LncRNA-BERT: An RNA Language Model for Classifying Coding and Long Non-Coding RNA

Luuk Romeijn, Davy Cats, Katherine Wolstencroft, Hailiang Mei *†‡§

January 17, 2025

Abstract

Understanding (novel) RNA transcripts generated in next generation sequencing experiments requires accurate classification, given the increasing evidence that long non-coding RNAs (lncRNAs) play crucial regulatory roles. Recent developments in Large Language Models present opportunities for classifying RNA coding potential with sequence-based algorithms that can overcome the limitations of classical approaches that assess coding potential based on a set of predefined features. We present lncRNA-BERT, an RNA language model pre-trained and fine-tuned on human RNAs collected from the GENCODE, RefSeq, and NONCODE databases to classify lncRNAs. LncRNA-BERT matches and outperforms state-of-the-art classifiers on three test datasets, including the cross-species RNAChallenge benchmark. The pre-trained lncRNA-BERT model distinguishes coding from long non-coding RNA without supervised learning which confirms that coding potential is a sequence-intrinsic characteristic. LncRNA-BERT has been shown to benefit from pre-training on human data from GENCODE, RefSeq, and NONCODE, improving upon configurations pre-trained on the commonly used RNAcentral dataset. In addition, we propose a novel Convolutional Sequence Encoding method that is shown to be more effective and efficient than K-mer Tokenization and Byte Pair Encoding for training with long RNA sequences that are otherwise above the common context window size. lncRNA-BERT is available at `https://github.com/luukromeijn/lncRNA-Py`.

Index terms— coding potential, deep learning, large language model, long non-coding RNA, nucleotide language model, transformer

1 Introduction

Owing to advancements in RNA library extraction and sequencing technologies, bulk RNA and single cell RNA sequencing have been regularly applied in research projects and routine clinical diagnostics. Bioinformatics analysis of these RNAseq datasets leverages the well-maintained annotations from gene annotation databases, such as RefSeq and Ensembl, to examine the expression profiles of genes or transcripts. While earlier research has focused on protein coding transcripts, more recent work has investigated non-coding RNAs (ncRNAs) for their important regulatory functions [1–3]. Non-coding RNAs are often classified into two large groups based on their transcript length: short ncRNAs (including biotypes such as miRNA, piRNA, snoRNA, and tRNA) and long non-coding RNAs (lncRNAs) with a length $\geq$200 nt. Specific databases, such as NONCODE and RNAcentral, have been implemented to track newly reported ncRNAs across different species [4, 5]. In fact, lncRNAs are a highly prevalent type of RNA, underlined by the 173,112 human lncRNA transcripts stored in the NONCODE (v6) database versus the 197,151 mRNAs in RefSeq (r225).

To maintain high-quality curation, it is important to classify novel transcripts into a proper biotype and, in particular, distinguish lncRNAs from protein-coding RNAs (pcRNA) as they often share a similar transcript length. Currently, researchers can choose between more than 40 algorithms to classify whether a novel transcript sequence is a lncRNA.

*Luuk Romeijn is with Leiden Institute of Advanced Computer Science and Department of Biomedical Data Sciences, Leiden University Medical Center (e-mail: l.romeijn@umail.leidenuniv.nl)

†Davy Cats is with Department of Biomedical Data Sciences, Leiden University Medical Center (e-mail: D.Cats@lumc.nl)

‡Katherine Wolstencroft is with Leiden Institute of Advanced Computer Science, and Amsterdam University Medical Center (e-mail: k.j.wolstencroft@amsterdamumc.nl)

§Hailiang Mei is with Department of Biomedical Data Sciences, Leiden University Medical Center (e-mail: H.Mei@lumc.nl)

Most of these classifiers are machine learning models that have been trained on annotated RNA data from RefSeq or GENCODE (e.g. CPC, CNCI, CPAT, and CPC2 [6–9]), extracting features such as ORF length, protein database alignment hits, and k-mer frequencies as predictors for coding potential. The challenge of the classification problem lies in its inherent ambiguity: some lncRNAs contain short ORFs that translate into small peptide chains [10], and pcRNA genes can have non-coding isoforms [11,12]. This has stimulated the continuous development of novel classifiers, leveraging the latest available annotations and methodological advancements, such as feature selection algorithms [13,14], boosting models [15], and neural networks [16–18].

