory and can process longer sequences and larger batch sizes. Memory efficiency is crucial since the transformer architecture has $\mathcal{O}(n^2)$ memory requirement which especially penalizes long sequence tasks. In addition to BiRNA-BERT, We train two more models BPE-Only-BERT and NUC-Only-BERT to show that dual tokenization does not affect downstream performance. Performing BPE tokenization on long sequences effectively compresses the sequence with average BPE token length $\bar{L} \approx 6.0768$. We show in the subsequent sections that the empirical per-character entropy ratio is:

$$\frac{\hat{H}_e(X_{BPE})}{\hat{H}_e(X_{NUC})} \approx 0.7514 < 1$$

where $\hat{H}_e(X_{BPE})$ is the average character-level entropy of the BPE representation of a sequence and $\hat{H}_e(X_{NUC})$ is the per-character entropy for nucleotide tokenization. Therefore, BPE tokenization is essentially a trade-off between information compression and computational efficiency,

Given an RNA sequence $\mathbf{r} = (r_1, r_2, \ldots, r_n)$, we apply two types of tokenization: Byte Pair Encoding (BPE) and nucleotide-level (Nuc).

- **BPE Tokenization:** The BPE tokenized sequence is $\mathbf{r}_{BPE} = (\beta_1, \beta_2, \ldots, \beta_m)$ with a vocabulary size of 4096.
- **Nucleotide-Level Tokenization:** The nucleotide tokenized sequence is $\mathbf{r}_{Nuc} = (\nu_1, \nu_2, \ldots, \nu_n)$ with a vocabulary size of 4.

We randomly mask tokens in both $\mathbf{r}_{BPE}$ and $\mathbf{r}_{Nuc}$ for Masked Language Modeling (MLM) training:

- **Masked BPE Sequence:** $\mathbf{r}_{BPE}^{mask}$ where a subset of β_i is replaced with a mask token $[MASK]$.
- **Masked Nucleotide Sequence:** $\mathbf{r}_{Nuc}^{mask}$ where a subset of ν_i is replaced with a mask token $[MASK]$.

The total loss $\mathcal{L}_{total}$ is the sum of the BPE and nucleotide losses:

$$\mathcal{L}_{total} = -\sum_{i \in \mathcal{M}_{BPE}} \log P(\beta_i | \mathbf{r}_{BPE}^{mask}) - \sum_{i \in \mathcal{M}_{Nuc}} \log P(\nu_i | \mathbf{r}_{Nuc}^{mask}).$$

The parameters of the BERT model, Θ, are optimized to minimize the total loss:

$$\Theta^* = \arg\min_{\Theta} \mathcal{L}_{total}.$$

3.4 Pretraining Configuration

We pretrain three BERT-based RNA models to investigate the effect of dual tokenization: BiRNA-BERT, BPE-Only-BERT, and NUC-Only-BERT. We use the MosaicML framework [28] for efficient training. We pretrain all models with a learning rate (LR) of 2×10^{-4}, a warmup ratio of 0.06, a linear rate scheduler that decreases to 0.02 of the starting LR, and a batch size of 200 per device, totaling 1600 samples per batch across eight Nvidia RTX 3090 GPUs. Due to the investigative nature of our work and compute constraints, we pretrain each model for only 1 epoch. The training configurations are summarized in Table 1.

Table 1: Pretraining details for various models

Model	Train Tokens	Train Time	Hardware
RNA-BPE only	4.384B	3 hours	8×3090
RNA-NUC only	27.87B	45.1 hours	8×3090
RNA-BiRNA	32.254B	48.42 hours	8×3090
DNA-BPE only	586.854M	2.42 hours	1×4090
DNA-NUC only	3.027B	12 hours	1×4090
DNA-BiDNA	3.614B	14.33 hours	1×4090

3.5 Pretraining Dataset

For pretraining our models, we consider both non-coding (ncRNA) and coding/messenger (mRNA) RNA databases. ncRNA sequences do not encode proteins but play crucial roles in regulating gene expression and maintaining cellular functions. In contrast, mRNA sequences serve as templates for protein synthesis, directly influencing the proteome of the cell.

Non-coding RNA Sequences: We use the RNAcentral database [8], a comprehensive compilation of multiple ncRNA databases. There are 36 million RNA sequences in RNAcentral with a total of 26.42 billion nucleotides.

Coding RNA Sequences: We utilized mRNA sequences from the RefSeq database [2]. For each species in the database, we considered the mRNA sequences, resulting in a total of 532,852 mRNA sequences with an average length of 4,175 nucleotides.

4 Results

This section highlights the results from pretraining BiRNA-BERT and its performance on various downstream tasks. Section 4.1 covers the unsupervised clustering performance of BiRNA-BERT across different tasks including RNA secondary structural clustering, and RNA family clustering.

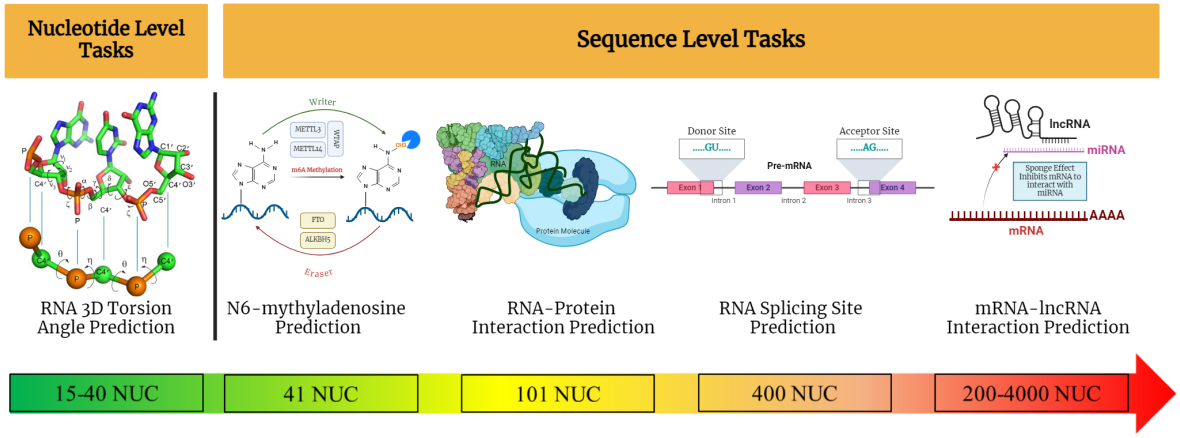

Figure 2: From left to right the sequence length of the concerned tasks increases. We consider one nucleotide level task and four sequence level tasks. Nucleotide level task include RNA 3d torsion angle prediction. Sequence level tasks include: RNA N6-mythyladenosine prediction, RNA-protein interaction prediction, RNA splicing site prediction and mRNA-IncRNA interaction prediction.

