BiRNA-BERT: Adaptive Tokenization for Efficient RNA Language Modeling

Md Toki Tahmid[*]
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology
sharifulislamtoki@gmail.com

Haz Sameen Shahgir[*]
Computer Science and Engineering
University of California, Riverside
sameen2080@gmail.com

Sazan Mahbub
Department of Computer Science
University of Maryland, College Park, MD 20742, USA
smahbub@cs.umd.edu

Yue Dong
Computer Science and Engineering
University of California, Riverside
yue.dong@ucr.edu

Md. Shamsuzzoha Bayzid
Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology
shams.bayzid@gmail.com

Abstract

Recent advancements in Transformer-based language 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, distance map prediction or secondary structure prediction. To tackle these challenges, we propose an adaptive dual tokenimzation 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 28 billion nucleotides across 36 million coding and non-coding RNA sequences. The learned representation by BiRNA-BERT generalizes across a range of applications. The BiRNA-BERT model achieves state-of-the-art results in long-sequence downstream tasks, performs comparably well in short-sequence tasks, and matches the performance in nucleotide-level structural prediction tasks, of models six times larger in parameter size, while requiring 27 times 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. We further demonstrate the applicability of the dual-pretraning and adaptive tokenization strategy employing this concept on a DNA language model which provides comparable performance to 66X compute heavy DNA language models. BiRNA-BERT can dynamically adjust its tokenization strategy based on sequence lengths, utilizing NUC for shorter sequences and switching to BPE for longer ones, thereby offering,

[*] These authors contributed equally. Corresponding author: sharifulislamtoki@gmail.com

Foundation Models for Science Workshop,38th Conference on Neural Information Processing Systems (NeurIPS 2024).

for the first time, the capability to efficiently handle arbitrarily long DNA/RNA sequences. [2]

1 Introduction

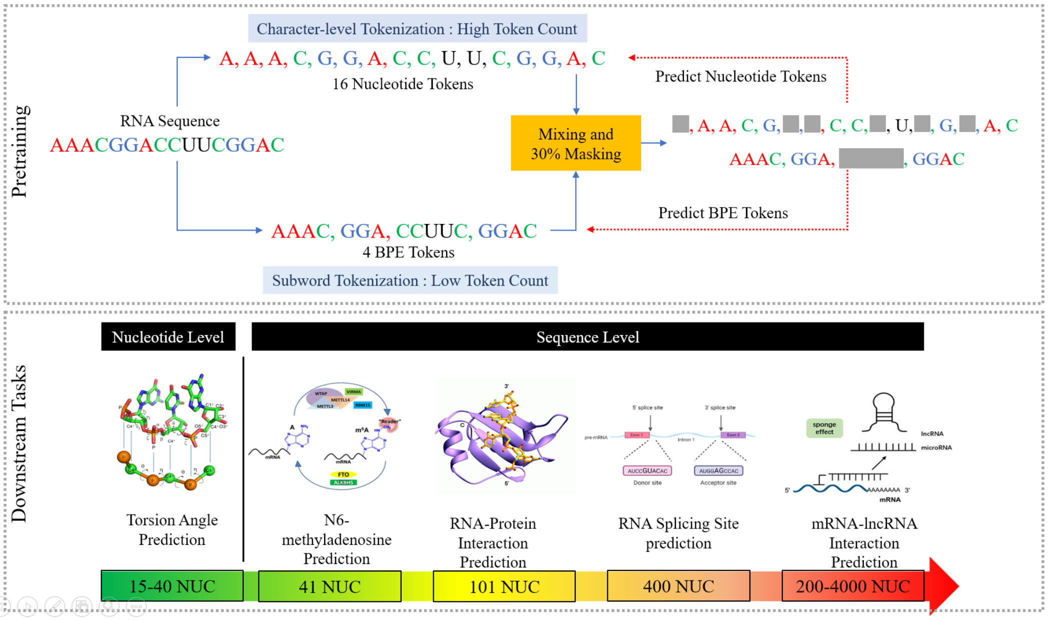


Figure 1: Overview of BiRNA, which processes a given RNA sequence through nucleotide and BPE tokenization. The resulting tokens, which vary in tokenization precision, are mixed and utilized to pretrain the masked language model. A number of downstream tasks are conducted to validate BiRNA's performance under different conditions.