Although feature-based algorithms can learn the relationship between the target and a predetermined set of predictors, they may fail to capture the true underlying signal. Purely sequence-based deep learning methods for lncRNA classification have been proposed [17, 19, 20], but these were outperformed by feature-based or hybrid methods in a recent benchmark [21]. These approaches utilize convolutional and/or recurrent neural networks (CNN/RNN), which suffer from limited receptive fields and exploding/vanishing gradients, respectively. The transformer architecture improves upon these designs by employing an attention mechanism and is nonrecurrent [22].

We propose using a transformer-based Nucleotide Language Model (NLM) to overcome the limitations of previous classifiers. NLMs are Large Language Models (LLMs) that have been pre-trained and fine-tuned on genomic data. DNA NLMs have demonstrated their potential to predict chromatin features, identify promoter regions, and perform variant prioritization [23–25]. RNA NLMs have been shown to be capable of predicting splice sites, secondary structures, and interactions of RNA [26–30]. Despite being presented for general purposes, previous RNA language models may not be properly adapted for detecting coding potential due to utilizing RNAcentral (which excludes pcRNA) as the main data source and having a limited context length (< 1024 nt).

In this work, we present a new lncRNA classification method called lncRNA-BERT (long non-coding RNA Bidirectional Encoder Representations from Transfomers), an RNA language model specifically adapted to classifying RNA as coding or long non-coding. The main contributions of our study are threefold. First, lncRNA-BERT obtained state-of-the-art classification performance on three different test sets and significantly outperformed all other methods in the most difficult cross-species RNAChallenge test set. Second, lncRNA-BERT accurately distinguished between pcRNA and lncRNA without access to target labels by pre-training on human RNA from GENCODE, RefSeq, and NONCODE, rather than multi-species ncRNA from RNAcentral. Third, an in-depth comparison of four sequence encoding methods demonstrates the effectiveness of a novel Convolutional Sequence Encoding (CSE) method for pre-training on long RNA sequences and identifies CSE and 3-mer tokenization as the most suitable methods for classifying lncRNAs.

2 Methods

lncRNA-BERT was implemented in the Python package lncRNA-Py, which is available from GitHub (`https://github.com/luukromeijn/lncRNA-Py`). Additional experiments, such as architecture and hyperparameter tuning, have also been reported.

2.1 Data

An overview of the datasets utilized for the different tasks is presented in Table 1. The main pre-training dataset is a human set of 297,724 coding and 238,470 non-coding RNA sequences from GENCODE (v46) [31], NONCODE (v6) [4], and RefSeq (v255) [32]. A randomly selected 5% of the GENCODE sequences was held out for validation. We experimented with RNAcentral (v24) [5], which contains 37 million ncRNAs, as an alternative multi-species pre-training data source and kept aside 2,500 sequences for validating this model.

For fine-tuning, we used the CD-HIT algorithm [33] to ensure non-redundancy and test set independence, similar to [13, 15]. We ran the CD-HIT algorithm with a 90% sequence identity threshold on the combined human pcRNA/lncRNA data from GENCODE and RefSeq and randomly selected 90% of representative sequences for training, 5% for validation, and 5% for testing. The fine-tuned set contains 101,270 protein-coding and 48,785 non-coding RNAs. NONCODE was deliberately excluded during fine-tuning to maximize the data reliability.

Three test sets were used to assess the performance of lncRNA-BERT and the other classifiers: GENCODE/RefSeq, CPAT, and RNAChallenge. The GENCODE/RefSeq test set (5,650 pcRNAs, 2,686 lncRNAs) is guaranteed to not overlap with our fine-tuning data because of the above-described redundancy

Task	Name	# pcRNA	# ncRNA	Length
Pre-train	Human	297,724	238,470	2,835
Validation		5,583	2,998	2,037
Pre-train	RNAcentral	0	37,942,367	659
Validation		0	2,500	747
Fine-tune	GENCODE / RefSeq	101,270	48,785	2,558
Validation		5,650	2,686	2,581
Test		5,634	2,703	2,601
	CPAT	4,000	4,000	1,958
	RNAChallenge	16,243	11,040	1,635

Table 1: Data specification. Human data from GENCODE, RefSeq, and NONCODE is used. The three GENCODE/RefSeq datasets are obtained by first clustering the data with CD-HIT (90% identity threshold), and then randomly selecting 90%, 5%, and 5% for fine-tuning, validation, and testing, respectively. Average length is reported.