We perform a number of downstream tasks of varying sequence lengths, as shown in Figure 2. In Section 4.2, we demonstrate that BiRNA sets new state-of-the-art results on long sequence RNA-RNA interaction by leveraging dynamic tokenization which the previous language models handled with truncation. However, BiRNA-BERT uses the information compression technique using BPE and encodes the long sequences with minimal information loss hence providing significant performance gain over the traditional models.

[2] https://ftp.ncbi.nlm.nih.gov/refseq/release/

We cover short sequence downstream tasks in Section 4.4. BiRNA-BERT significantly surpasses similar-sized models such as RNA-FM and BERTRBP on short-sequence tasks.

In Section 4.3, we demonstrate that along with sequence level tasks shown already, BiRNA-BPE also has the simultaneous capability of handling nucleotide level tasks efficiently which we evaluate with RNA 3d torsion angle prediction task.

In section 4.4, we provide a rigorous mathematical analysis of the dual tokenization scheme from an information theoretic perspective which demonstrates the information loss from the BPE compression and how it compensates with the available GPU memory constraints. Moreover, we perform two different downstream tasks of RNA-Protein interaction and RNA N6-methyladenosine sites prediction to empirically show the validity of the information theoretic analysis. We demonstrate that, when the sequence lengths are small enough to be handled by the available GPU memory, nucleotide tokenization consistently outperforms the BPE compression scheme. Which strengthens our proposal of using the adaptive tokenization scheme (Nucleotide tokenization for short sequences and shifting to BPE compression for much longer sequences).

It is to be noted that, BiRNA-BERT is computationally magnitude times (27X) smaller architecture than the largest RNA language model RiNALMo [27] . Moreover, we trained the BiRNA-BERT for only 1 epoch. Even with such constrained architecture, BiDNA-BERT significantly outperforms all foundation models by large margin in long sequence tasks, which remained one of the most challenging tasks in biological language modeling till date. Moreover, in the unsupervised clustering tasks, we show the superior performance of BiRNA-BERT over the current state of the art models which clearly indicates BiRNA-BERT's better understanding capability of the language of RNA genomic language.

With the adaptive tokenization scheme, BiRNA-BERT also provides comparable performance in short sequence tasks however, not always surpassing the largest foundation models, which can be achieved in a straightforward way by increasing the language model size as as downstream performance is directly aligned with the model size of the BERT based language models [38].

In section, 4.6 we validate the significance of dual tokenization based approarch by training a smaller language model *BiDNA-BERT* for DNA sequences, which is 66 times smaller architecture than the state of the art DNA language model DNABERT-2 [51]. We show that, BiDNA-BERT provides comparable performance to DNABERT-2 on the human GLU benchmark.

4.1 Unsupervised Clustering Performance

To understand BiRNA-BERT's capacity of unsupervised representation of the embedding space for RNA sequences from different structural families, we perform two different unsupervised clustering tasks. For the first task, we perform unsupervised clustering on 9 RNA structural families: 16s, 23s, 5s, RNaseP, grp1, srp, tRNA, telomerase, tmRNA. For the second task, we cluster the thirty RNA families from the rfam database. Here we compare unsupervised clustering performance between RNA-FM, RiNALMo and BiRNA-BERT.

The Silhouette Coefficient is a metric used to evaluate the quality of clustering. It measures how similar a data point is to its own cluster compared to other clusters. The coefficient for a data point i is defined as:

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \tag{7}$$

Here, $a(i)$ represents the average distance between the data point i and all other points

within the same cluster. In contrast, $b(i)$ is the average distance between the data point i and all points in the nearest cluster to which i does not belong. The value of $s(i)$ ranges from -1 to 1. A value of 1 indicates that the data point is well-matched to its own cluster and poorly matched to neighboring clusters. A value of 0 indicates that the data point is on or very close to the decision boundary between two neighboring clusters. Negative values suggest that the data point might have been assigned to the wrong cluster. A higher Silhouette Coefficient indicates better-defined clusters.

Table 2: Clustering Performance(Silhouette Coefficient) of Different Models

	RNA Structural Family Clustering	RFAM Family Clustering
RNA-FM	0.139	0.238
RINALMO	-0.001	**0.377**
BiRNA-BERT	**0.156**	0.351

We can see from Table 2 that, in the RNA structural family clustering task, BiRNA-BERT outperforms RNA-FM and RiNALMo. In the RNA Rfam clustering task, BiRNA-Bert outperformed RNA-FM by a good margin however, RiNALMo performed better than BiRNA-BERT. There is a clear reason behind the superior performance of RiNALMo in the second task. Rfam dataset is actually one of the datasets in the RNAcentral database that is used for pretraining the language models. As RiNALMo and BiRNA-BERT, both are trained in a masked language model based approach, the longer a model is trained, the better the unsupervised clustering performance will increase which also reflects in the downstream performance. As RiNALMo is trained for 14 days whereas BiRNA-BERT is only trained for 2 days with 6 times smaller model, it is obvious that the clustering performance of BiRNA-BERT on Rfam dataset would not outperform RiNALMo. However, the secondary structural family dataset is not explicitly included in the pretraining database. BiRNA-BERT outperforms all the larger language models in this task which indicates that, the actual unsupervised language understanding capability of BiRNA-BERT is better than all other models when it comes to unseen databases.

4.2 Long-Sequence Task with Dynamic Tokenization: miRNA-lncRNA Interaction Prediction

Long non-coding RNA (lncRNA) related studies frequently involve much longer sequences which necessitated task-specific architectures such as PmliPred [22] and CORAIN [46]. We evaluate miRNA-lncRNA interaction to verify the impact of sequence truncation during feature embedding by previous models and compare it with our approach, which avoids sequence cropping to fit computational memory. Instead, we dynamically compress sequence information using our adaptive tokenization scheme.

We use three benchmarking datasets for the RNA-RNA interaction prediction task compiled by [22]. For evaluation, one benchmarking dataset is used as the training set, and another dataset is used for validation following the strategy used in [46]. Thus, we have 6 train-test combinations and we report performance in all these combinations. The lengths of the sequences in the miRNA dataset are 10 to 50, whereas the lncRNA dataset ranges from 200 to 4000. The length distributions of sequences for the miRNA-lncRNA Interaction Prediction task are shown in Figure 4.