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 [4, 12]. This success has inspired researchers to apply the pretraining-finetuning paradigm to a wide range of topics beyond NLP, including protein language models [11], biological sequence modeling[8, 25, 2, 13, 5], and single cell language models[1] with remarkable success . 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 long sequences for certain tasks, as biological sequences can range from tens to millions of nucleotides or amino acids [17]. The standard transformer encoder architecture (e.g., BERT [4]) uses positional embeddings that limit the maximum input length to usually 512 or 1024, requiring truncation or inefficient workarounds for biological sequences (truncate at 1024 length for ProtTrans [5], and 5120 in EMS-2 [11]). Longer sequence lengths also result in the loss of attention a transformer architecture with fixed positional encoding, like cosine positional embeddings. The limited number of approaches to adopting new positional encoding for larger context window handling and improved pretraining developed in NLP often fall short in biological sequencing. For example, Nucleotide Transformers (NT) [3] 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 [25], where they demonstrate the poor sample efficiency of k-mer tokenization and propose adapting *BPE tokenization (concatenating statistically significant nucleotides into a single token)* for DNA sequences. However, in domains like RNA or protein sequences where residue level tasks (per residue/nucleotide embedding) 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.

[2] The code and model weights are available at `https://github.com/buetnlpbio/BiRNA-BERT`

Thus the challenges with biological language models are threefold. First, handling the longer sequences without the loss of attention. It is essential to make sure that even if during the training we train the model with short sequences, the model can extrapolate for much longer sequences during a downstream inference task. Secondly, it is preferable to avoid truncation of the sequences which is commonly done in most of the biological language models to fit into the available computing memory. Thus we need to make sure we use utilize the complete sequence length and in the same time make the computation efficient to be fitted in available computing resources. Finally, it is essential to ensure that any compression technique such as k-mer compression of statistical compression of tokens with BPE tokenization do not sacrifice the capability to generate token-wise embedding when necessary for residue/nucleotide level tasks such as structural properties prediction.

Given the above-mentioned 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 the miRNA-lncRNA interaction dataset [23]. Along with sequence-level tasks, the dual tokenization approach allows us to conduct nucleotide-level analyses, such as RNA 3D torsion angle prediction [19], 3D distance map prediction [21], secondary structure prediction [9] , as additional concurrent downstream tasks.

Our main contributions can therefore be summarized as follows:

1. We present an effective approach - dual tokenization pretraining - that extends the effective context window of biological foundation models with efficient tokenization while retaining the ability to generate character-level embeddings. Shorter sequences are tokenized at nucleotide level (each nucleotide is considered one single token) and for longer sequences BPE tokenization is used (compressing statistically significant nucleotides into a single token) instead of truncation.

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 the available computing resource. BiRNA-BERT can also achieves very comparable performance for various downstream tasks with structural prediction such as 3D torsion angle prediction, 3D distance map prediction , and secondary structure prediction.

3. Through information theoretic analysis we demonstrate the information loss that occur during BPE tokenization compared to nuclotide level tokenization. To validate this result, we also conduct an empirical study on several tasks to show that NUC outperforms BPE on tasks where sequences are short enough to fit into GPU memory.

4. The training and adaptive tokenization approach that BiRNA-BERT proposes can be seamlessly integrated to any other biological language modeling tasks. Do demonstrate that, we also propose BiDNA-BERT which is a significantly smaller DNA language model compared to DNABERT-2, however provides comparable performance with magnitude times larger and compute heavy models.

2 Methods

In this section, we first describe the applicability and methodology of using BPE tokenization for BiRNA-BERT (Subsection 2.1). Then we describe the motivation behind using relative positional encoding (ALiBi) for extrapolating the trained model to longer sequences in downstream tasks (Section 2.2). In 2.3, we discuss the dual tokenization pretraining approach and its motivation. Finally, we describe the pretraining configurations and datasets used in Section 2.4. The overall training approach of BiRNA-BERT is shown in Figure 1.

2.1 Byte-Pair Encoding

Byte-Pair Encoding (BPE) [6, 18] 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 $4^k + 5$ [24], 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.

2.2 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 [22] or learned embeddings [4], 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. Popular algorithms for relative positional embeddings are T5 Bias [16], Rotary Positional Embedding (RoPE) [20], and Attention with Linear Biases (ALiBi) [15]. Recently, RiNALMo [13] 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.

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.