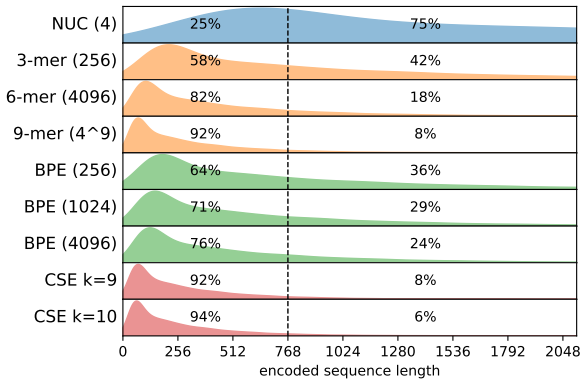

Figure 1: Density plots of encoded sequence lengths from the human pre-training dataset (536,194 sequences) when using NUC, K-mer Tokenization, BPE, and CSE. Vocabulary sizes are provided in parentheses. The percentages represent the number of encoded sequences that fall within a medium-sized context length of 768 input positions.

removal with CD-HIT. The CPAT set (4,000 pcRNAs, 4,000 lncRNAs) is a published and widely used test set [8], although some overlap with the training sets of each of the evaluated classifiers is expected. Finally, we used another published benchmark, RNAChallenge [21], containing 27,283 hard-to-classify RNA sequences, to assess the generalizability of our model to ambiguous RNA sequences from animal, plant, and fungi species.

2.2 Encoding Methods

We compared four methods for converting nucleotide sequences into a numerical format: Nucleotide-Level Tokenization (NUC), K-mer Tokenization (K-mer), Byte Pair Encoding (BPE), and a novel Convolutional Sequence Encoding (CSE) method. Efficient encoding is required for long sequences, as the transformer's attention mechanism only accepts a limited number of input positions (512-1024 on standard GPUs) due to its quadratic memory complexity. The effects of different encoding methods on the encoded sequence length are illustrated in Fig.1.

2.2.1 Nucleotide-Level Tokenization

Most RNA NLMs apply NUC as the sequence encoding method using a vocabulary of four nucleotide tokens (A,C,T,G). This technique allows attention to nucleotide resolution and works well for short sequences, such as most RNAs in RNAcentral. However, NUC is insufficient for long RNAs in our human pre-training set, as only 25% would fit into a medium-sized context length of 768 when NUC-encoded (Fig.1).

2.2.2 K-mer Tokenization

With a vocabulary comprised of all nucleotide combinations of length k, K-mer Tokenization tokenizes the input as non-overlapping consecutive k-mers [25]. This reduces the sequence length by a factor of k and yields a vocabulary of size 4^k. For a large k, the exponentially large vocabulary size introduces token sampling efficiency problems [23] and results in an explosion of parameters in the transformer's embedding layer. This complicates the training and may not reflect the actual complexity of the data, as two k-mers are learned independently even when they are highly similar. Moreover, the model must learn and recognize each data signal in k alternative reading frames, taking up parameters/dimensions that would preferably be dedicated to other patterns.

2.2.3 Byte Pair Encoding

BPE considers the most frequently occurring combinations of characters as tokens [34] and was first applied to nucleotide sequences in [23]. During training, BPE iteratively expands its vocabulary by merging the most frequent token pairs in the input corpus. This process is repeated until a prespecified vocabulary size (vs) is reached. During tokenization, BPE merges the subwords in the same order. BPE yields a highly efficient, fixed-size vocabulary of variable-length tokens. This method has three advantages over K-mer Tokenization: 1) a larger sequence length reduction for the same vocabulary size (Fig.1); 2) a higher token sampling efficiency; and 3) a better robustness against frameshifts. BPE has been shown to improve upon K-mer Tokenization in multiple NLM applications [23, 35], but still requires a large vocabulary to obtain strong compression [24].