This is a binary classification task to determine whether a miRNA-lncRNA pair interacts or not. For fine-tuning, we follow the methodology of CORAIN [46] by freezing the backbone encoders, concatenating output features of miRNA and lncRNA, and using a simple convolu-

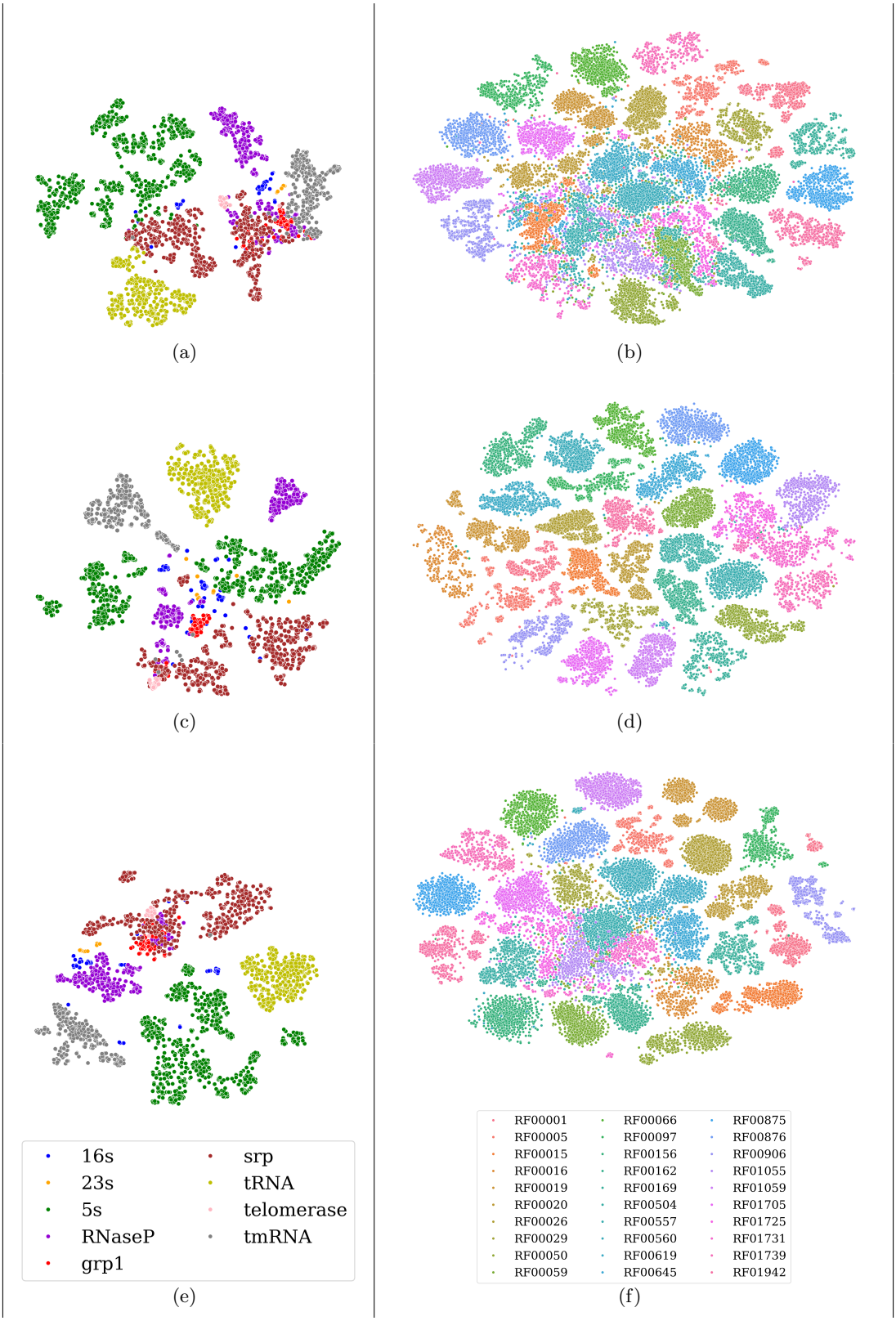

Figure 3: Comparison of unsupervised clustering performance on RNA structural families and RNA Rfam families. (a)RNA structural family clustering with RNA-FM (b)RNA Rfam family clustering with RNA-FM (c)RNA structural family clustering with RiNALMo (d)RNA Rfam family clustering with RiNALMo (e)RNA structural family clustering with BiRNA-BERT (f)RNA Rfam family clustering with BiRNA-BERT.

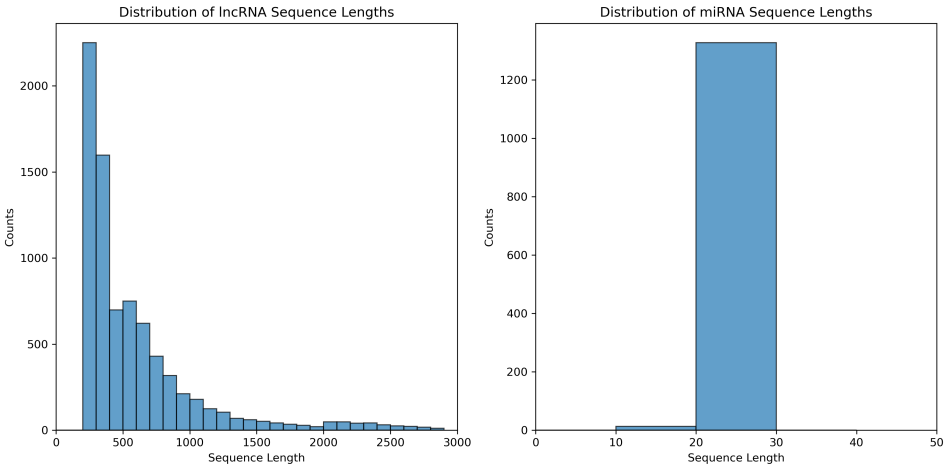

Figure 4: Sequence length distribution for lncRNA and miRNA datasets

Table 3: Accuracy of different models on miRNA-lncRNA dataset. CORAIN [46] is a task-specific CNN-autoencoder and the current state-of-the-art. We fine-tuned RNA-FM, RiNALMo, and both variants of BiRNA-BERT with optimal hyperparameters found using grid search.

Model	**Train-Test Dataset**					
	ATH-GMA	ATH-MTR	GMA-ATH	GMA-MTR	MTR-ATH	MTR-GMA
CORAIN	69	74	67	93	58	84
RNA-FM	68.56	71.68	69.90	94.68	57.15	83.92
RiNALMo	72.14	75.38	**70.93**	**95.08**	65.55	85.82
BiRNA-NUC	72.56	77.90	66.01	94.94	61.05	84.18
BiRNA-BPE	**75.52**	**79.56**	70.55	93.84	**67.75**	**88.54**

tional prediction head. We test two strategies using BiRNA: BPE and NUC with truncation, named BiRNA-BPE and BiRNA-NUC. Both strategies use the same BiRNA models and only the input tokenization scheme differs. We always encode miRNA using NUC due to their short lengths. We use NUC with truncation for lncRNA for RNA-FM, RiNALMo, and BiRNA-NUC. We truncate all inputs to 1022 tokens due to the context window limitation of RNA-FM and RiNALMo even though BiRNA-NUC can process longer sequences due to ALiBi. We do not truncate input sequences to BiRNA-BPE since after BPE tokenization the maximum sequence length is 807, still lower than NUC.