2.3 Dual Pretraining and Adaptive Tokenization

To address the limitations of BPE for shorter sequences and nucleotide-specific tasks, we introduce a dual pretraining and adaptive tokenization strategy. During pretraining, each RNA sequence is tokenized using both nucleotide-level (NUC) and Byte Pair Encoding (BPE). The model is trained simultaneously on both token types, with 30% of tokens masked in each set, allowing it to learn from both granular and compressed representations. During inference, we adopt an adaptive tokenization approach: BPE is used for longer sequences to enhance memory efficiency, while NUC is applied to shorter sequences for more accurate nucleotide-specific processing. This combined strategy enables the model to handle diverse sequence lengths effectively, improving generalization and computational efficiency. These two phases are described in Algorithm 1, and 2.

We train a BPE tokenizer on the whole collection of RNAcentral database with around 36 million non-coding RNA sequences and 530K mRNA sequences from RefSeq database. With a maximum vocabulary size of 4096, we find the statistically most frequent RNA subsequences which we refer as the BPE vocabulary. This collection of BPE tokens follows an exponential distribution which is described in detail in the Appendix/Supplementary material. On our pretraining data mixture, we determine that $P(A) \approx 0.2726$, $P(U) \approx 0.2144$, $P(C) \approx 0.26642$, and $P(G) \approx 0.2465$. With this nucleotide level tokens distribution, we analyze the information loss during BPE compression compared to nucleotide level tokenization.

Performing BPE tokenization on long sequences effectively compresses the sequence with an 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

Algorithm 1 Pretraining Phase

```
Require: Set of RNA sequences $\{seq_1, \ldots, seq_n\}$, where $n \approx 36$ million
Ensure: Pretrained Model
for each $seq_i$ in $\{seq_1, \ldots, seq_n\}$ do
    Generate NUC Tokens: $tokens_{NUC} \leftarrow$ NUC Tokenization$(seq_i)$    ▷ Tokenize at the nucleotide level
    Generate BPE Tokens: $tokens_{BPE} \leftarrow$ BPE Tokenization$(seq_i)$    ▷ Apply BPE tokenization
    Simultaneous Masking:
    Randomly mask 30% of $tokens_{NUC}$ and 30% of $tokens_{BPE}$
    $\mathbf{r}_{NUC}^{mask} \leftarrow$ Masked NUC tokens
    $\mathbf{r}_{BPE}^{mask} \leftarrow$ Masked BPE tokens
    Compute Losses:
    $\mathcal{L}_{NUC} \leftarrow -\sum_{i \in \mathcal{M}_{NUC}} \log P(\nu_i | \mathbf{r}_{NUC}^{mask})$    ▷ NUC loss
    $\mathcal{L}_{BPE} \leftarrow -\sum_{i \in \mathcal{M}_{BPE}} \log P(\beta_i | \mathbf{r}_{BPE}^{mask})$    ▷ BPE loss
    Total Loss:
        $\mathcal{L}_{total} = \mathcal{L}_{NUC} + \mathcal{L}_{BPE}$
    Optimize Parameters:
    Update model parameters $\Theta \leftarrow \Theta - \eta \nabla \mathcal{L}_{total}$    ▷ Optimize with gradient descent
end for
return Pretrained Model
```

Algorithm 2 Inference Phase

```
Require: RNA Sequence $seq$, Max Token Limit $MAX\_TOKENS$, Trade-off Hyperparameter $k$
Ensure: Tokenized Sequence
if len$(seq) > k \times MAX\_TOKENS$ then
    $tokens \leftarrow$ BPE Tokenization$(seq)$    ▷ Use BPE for long sequences
else
    $tokens \leftarrow$ NUC Tokenization$(seq)$    ▷ Use NUC for short sequences
end if
return $tokens$    ▷ Tokenized sequence for inference
```

tokenization. Therefore, BPE tokenization is essentially a trade-off between information compression and computational efficiency. 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.

2.4 Pretraining Configuration and Dataset

We pretrain the BiRNA model, which simultaneously uses both Byte Pair Encoding (BPE) and nucleotide-level (NUC) tokenizations. This configuration processes 32.254 billion tokens over 48.42 hours of training. The model is trained using MosaicML [14] with a learning rate of 2×10^{-4}, a warmup ratio of 0.06, and a batch size of 200 per device across eight Nvidia RTX 3090 GPUs.