2.2.4 Convolutional Sequence Encoding

With CSE, nucleotide sequences are directly embedded into a high-dimensional space by means of a convolution (Fig.2). First, we convert an input sequence of length l to a $4 \times l$ Position Weight Matrix (PWM), for example $\mathrm{A} = [1, 0, 0, 0]^\top$ and $\mathrm{N} = [0.25, 0.25, 0.25, 0.25]^\top$. A one-dimensional convolution layer (ReLU activated) with four input channels then transforms the PWM into d_{model} dimensions for the transformer using d_{model} learnable kernels. The stride is set to kernel size k, such that the sequence length is reduced k times, analogous to K-mer Tokenization. We add zero-padding to allow mini-batch training, and mask these positions during the attention operation.

CSE effectively reduces the embedded sequence length with an efficient number of trainable parameters while maintaining nucleotide-level resolution. The method is inspired by the Vision Transformer [36], a similar approach for short DNA sequences was taken in [37]. Through the use of convolutions, CSE extracts important patterns from the data, seeing the input positions as combinations of nucleotides instead of independent tokens.

2.3 Language Model Architecture

We adapted the transformer encoder architecture as used by the $\mathrm{BERT}_{\mathrm{medium}}$ [22, 38], with $N = 12$ transformer blocks, a dimensionality of $d_{model} = 768$, $d_{ff} = 3072$ nodes in the feed-forward layers, and $h = 12$ attention heads. lncRNA-BERT uses a medium-sized context length of $c = 768$ input positions. The transformed embedding of the `CLS` token was used as input to the lncRNA classification output head, which is a sigmoid-activated linear layer containing a single node. The model has 85M trainable parameters.

Our CSE-based models use a modified version of BERT (Fig.2). For classification, we replaced the `CLS` token with a learnable `CLS` embedding, as in ViT [36]. For the Masked Language Modeling (MLM) pre-training task, we enable nucleotide-level predictions by applying a softmax-activated, transposed convolution with stride/kernel size k on linearly transformed embeddings.

2.4 Training

We pre-train lncRNA-BERT for 7 days on an A100 MIG 4g.40GB GPU in the ALICE compute resources provided by Leiden University, after identifying an optimal model configuration based on pre-training, fine-tuning, and probing. The optimal model checkpoints were stored based on the validation set performance.

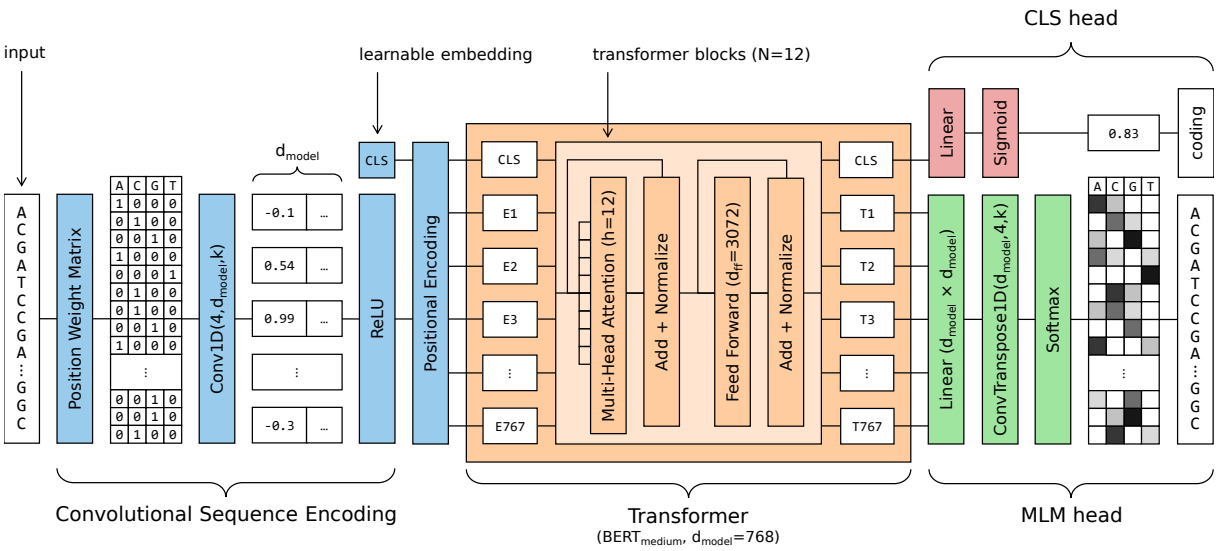

Figure 2: Architecture of lncRNA-BERT with Convolutional Sequence Encoding (CSE), adopting the $\text{BERT}_{\text{medium}}$ transformer architecture [38]. CSE embeds a nucleotide sequence into d_{model} dimensions by means of a 1D convolution on its Position Weight Matrix, using d_{model} learnable kernels of size k. Fine-tuning tasks such as lncRNA classification are performed using an output head connected to the transformed `CLS` embedding. A dedicated MLM output head performs a transposed convolution, which enables masking and prediction at nucleotide resolution.