The results in Table 3 offer several interesting insights:

1. RNA-FM performs significantly worse than the non-LM-based approach in 4 of 6 test datasets. However, RiNALMo outperforms the current state-of-the-art in all datasets despite sequence truncation. This is noteworthy, as RiNALMo is a six times larger language model than RNA-FM, significantly enhancing its expressive capability.

2. BiRNA-NUC outperforms RNA-FM in 5 out of 6 datasets and provides comparable performance to RiNALMo, despite being the same size as RNA-FM. BiRNA-BPE outperforms RiNALMo by a substantial margin, with improvements of 4.74%, 5.54%, 3.35%, and 3.16% on the ATH-GMA, ATH-MTR, GMA-MTR, and MTR-GMA datasets. It also offers comparable performance in the GMA-ATH and GMA-MTR datasets, within 0.51% and 1.2% margins.

3. An intuitive way to compare NUC and BPE tokenization is by considering information loss. NUC explicitly truncates sequences to 1022 nucleotides, losing all subsequent information. BPE, on the other hand, compresses the entire sequence. In miRNA-lncRNA interaction tasks, we demonstrate that compression (BPE) is preferable to explicit information loss (NUC), while also being more computationally efficient (807 tokens versus 1024 tokens).

4.3 Nucleotide-Level Task: 3D Torsion Angle Prediction

Table 4: Mean Absolute Error of RNA 3D Torsion Angle Prediction

Data	Method	Total Error	Alpha	Beta	Gamma	Delta	Epsilon	Zeta	Chi
V	RNA-FM	234.34	36.52	35.74	33.58	37.87	30.62	28.56	31.45
	RINALMo	**227.08**	36.20	36.15	31.58	35.25	31.56	28.84	27.51
	BiRNA-NUC	234.37	37.12	36.57	32.92	37.49	31.64	28.90	29.72
	NUC-only	233.62	37.20	36.53	32.57	36.97	31.61	28.76	29.98
T1	RNA-FM	205.06	29.58	29.09	28.64	29.43	29.44	31.33	27.55
	RINALMo	**204.14**	28.51	29.09	27.66	29.25	29.42	30.24	29.98
	BiRNA-NUC	204.29	29.86	29.73	27.83	29.10	29.53	31.26	26.97
	NUC-only	205.91	30.62	29.79	28.05	29.62	29.53	31.21	27.10
T2	RNA-FM	216.04	35.19	34.83	27.25	30.21	23.28	29.27	36.02
	RINALMo	**208.83**	36.87	33.48	25.38	27.19	23.05	27.55	35.33
	BiRNA-NUC	213.98	34.48	34.31	29.01	28.41	22.85	28.69	36.22
	NUC-only	214.89	34.35	35.63	29.02	28.70	22.35	29.02	35.82
T3	RNA-FM	244.65	36.97	38.30	35.27	38.78	31.14	30.58	33.61
	RINALMo	242.89	36.63	38.31	34.64	38.37	30.87	30.40	33.67
	BiRNA-NUC	246.42	36.87	39.88	35.47	38.03	32.39	29.69	34.08
	NUC-only	**239.40**	36.92	36.41	32.71	38.72	30.87	30.45	33.32

There are seven torsion angles commonly referred to as α, β, γ, δ, ϵ, ζ, and χ. These angles describe the rotations around the bonds that connect the nucleotides within an RNA strand, influencing its overall structure and stability. These angles are mathematically represented as the dihedral angles between four consecutive atoms in the RNA backbone. For example, the α angle is measured as the dihedral angle between O5'-P-O3'-C3'. The dataset for RNA torsion angle prediction is collected from https://sparks-lab.org/server/spot-rna-1d/. The training (TR), validation (VL), and three test sets (TS1, TS2, and TS3) have 286, 30, 63, 30, and 54 RNA chains, with average sequence lengths of 122, 15, 30, 14, and 24 respectively. We present the results in Table ??. BiRNA-BERT outperforms RNA-FM on all test datasets and matches closely with NUC-Only-BERT, our model trained using only NUC tokenization. This highlights that dual tokenization pretraining did not compromise the quality of nucleotide embeddings. The performance of both BiRNA-BERT and NUC-Only-BERT are comparable to RiNALMo 650M despite having only 117M parameters and being pretrained on 27× less compute.

4.4 How Close in Short Sequences? : RNA Splicing Site Prediction

We consider the downstream tasks that can be performed within the computational limit of traditional BERT-based architecture as short-sequence tasks. We do not need to truncate the

Table 5: F1 Score for RNA Splicing Site Prediction on Independent Test Sets

Model	FT	Fish	Fly	Plant
RINALMo	✓	0.974	**0.958**	**0.953**
RNA-FM	✓	0.937	0.919	0.877
Spliceator	×	0.919	0.910	0.908
BiRNA-NUC	✓	**0.993**	0.938	0.938

sequences in this case explicitly. We consider the task of binary classification of RNA splicing site prediction specifically for acceptor sites. RNA splicing is a crucial process in eukaryotic gene expression, where introns are removed from precursor messenger RNAs (pre-mRNAs), and exons are joined together to form mature mRNAs. This process is essential for generating functional mRNAs that can be translated into proteins. Identifying splice sites—the donor sites at the 5' end of introns and the acceptor sites at the 3' end—is vital for accurately predicting gene structure and location. For this task, we consider the dataset proposed by [33]. Particularly we use the gold standard dataset GS_1 which contains an equal number of positive and negative samples. The dataset consists of "confirmed" error-free splice-site sequences from a diverse set of 148 eukaryotic organisms, including humans. We have tested the performance of the trained model on three independent test datasets containing the samples from 3 different species of fish (*Danio rerio*), fruit fly (*Drosophila melanogaster*), and plant (*Arabidopsis thaliana*). Here the sequence length is 400 and the train and independent test tests have 20000 sequences each for training and testing respectively. We compare our model with RiNALMo, RNA-FM, and non-LM-based SOTA approaches in Table 5. BiRNA substantially outperforms Spliceator and the similar-sized RNA-FM (BERT) in all the datasets. Bi-RNA outperforms RiNALMo by 1.6% on the Fish dataset and is within 1.3% and 1.6% margins on Fly and Plant datasets.