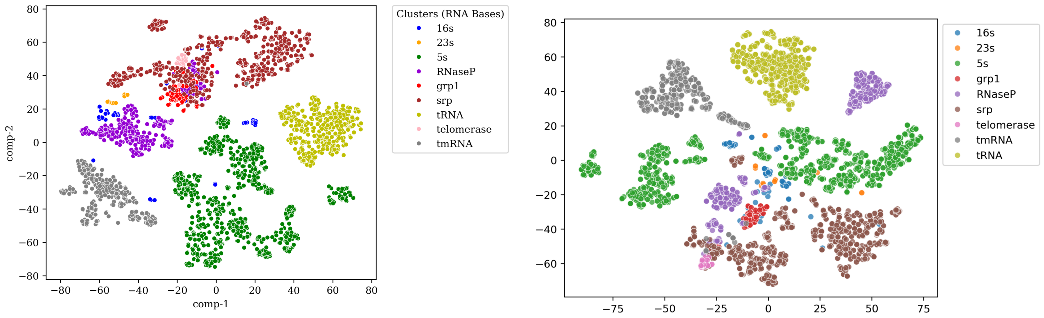
The pretraining datasets include both non-coding RNA (ncRNA) and coding RNA (mRNA) sequences. ncRNA data were sourced from RNAcentral, comprising 36 million sequences and 26.42 billion nucleotides. mRNA sequences were obtained from RefSeq, totaling 532,852 sequences and 2.22 billion nucleotides. This comprehensive dataset enables the model to capture the functional diversity of RNA sequences across various biological contexts.

3 Results

This section highlights the results from pretraining BiRNA-BERT and its performance on various downstream tasks. Section 3.1 covers the results of the base pretrained model and unsupervised species clustering. In Section 3.2, we demonstrate that BiRNA sets new state-of-the-art results on RNA-RNA interaction by leveraging dynamic tokenization. In Section 3.3, we demonstrate that nucleotide embeddings from BiRNA have identical performance to a BERT trained solely on nucleotides. We cover other short sequence downstream tasks in Section 3.4 to establish that dual tokenization training has no drawbacks over conventional training. BiRNA-BERT even significantly surpasses similar-sized models such as RNA-FM and BERTRBP on short-sequence tasks.

3.1 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 unsupervised clustering on 9 RNA structural families: 16s, 23s, 5s, RNaseP, grp1, srp, tRNA, telomerase, tmRNA. Upon extracting the embeddings with BPE tokenization we perform a TSNE dimensionality reduction and plot the 2D embedding space in Figure 2. Comparing the clustering performance with Rinalmo, we can visualize that different families are more distinctly clustered with BiRNA-BERT. This comparison indicates the better expressive capacity of BiRNA-BERT with BPE tokenization.



(a) BiRNA-BERT-BPE (Ours)

(b) RINALMo

Figure 2: Comparison of unsupervised RNA structural family classification.

3.2 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 [10] and CORAIN [23]. 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 **(author?)** [10]. This is a binary classification task to determine whether a miRNA-lncRNA pair interacts or not. 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 1 offer several interesting insights: 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. 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. 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 (See Appendix). In miRNA-lncRNA interaction tasks, we demonstrate that compression

Table 1: Accuracy of different models on miRNA-lncRNA dataset. CORAIN [23] 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**

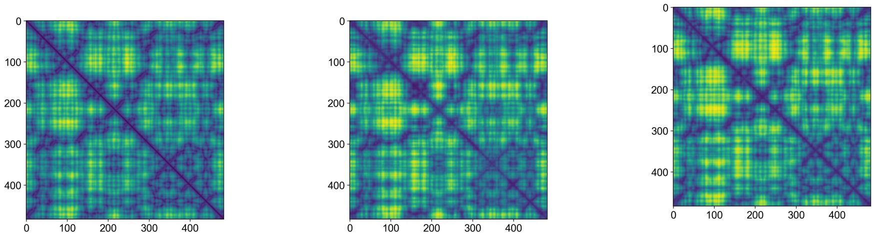
Table 2: Performance analysis of RNA structural properties prediction by BiRNA-BERT, RNA-FM, and RiNALMO. *BiRNA-BERT is 6X smaller and uses 27X less compute compared to RINALMo.*

Method	**3D Torsion Angle (Mean Absolute Error)**				**3D Distance Map (R2 Score)**	**Secondary Structure (F1 Score)**	
	VL	**TS1**	**TS2**	**TS3**	**Validation**	**Validation**	**TS0**
BiRNA-BERT	*28.085*	**28.181**	*26.704*	*31.979*	**0.82**	*0.694*	*0.700*
RNA-FM	28.333	29.916	27.710	32.000	0.71	0.657	0.685
RINALMo	**27.888**	*28.622*	**25.915**	**31.513**	*0.81*	**0.712**	**0.701**

(BPE) is preferable to explicit information loss (NUC), while also being more computationally efficient (807 tokens versus 1024 tokens).