2.4.1 Pre-training

MLM was used as a pre-training task, where tokens or nucleotides were selected with a probability of $p_{mlm} = 0.15$, of which 80% is masked and 10% is randomly replaced. The model's task is to predict which tokens or nucleotides occur at the selected positions. With CSE, we use the IUPAC symbol 'N' to mask out nucleotides, as it indicates an equal probability for any of the four bases. The model is pre-trained using a cross entropy loss function, a batch size of 8, the Adam optimizer, and the learning rate schedule proposed in [22], with 32,000 warmup steps.

2.4.2 Fine-tuning

LncRNA-BERT was fine-tuned for coding potential classification on 101,270 coding and 48,785 long non-coding RNAs from GENCODE and RefSeq. We optimized 100 epochs of 10,000 samples using Adam, a fixed learning rate of 10^{-5}, and a batch size of 8. A binary cross entropy loss function was used, with the reciprocal class sizes as weights to counteract the class imbalance.

2.4.3 Probing

Probing is used to evaluate the extent to which a pre-trained model encodes the coding potential in its sequence embeddings, without fine-tuning the weights of the network itself [25]. Hereto, we trained a small Multi-Layer Perceptron with a single hidden layer of 256 nodes on the mean-pooled output embeddings of a model. Optimization follows the same settings as during fine-tuning, except for an increased learning rate of 10^{-4}.

2.5 Experimental Setup

We compared lncRNA-BERT to six published lncRNA classification algorithms. These include three classical approaches (CPAT [8], LncFinder [14], PredLnc-GFStack [13]), and three deep learning methods (feature-based LncADeep [16], sequence-based mRNN [17], and hybrid model RNAsamba [18]). LncADeep, RNAsamba, and mRNN are the best-performing, publicly available, human models in a recent benchmark [21]. Out-of-the-box models were used without re-training.

We experimented with two pre-training datasets (human, RNAcentral) and four encoding methods (NUC, K-mer, BPE, CSE) to determine an optimal lncRNA-BERT configuration. These comparisons

were based on 500 pre-training epochs. The macro-averaged F1-score was used as the main performance metric, balancing precision and recall while equally weighing pcRNA/ncRNA to address class imbalance.

3 Results

3.1 LncRNA-BERT Outperforms and Matches State-Of-The-Art Performance in lncRNA Classification and Demonstrates a Better Generalizability

We selected 3-mer tokenization and CSE with $k = 9$ as the optimal encoding methods (Section 3.3), and benchmarked these configurations of lncRNA-BERT against six existing methods. As illustrated in Table 2, lncRNA-BERT matched state-of-the-art lncRNA classifiers in the two test sets and significantly outperformed all other methods in the most difficult RNAChallenge test set. This result proves the applicability of NLM for distinguishing coding from long non-coding RNA. In particular, we improved upon a previous sequence-based model (mRNN) by using a more advanced architecture and training procedure (BERT instead of RNN).

Our BERT-based lncRNA classification method did not result in a large performance gain in comparison to the existing methods for two human test sets. Previous studies have set a high standard for lncRNA classification and a performance plateau is likely within reach (Section 4). In particular, for the CPAT test set, all methods performed almost equally well with a macro F1-score between 0.950 and 0.970 (lncRNA-BERT's score is 0.963). The fact that the CPAT test set has been a well-known set for many years and contains more distinguishable pcRNAs and lncRNAs also makes true performance distinction difficult. Furthermore, some methods also benefit from sequence-extrinsic features, which cannot be learned by NLMs. For example, LncADeep performs alignments against a protein reference database to which lncRNA-BERT does not have access to. Nevertheless, we showed that coding potential is mostly a sequence-intrinsic characteristic, as lncRNA-BERT already distinguishes between pcRNA and lncRNA after self-supervised pre-training (Section 3.2).