4.5 Information Theoretic Analysis of Nuelcotide vs BPE Tokenization

In this section, we compare the information content of a nucleotide token and a BPE token inspired by key empirical observations in the training data. Information-theoretic analysis of biological sequences is a well-studied field of research where the key challenges include determining the prior distribution of nucleotides or k-mers, the fact that only a fraction of possible biological sequences occurs in nature and the difficulty in comparing results from biological sequences with those from linguistics due to significant differences morphology [1, 34, 44]. In addition to the information theoretic analysis, we further demonstrate the claims with two RNA downstream tasks in the following discussion.

Information content of a Nucleotide token We consider each nucleotide in a sequence as an independent variable that carries some amount of information. We wish to quantify the maximum amount of information for each new nucleotide in a sufficiently long sequence. We can derive the per-token upper bound of the Shannon Entropy of a DNA/RNA sequence as follows.

$$\begin{aligned} H(X_{NUC}) &= -\sum_{i=1}^{4} P(x_i)\log_2 P(x_i) \\ &\leq -\left(4 \cdot \frac{1}{4}\log_2 \frac{1}{4}\right) \\ &= 2 \text{ bits} \end{aligned} \tag{8}$$

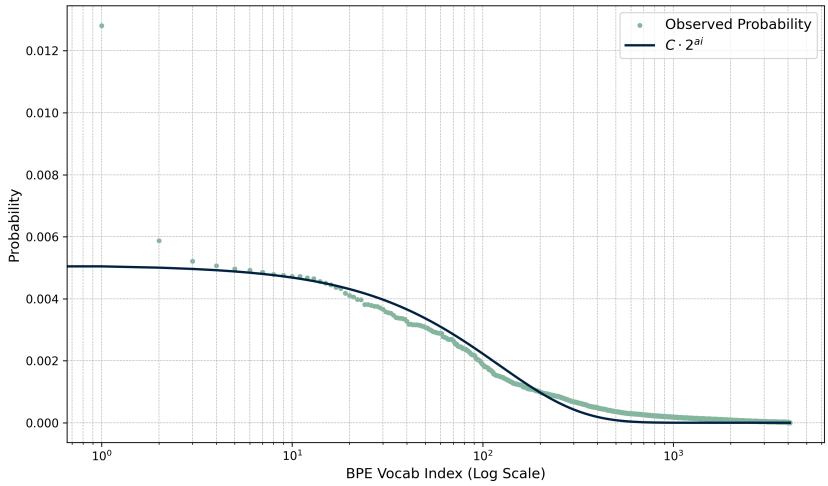

Figure 5: Fitting the exponential function $C \cdot 2^{-ai}$ to the empirically observed BPE token probabilities on pretraining datasets. Since the index assigned to a token is arbitrary, we reverse sort by probability/frequency and reindex tokens to represent the probability as a function of the index. We determined best-fit when $C \approx 0.005086$ and $a \approx 0.011909$.

Information content of a BPE token As before, we consider each BPE token in a sequence as an independent variable that carries some amount of information. Let the size of the vocabulary be N. On our pretraining datasets, we observe that the frequency of the BPE tokens is exponentially distributed, and as a result, the probability of a token can modeled by an exponential function.

$$P(x_i) = \frac{C}{2^{ai}} \tag{9}$$

Since the index assigned to a token is arbitrary, tokens can be sorted by descending probability and reindexed without issue. Under this formulation, the token with the index $i = 1$ is the most frequent (the GGG token in our pretraining dataset) and $i = 4095$ is the least frequent (the $TTGTCGGGTAAG$ token).

$$\begin{aligned}
&\sum_{i=1}^{N} \frac{C}{2^{ai}} = C \sum_{i=1}^{N} \frac{1}{2^{ai}} = 1 \\
&\Leftrightarrow \frac{C}{2^a} \cdot \frac{1 - \left(\frac{1}{2^a}\right)^N}{1 - \frac{1}{2^a}} = 1 \\
&\Leftrightarrow \frac{C\left(1 - \frac{1}{2^{aN}}\right)}{2^a - 1} = 1 \\
&\Leftrightarrow C = \frac{2^a - 1}{1 - \frac{1}{2^{aN}}}
\end{aligned} \tag{10}$$

When the vocabulary size N is large, we can approximate $C \approx 2^a - 1$ and $a \approx log_2(C + 1)$.

Now we can derive a general expression for the entropy of BPE tokens as,

$$
\begin{aligned}
H(X_{BPE}) &= -\sum_{i}^{N} P(x_i)\log_2 P(x_i) \\
&= -\sum_{i=1}^{N} \frac{C}{2^{ai}} \log_2\left(\frac{C}{2^{ai}}\right) \\
&= -\sum_{i=1}^{N} \frac{C}{2^{ai}} (\log_2 C - ai) \\
&= -\log_2 C \sum_{i=1}^{N} \frac{C}{2^{ai}} + aC \sum_{i=1}^{N} \frac{i}{2^{ai}}, \\
&= -\log_2 C + aC \sum_{i=1}^{N} \frac{i}{2^{ai}}, \\
&\approx -\log_2 C + aC \frac{2^a}{(2^a-1)^2}\ , \qquad \text{When } N \text{ is large} \\
&= -\log_2 C + \log_2(C+1) C \frac{C+1}{(C)^2} \\
&= -\log_2 C + \log_2(C+1)\left(\frac{C+1}{C}\right) \\
&= \log_2\left(\frac{(C+1)^{(C+1)/C}}{C}\right).
\end{aligned} \tag{11}
$$

If the weighted average length of a BPE token is $\bar{L} = \sum_{i=1}^{N} P(x_i)\text{len}(x_i)$, the average character-level entropy of BPE representation of a sequence will be $\hat{H}(X_{BPE}) = \frac{H(X_{BPE})}{\bar{L}}$. Since nucleotides are 1 character each, the per-character entropy is $\hat{H}(X_{NUC}) = H(X_{NUC})$. The BPE tokenization will lead to less entropy if,

$$
\begin{aligned}
&\frac{\hat{H}(X_{BPE})}{\hat{H}(X_{NUC})} < 1 \\
\Rightarrow\ &\frac{H(X_{BPE})}{\bar{L} \times H(X_{NUC})} < 1 \\
\Rightarrow\ &\log_2\left(\frac{(C+1)^{(C+1)/C}}{C}\right) < 2 \times \bar{L} \\
\Rightarrow\ &\frac{(C+1)^{(C+1)/C}}{C} < 4^{\bar{L}}.
\end{aligned} \tag{12}
$$

When $C << 1$, we can approximate $(C+1) \approx 1$. Then the inequality in Equation 12 can be further simplified as

$$\frac{1}{C} < 4^{\bar{L}} \Rightarrow C > 4^{-\bar{L}}.$$

Empirical Entropy Ratio On our pretraining data mixture, we determine that $P(A) \approx 0.2726$, $P(A) \approx 0.2144$, $P(A) \approx 0.26642$, $P(A) \approx 0.2465$, and averge BPE token length $\bar{L} \approx 6.0768$. This yields the empirical entropy of nucleotide tokens $H_e(X_{NUC}) \approx 1.9939$ bits. As shown in Figure 5, the **empirical value of C is 0.005086** when determined on 33 million sequences of our pretraining dataset.