3.3 Nucleotide-Level Task: Structural Properties Prediction

Along with sequence-level tasks described in the previous sections, BiRNA-BERT can simultaneously be applied to different nucleotide level tasks where the embedding information per nucleotide is required. To verify BiRNA-BERT's capability of such granular-level task, we investigate the performance of BiRNA-BERT on three structural tasks for RNA: RNA 3d torsion angle prediction, RNA 3d distance map prediction, and RNA secondary structure prediction. Detail description of the datasets along with the training and testing data are provided in the Appendix. In Table 2, we compare the performance of BiRNA-BERT with RiNALMo and RNA-FM. All the models are trained for 30 epochs with full fine-tuning. On the 3d torsion angle prediction task, out of the three datasets, BiRNA-BERT outperforms RiNALMo in one dataset and achieves second best performance on the other two datasets even being a 27X smaller model compared to RiNALMo. BiRNA-BERT significantly outperforms similar sized model RNA-FM on all the datasets. In the 3d distance map prediction task, on the independent validation dataset, BiRNA-BERT outperforms RiNALMo and RNA-FM in terms of R^2 score with an score of 0.82. In the secondary structure prediction dataset,

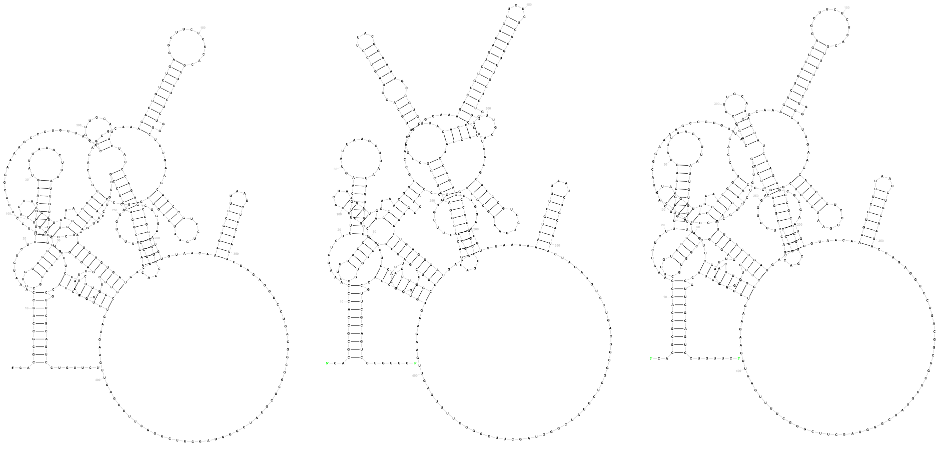


True 3D Distance Map (C1' - C1' Distance)

Predicted Distance Map by RiNALMo (R^2 Score = 0.915)

Predicted Distance Map by BiRNA-BERT (R^2 Score = 0.917)

Figure 3: Visualization of RNA 3d distance map prediction for the longest sequence in the testing dataset by RiNALMo and BiRNA-BERT.



True Secondary Structure

Secondary Structure Predicted by RiNALMo (F1 score = 0.92)

Secondary Structure Predicted by BiRNA-BERT (F1 score = 0.98)

Figure 4: Visualization of RNA secondary structure prediction task for the longest sequence in the testing dataset by RiNALMo and BiRNA-BERT.

on the independent test dataset (TS0), BiRNA-BERT and RiNALMo perform equally well compared to RNA-FM. In Figure 3, we show the prediction performance of BiRNA-BERT and RiNALMo on the longest RNA sequence in the test dataset. We see that, RiNALMo and BiRNA-BERT both provide very accurate prediction with R^2 score of 0.915 and 0.917 by RiNALMo and BiRNA-BERT accordingly. Similarly, in Figure 4, similar visualization is performed for RNA secondary structure prediction task for the longest sequence in the TS0 dataset. BiRNA-BERT provides an F1 score of 0.98 whereas RiNAMLo achieves 0.92. These results demonstrate that, BiRNA-BERT can effectively handle long sequence prediction tasks too. Particularly for the nucleotide level prediction tasks described in this section, BiRNA-BERT performs equally as good as 27X times compute heavy RiNAMLo model.

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

Table 3: 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

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 sequences in this case explicitly. We consider the task of binary classification of RNA splicing site prediction specifically for acceptor sites. We compare our model with RiNALMo, RNA-FM, and non-LM-based SOTA approaches in Table 3. 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.