Plugging in this value in Eqn. 11, yields $H_e(X_{BPE}) \approx 9.1044$ bits. Therefore, the empirical per-character entropy ratio is

$$\frac{\hat{H}_e(X_{BPE})}{\hat{H}_e(X_{NUC})} = \frac{H_e(X_{BPE})}{\bar{L} \times H_e(X_{NUC})} \approx \frac{9.1044}{6.0768 \times 1.9939} \approx 0.7514 < 1.$$

The empirical per-character entropy ratio of 0.7514 indicates that the Byte-Pair Encoding (BPE) tokenization technique effectively compresses the input sequence. Although compressed information is likely more difficult for Language Models to process, it is well-compensated by the ability to process sequences up to 6 times longer than the original input with the same GPU memory constraints. This also partially explains why we observed BPE underperforming their NUC counterparts on short-sequence downstream tasks from an Information-theoretic perspective.

Therefore, BPE tokenization is essentially a trade-off between information compression and computational efficiency, which BiRNA-BERT can dynamically adjust depending on the hardware constraints and sequence length.

We present a comparative performance analysis between BPE and NUC tokenization across two downstream binary classification tasks: **RNA-Protein interaction prediction** and **RNA N6-methyladenosine site prediction**. RNA-Protein interaction focuses on finding the binding sites and interactions between RNA molecules and proteins to understand post-transcriptional regulation. Benchmark dataset for RNA-protein interaction prediction is collected from RBPsuit available at http://www.csbio.sjtu.edu.cn/bioinf/RBPsuite/. This database contains datasets for 154 different proteins and a collection of interacting and non-interacting RNA sequences for each protein. We consider a subset of 5 datasets for our evaluation (AARS, AATF, AKAP1, AGGF1, ABCF1). The length of RNA sequences used for this task is 101 across all the datasets. N6-methyladenosine (m6A) is a common and critical modification in eukaryotic mRNA, affecting various aspects of RNA metabolism. This includes stability, splicing, and translation. The prediction and detection of m6A sites are essential for understanding how this modification influences gene expression and cellular processes. In our work, we utilized datasets from human, rat, and mouse tissues, specifically focusing on brain, kidney, and liver samples for each species. Data sets were derived from the iRNA-m6A study available at http://www.biolscience.cn/Deepm6A-MT/data/, employing an antibody-independent m6A-REF-seq protocol, which is both high-throughput and accurate for m6A detection. Positive samples were selected based on the presence of m6A at the center of 41 continuous nucleotide residues, while negative samples were randomly selected from the same tissues but without m6A sites.

In Table 6, we present results on RNA-Protein interaction prediction for BPE and NUC tokenization under two different configurations of BiRNA-BERT which we described in Section 3.4. NUC consistently outperforms BPE on all five concerned datasets by 3.56% (AARS), 2.91% (AATF), 4.15% (ABCF1), 2.35% (AGGF1), and 3.50% (AKAP1). BiRNA outperforms the current SOTA (BERTRBP [48]) by 3.49% (AARS), 2.89% (AATF), 7.43% (ABCF1), 3.84% (AGGF1), and 4.10% (AKAP1). BERTRBP is explicitly trained for predicting RNA-Protein binding prediction task finetuned on DNABERT [20]. Results for RNA N6-methyladenosine site prediction are presented in Table 7. Similar to the RNA-Protein interaction task, NUC versions perform consistently better than BPE versions. The average improvement of NUC over BPE for human, mouse and rat are 3.88%, 2.24%, and 1.68% across three different tissues: brain, kidney, and liver. Moreover, BiRNA outperforms the current SOTA (Deepm6A-MT [19]) by 0.33% (Human), 0.45% (Mouse), and 0.06% (Rat).

Table 6: F1 Score for RNA-Protein Interaction Prediction

Methods	Model	Datasets				
		AARS	**AATF**	**ABCF1**	**AGGF1**	**AKAP1**
BPE	BPE-Only	0.6792	0.7014	0.7266	0.6946	0.6778
	BiRNA-BPE	0.6769	0.6931	0.7296	0.6960	0.6835
NUC	NUC-only	0.6964	0.7217	**0.7599**	0.7101	0.7030
	BiRNA-NUC	**0.7034**	**0.7218**	0.7528	**0.7124**	**0.7074**
	BERT-RBP	0.6797	0.7016	0.7066	0.6861	0.6795

Table 7: Accuracy for RNA N6-methyladenosine Sites Prediction in Multiple Tissues Across Different Organisms

Species	**Human**			**Mouse**			**Rat**		
Tissue	**Brain**	**Kidney**	**Liver**	**Brain**	**Kidney**	**Liver**	**Brain**	**Kidney**	**Liver**
BPE-Only	0.710	0.778	0.786	0.773	0.792	0.708	0.764	0.817	0.815
BiRNA-BPE	0.714	0.786	0.795	0.779	0.794	0.709	0.769	0.817	0.818
NUC-Only	**0.753**	0.807	**0.823**	**0.802**	0.816	0.745	**0.788**	0.831	0.824
BiRNA-NUC	0.747	**0.811**	0.816	0.801	**0.820**	0.747	0.785	0.831	**0.825**
Deepm6A-MT	0.751	0.809	0.815	0.799	0.816	0.733	0.785	**0.840**	0.818

4.6 Beyond RNA Sequences: BiDNA-BERT

We have extended the idea of dual tokenization with AliBi encoding to DNA sequences too to verify the adaptibility of our proposed scheme to other biological language models beyond RNA. We compare our performance with DNABERT-2 [51], which is the current SOTA for DNA language modeling. BiDNA is trained only on human genome with 3 milion DNA sequences, whereas DNABERT-2 is trained on multi-species genome dataset encompasses genomes from 135 species, nearly 12 times the volume of the human genome dataset.We evaluate all three variants on human-genome-related downstream tasks from the GUE benchmark.

We see from Table 8 that, in promoter site detection task, BiRNA with NUC tokenization is comparable to DNABERT-2, within only 0.8% performance margin. In the core promoter site detection task, BiDNA-NUC outperforms DNABERT-2 with 4.4% MCC. In the transcription factor binding site task, BiDNA achieves a competitive performance within 1.2% margin. It is to be noted that BiDNA uses 66x less compute than DNABERT-2.

Table 8: Comparison of BiDNA with DNABERT-2 on Three Downstream Tasks (MCC Metric)

Task	**Promoter Site Detection**			**Core Promoter Site Detection**		
Dataset	**All**	**No tata**	**Tata**	**All**	**No tata**	**Tata**
BPE-Only	0.916	0.954	0.799	0.806	0.824	0.765
BiDNA-BPE	0.918	0.954	0.789	0.812	0.828	0.758
NUC-Only	0.927	0.963	0.807	**0.832**	0.836	**0.876**
BiDNA-NUC	0.936	0.966	0.817	**0.832**	0.839	0.844
DNABERT-2	**0.941**	**0.971**	**0.830**	0.831	**0.849**	0.814

Table 8: Comparison of BiDNA with DNABERT-2 on Three Downstream Tasks (MCC Metric) (Continued)

Model	**Transcription Factor Binding Dataset**					
	tf0	**tf1**	**tf2**	**tf3**	**tf4**	**Avg tf**
BPE-Only	0.847	0.852	0.807	0.738	0.847	0.818
BiDNA-BPE	0.847	0.848	0.803	0.753	0.852	0.821
NUC-Only	0.844	0.871	0.838	0.759	0.877	0.838
BiDNA-NUC	0.850	0.874	**0.843**	0.770	0.874	0.842
DNABERT-2	**0.856**	**0.886**	0.841	**0.790**	**0.888**	**0.852**

5 Conclusions

This study first empirically demonstrates that two popular tokenization approaches, NUC and BPE, each have their own advantages in RNA sequencing tasks: BPE, along with ALiBi positional encoding, enables transformer encoder models to process long biological sequences, while NUC enhances predictions for high-granularity tasks, yielding performance gains over shorter sequences. Given these observations, we propose a dual tokenization approach and show that pretraining on both NUC and BPE tokenizations of a sequence allows a single model to support both with no downsides compared to training on either alone. We release a new RNA foundational model, BiRNA-BERT, trained over our proposed tokenization approach, which achieves state-of-the-art results in long-sequence tasks and outperforms similar-sized models in short-sequence and nucleotide-level tasks. We also validate our methodology on DNA data and provide an information-theoretic analysis comparing NUC and BPE tokenization. We will release our code and models upon review decision. Our current approach reflects the simplest design of combining NUC and BPE. This naturally led to BPE tokens being underrepresented in the pretraining data: one BPE token corresponds to 6 nucleotides on average. The quality of embeddings generated with BPE tokens might improve by oversampling BPE tokens. BiRNA-BERT achieves state-of-the-art results in multiple downstream tasks despite only being trained for a single epoch, which is significantly shorter than other biological sequence models. We invite the research community to improve upon our findings with larger foundation models. We opted for ALiBi over competition methods such as RoPE [41] since ALiBi supports dynamically increasing the context window without retraining. Although BPE is prevalent in NLP, it has its drawbacks [2]. Exploring dual tokenization schemes with varied positional encodings and algorithms holds promise for future research.

References

[1] Christoph Adami. Information theory in molecular biology. *Physics of Life Reviews*, 1(1):3–22, 2004.

[2] Kaj Bostrom and Greg Durrett. Byte pair encoding is suboptimal for language model pretraining. In Trevor Cohn, Yulan He, and Yang Liu, editors, *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4617–4624, Online, November 2020. Association for Computational Linguistics.

[3] Nadav Brandes, Dan Ofer, Yam Peleg, Nadav Rappoport, and Michal Linial. Proteinbert: A universal deep-learning model of protein sequence and function. *bioRxiv*, 2021.

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[5] Bo Chen, Xingyi Cheng, Pan Li, Yangli-ao Geng, Jing Gong, Shen Li, Zhilei Bei, Xu Tan, Boyan Wang, Xin Zeng, et al. xtrimopglm: unified 100b-scale pre-trained transformer for deciphering the language of protein. *arXiv preprint arXiv:2401.06199*, 2024.

[6] 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.

[7] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*, 2020.

[8] The RNAcentral Consortium. RNAcentral: a hub of information for non-coding RNA sequences. *Nucleic Acids Research*, 47(D1):D221–D229, 11 2018.

[9] Fiona Cunningham, James E Allen, Jamie Allen, Jorge Alvarez-Jarreta, M Ridwan Amode, Irina M Armean, Olanrewaju Austine-Orimoloye, Andrey G Azov, If Barnes, Ruth Bennett, et al. Ensembl 2022. *Nucleic acids research*, 50(D1):D988–D995, 2022.

[10] Hugo Dalla-Torre, Liam Gonzalez, Javier Mendoza-Revilla, Nicolas Lopez Carranza, Adam Henryk Grzywaczewski, Francesco Oteri, Christian Dallago, Evan Trop, Bernardo P de Almeida, Hassan Sirelkhatim, et al. The nucleotide transformer: Building and evaluating robust foundation models for human genomics. *bioRxiv*, pages 2023–01, 2023.

[11] Hugo Dalla-Torre, Liam Gonzalez, Javier Mendoza Revilla, Nicolas Lopez Carranza, Adam Henryk Grzywaczewski, Francesco Oteri, Christian Dallago, Evan Trop, Hassan Sirelkhatim, Guillaume Richard, Marcin Skwark, Karim Beguir, Marie Lopez, and Thomas Pierrot. The nucleotide transformer: Building and evaluating robust foundation models for human genomics. *bioRxiv*, 2023.

[12] Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*, 2023.

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[14] Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, et al. Prottrans: Toward understanding the language of life through self-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):7112–7127, 2021.

[15] Noelia Ferruz, Steffen Schmidt, and Birte Höcker. Protgpt2 is a deep unsupervised language model for protein design. *Nature Communications*, 13, 2022.

[16] Philip Gage. A new algorithm for data compression. *The C Users Journal archive*, 12:23–38, 1994.

[17] Jonas Geiping and Tom Goldstein. Cramming: Training a language model on a single gpu in one day. In *International Conference on Machine Learning*, pages 11117–11143. PMLR, 2023.

[18] Sam Griffiths-Jones, Alex Bateman, Mhairi Marshall, Ajay Khanna, and Sean R Eddy. Rfam: an rna family database. *Nucleic acids research*, 31(1):439–441, 2003.

[19] Guohua Huang, Xiaohong Huang, and Jinyun Jiang. Deepm6a-mt: A deep learning-based method for identifying rna n6-methyladenosine sites in multiple tissues. *Methods*, 226:1–8, 2024.