3.5 Information Theoretic Analysis of Nuelcotide vs BPE Tokenization with Empirical Validation

We compare the information content of nucleotide and BPE tokens to assess their efficiency in representing RNA sequences. Nucleotide tokens have a per-token entropy upper bound of $H(X_{NUC}) = 2$

bits, assuming a uniform distribution. In contrast, BPE tokens, modeled with exponentially distributed probabilities $P(x_i) = \frac{C}{2^{ai}}$, have an entropy of $H(X_{BPE}) \approx \log_2\left(\frac{(C+1)^{(C+1)/C}}{C}\right)$. The character-level entropy for BPE is $\hat{H}(X_{BPE}) = \frac{H(X_{BPE})}{\bar{L}}$, with BPE being more efficient when $\frac{\hat{H}(X_{BPE})}{\hat{H}(X_{NUC})} < 1$. Empirically, $H_e(X_{NUC}) \approx 1.9939$ bits and $H_e(X_{BPE}) \approx 9.1044$ bits, with an average BPE token length $\bar{L} \approx 6.0768$, yielding a per-character entropy ratio of ≈ 0.7514. While BPE provides compression, it may lose some information, explaining its lower performance on short sequences, despite enabling models like BiRNA-BERT to handle longer inputs within the same computational limits. Full derivations are provided in the Appendix.

Advantages of NUC over BPE for Short Sequences: To validate the results of the information theoretical analysis discussed above, we compare the performance of BPE and NUC tokenization on short-sequence binary classification tasks (Table 4)) *RNA-Protein interaction prediction* and ii) *RNA N6-methyladenosine site prediction*. For RNA-Protein interaction, we use datasets from RBPsuit (AARS, AATF, AKAP1, AGGF1, ABCF1), each with sequence lengths of 101. Results in Table 4 show that BiRNA-NUC consistently outperforms BiRNA-BERT, with improvements ranging from 2.24% (AATF) to 4.30% (ABCF1). BiRNA-NUC also surpasses the current SOTA (BERTRBP [24]) by up to 7.43% (ABCF1).

For RNA N6-methyladenosine site prediction, BiRNA-NUC again outperforms BiRNA-BERT across human and mouse with average improvements of 3.88%, 2.24%, and 1.68%, respectively. BiRNA-NUC also outperforms Deepm6A-MT [7], with gains of 0.33% (Human), 0.45% (Mouse), and 0.06% (Rat). Overall, NUC tokenization offers better performance on short-sequence tasks due to less information loss, consistent with our information-theoretic analysis.

Table 4: F1 Score for RNA-Protein Interaction Prediction and Accuracy for RNA N6-Methyladenosine Sites Prediction across different models and datasets.

Model	**RNA-Protein Interaction**			**RNA N6-Methyladenosine**					
	AARS	**AATF**	**ABCF1**	**Human**			**Mouse**		
				Brain	**Kidney**	**Liver**	**Brain**	**Kidney**	**Liver**
BiRNA-BERT	0.6792	0.7014	0.7296	0.714	0.786	0.795	0.779	0.794	0.709
BiRNA-NUC	**0.7034**	**0.7218**	**0.7599**	**0.753**	**0.811**	**0.823**	**0.802**	**0.820**	**0.747**
BERT-RBP	0.6797	0.7016	0.7066	—	—	—	—	—	—
Deepm6A-MT	—	—	—	0.751	0.809	0.815	0.799	0.816	0.733

4 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.

References

[1] Haiyang Bian, Yixin Chen, Xiaomin Dong, Chen Li, Minsheng Hao, Sijie Chen, Jinyi Hu, Maosong Sun, Lei Wei, and Xuegong Zhang. scmulan: a multitask generative pre-trained language model for single-cell analysis. In *International Conference on Research in Computational Molecular Biology*, pages 479–482. Springer, 2024.

[2] 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.

[3] 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.

[4] 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.

[5] 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.

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

[7] 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.

[8] 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.

[9] Marek Justyna, Maciej Antczak, and Marta Szachniuk. Machine learning for rna 2d structure prediction benchmarked on experimental data. *Briefings in Bioinformatics*, 24(3):bbad153, 2023.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] 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.

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

[19] 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.

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

[21] Saisai Sun, Wenkai Wang, Zhenling Peng, and Jianyi Yang. Rna inter-nucleotide 3d closeness prediction by deep residual neural networks. *Bioinformatics*, 37(8):1093–1098, 2021.

[22] 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.

[23] 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.

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

[25] 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.