[20] Yanrong Ji, Zhihan Zhou, Han Liu, and Ramana V Davuluri. DNABERT: pre-trained Bidirectional Encoder Representations from Transformers model for DNA-language in genome. *Bioinformatics*, 37(15):2112–2120, 02 2021.

[21] John M. Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A A Kohl, Andy Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with alphafold. *Nature*, 596:583 – 589, 2021.

[22] Qiang Kang, Jun Meng, Jun Cui, Yushi Luan, and Ming Chen. PmliPred: a method based on hybrid model and fuzzy decision for plant miRNA–lncRNA interaction prediction. *Bioinformatics*, 36(10):2986–2992, 02 2020.

[23] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Sal Candido, and Alexander Rives. Language models of protein sequences at the scale of evolution enable accurate structure prediction. *bioRxiv*, 2022.

[24] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[25] Ali Madani, Ben Krause, Eric R. Greene, Subu Subramanian, Benjamin P. Mohr, James M. Holton, Jose Luis Olmos, Caiming Xiong, Zachary Z Sun, Richard Socher, James S. Fraser, and Nikhil Vijay Naik. Large language models generate functional protein sequences across diverse families. *Nature Biotechnology*, pages 1–8, 2023.

[26] Eric Nguyen, Michael Poli, Marjan Faizi, Armin Thomas, Michael Wornow, Callum Birch-Sykes, Stefano Massaroli, Aman Patel, Clayton Rabideau, Yoshua Bengio, et al. Hyenadna: Long-range genomic sequence modeling at single nucleotide resolution. *Advances in neural information processing systems*, 36, 2024.

[27] Rafael Josip Penić, Tin Vlašić, Roland G Huber, Yue Wan, and Mile Šikić. Rinalmo: General-purpose rna language models can generalize well on structure prediction tasks. *arXiv preprint arXiv:2403.00043*, 2024.

[28] Jacob Portes, Alexander R Trott, Sam Havens, Daniel King, Abhinav Venigalla, Moin Nadeem, Nikhil Sardana, Daya Khudia, and Jonathan Frankle. Mosaicbert: How to train bert with a lunch money budget. In *Workshop on Efficient Systems for Foundation Models@ ICML2023*, 2023.

[29] Ofir Press, Noah A Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. *arXiv preprint arXiv:2108.12409*, 2021.

[30] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

[31] Roshan Rao, Nicholas Bhattacharya, Neil Thomas, Yan Duan, Peter Chen, John Canny, Pieter Abbeel, and Yun Song. Evaluating protein transfer learning with tape. *Advances in neural information processing systems*, 32, 2019.

[32] Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C. Lawrence Zitnick, Jerry Ma, and Rob Fergus. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15):e2016239118, 2021.

[33] Nicolas Scalzitti, Arnaud Kress, Romain Orhand, Thomas Weber, Luc Moulinier, Anne Jeannin-Girardon, Pierre Collet, Olivier Poch, and Julie D Thompson. Spliceator: multi-species splice site prediction using convolutional neural networks. *BMC bioinformatics*, 22:1–26, 2021.

[34] Armin O Schmitt and Hanspeter Herzel. Estimating the entropy of dna sequences. *Journal of theoretical biology*, 188(3):369–377, 1997.

[35] Conrad L Schoch, Stacy Ciufo, Mikhail Domrachev, Carol L Hotton, Sivakumar Kannan, Rogneda Khovanskaya, Detlef Leipe, Richard Mcveigh, Kathleen O'Neill, Barbara Robbertse, et al. Ncbi taxonomy: a comprehensive update on curation, resources and tools. *Database*, 2020:baaa062, 2020.

[36] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*, 2015.

[37] Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

[38] Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*, 2019.

[39] Jaswinder Singh, Kuldip Paliwal, Jaspreet Singh, and Yaoqi Zhou. Rna backbone torsion and pseudotorsion angle prediction using dilated convolutional neural networks. *Journal of Chemical Information and Modeling*, 61(6):2610–2622, 2021.

[40] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[41] Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding, 2023.

[42] Mostofa Rafid Uddin, Sazan Mahbub, M Saifur Rahman, and Md Shamsuzzoha Bayzid. Saint: self-attention augmented inception-inside-inception network improves protein secondary structure prediction. *Bioinformatics*, 36(17):4599–4608, 2020.

[43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[44] Susana Vinga. Information theory applications for biological sequence analysis. *Briefings in bioinformatics*, 15(3):376–389, 2014.

[45] Xi Wang, Ruichu Gu, Zhiyuan Chen, Yongge Li, Xiaohong Ji, Guolin Ke, and Han Wen. Uni-rna: Universal pre-trained models revolutionize rna research. *bioRxiv*, 2023.

[46] Yunxia Wang, Ziqi Pan, Minjie Mou, Weiqi Xia, Hongning Zhang, Hanyu Zhang, Jin Liu, Lingyan Zheng, Yongchao Luo, Hanqi Zheng, et al. A task-specific encoding algorithm for rnas and rna-associated interactions based on convolutional autoencoder. *Nucleic Acids Research*, 51(21):e110–e110, 2023.

[47] Ruidong Wu, Fan Ding, Rui Wang, Rui Shen, Xiwen Zhang, Shitong Luo, Chenpeng Su, Zuofan Wu, Qi Xie, Bonnie Berger, Jianzhu Ma, and Jian Peng. High-resolution de novo structure prediction from primary sequence. *bioRxiv*, 2022.

[48] Keisuke Yamada and Michiaki Hamada. Prediction of rna–protein interactions using a nucleotide language model. *Bioinformatics Advances*, 2(1):vbac023, 2022.

[49] Tongchuan Zhang, Jaswinder Singh, Thomas Litfin, Jian Zhan, Kuldip Paliwal, and Yaoqi Zhou. RNAcmap: a fully automatic pipeline for predicting contact maps of RNAs by evolutionary coupling analysis. *Bioinformatics*, 37(20):3494–3500, 05 2021.

[50] Yikun Zhang, Mei Lang, Jiuhong Jiang, Zhiqiang Gao, Fan Xu, Thomas Litfin, Ke Chen, Jaswinder Singh, Xiansong Huang, Guoli Song, Yonghong Tian, Jian Zhan, Jie Chen, and Yaoqi Zhou. Multiple sequence alignment-based RNA language model and its application to structural inference. *Nucleic Acids Research*, 52(1):e3–e3, 11 2023.

[51] Zhihan Zhou, Yanrong Ji, Weijian Li, Pratik Dutta, Ramana Davuluri, and Han Liu. Dnabert-2: Efficient foundation model and benchmark for multi-species genome. *arXiv preprint arXiv:2306.15006*, 2023.