Overfitting is a difficult problem in machine learning, which can cause some models to perform well on a particular test set but fail to generalize to unseen data. This potential bias in our analysis is mostly caused by the similarity between the training and test sets, which we mitigated for lncRNA-BERT by generating an independent train/test split after redundancy removal with CD-HIT (Section 2). The consistent performance of lncRNA-BERT on an independent test set shows proper generalization within human data competitive with established methods. The performance on the cross-species RNAChallenge benchmark also showed improvements in generalization across species compared to established methods.

3.2 Pre-Training on Human pcRNA/lncRNA Captures Coding Potential Without Access to Target Labels

The MLM pre-training task enabled our model to differentiate coding from long non-coding RNA without depending on target labels, given an appropriate pre-training dataset, as seen in the embedding spaces in Fig.3A. This affirms that the coding potential is a prominent sequence-intrinsic signal. Although lncRNA-BERT is not the first method to solely base its predictions on sequence patterns [7, 17, 39], it is the first to be capable of discriminating between pcRNA and lncRNA in a fully self-supervised manner. Fig.3B indicates that pre-training on human data leads to faster convergence and higher F1-scores (0.01-0.08 performance gain) compared to RNAcentral or no pre-training. This shows that pre-training on a proper dataset benefits classification.

Pre-training on RNAcentral biases the model towards ncRNA types other than human pcRNA and lncRNA. This is reflected in the per-sequence MLM accuracy (% of correctly predicted tokens) in Fig.3C, which shows that the RNAcentral model performs well on sequences from RNAcentral (mean: 70%), but does not achieve the same accuracy on human lncRNAs (mean: 15%). The same can be concluded from the embedding spaces generated by both models. The RNAcentral model embeds pcRNA and ncRNA less distinctly than the human model. We also generated embedding spaces for the ArchiveII dataset, containing multi-species ncRNAs from 10 structural families [40]. Here, the RNAcentral model successfully separates the different classes, in contrast to the human model. Hence, the patterns learned from RNAcentral do not generalize well to human pcRNA/lncRNA, and vice versa. This may be caused by the abundance of specific ncRNAs in RNAcentral, as well as the complete lack of pcRNA. Moreover, lncRNAs are less well-conserved across evolution [12], complicating the detection of coding potential in a cross-species dataset.

Test set	Metrics	lncRNA-BERT		CPAT	LncFinder	PredLnc	LncADeep	RNAsamba	mRNN
		K-mer (k=3)	CSE (k=9)						
GENCODE/ RefSeq	F1 (macro)	0.940	**0.943**	0.889	0.866	0.941	0.930	0.902	0.907
	Precision (pcRNA)	0.965	0.957	0.943	0.962	0.956	**0.966**	0.947	0.949
	Recall (pcRNA)	0.958	**0.971**	0.910	0.851	0.969	0.942	0.924	0.929
	Precision (ncRNA)	0.912	**0.937**	0.824	0.748	0.934	0.884	0.848	0.858
	Recall (ncRNA)	0.926	0.907	0.884	**0.930**	0.906	0.929	0.892	0.895
CPAT	F1 (macro)	0.963	0.947	0.962	0.959	0.950	**0.970**	0.958	0.950
	Precision (pcRNA)	0.942	0.914	0.948	**0.959**	0.917	0.958	0.933	0.937
	Recall (pcRNA)	0.987	0.988	0.977	0.960	**0.990**	0.983	0.988	0.965
	Precision (ncRNA)	0.986	0.987	0.977	0.960	**0.989**	0.983	0.987	0.963
	Recall (ncRNA)	0.940	0.906	0.947	**0.959**	0.910	0.957	0.928	0.935
RNAChallenge	F1 (macro)	0.235	**0.242**	0.031	0.005	0.173	0.146	0.138	0.155
	Precision (pcRNA)	0.397	**0.412**	0.069	0.006	0.337	0.288	0.278	0.295
	Recall (pcRNA)	0.420	**0.453**	0.050	0.004	0.344	0.272	0.259	0.278
	Precision (ncRNA)	**0.066**	0.056	0.003	0.005	0.005	0.012	0.008	0.022
	Recall (ncRNA)	**0.060**	0.048	0.004	0.007	0.005	0.013	0.008	0.024

Table 2: Comparison of lncRNA classifiers in 3 test sets

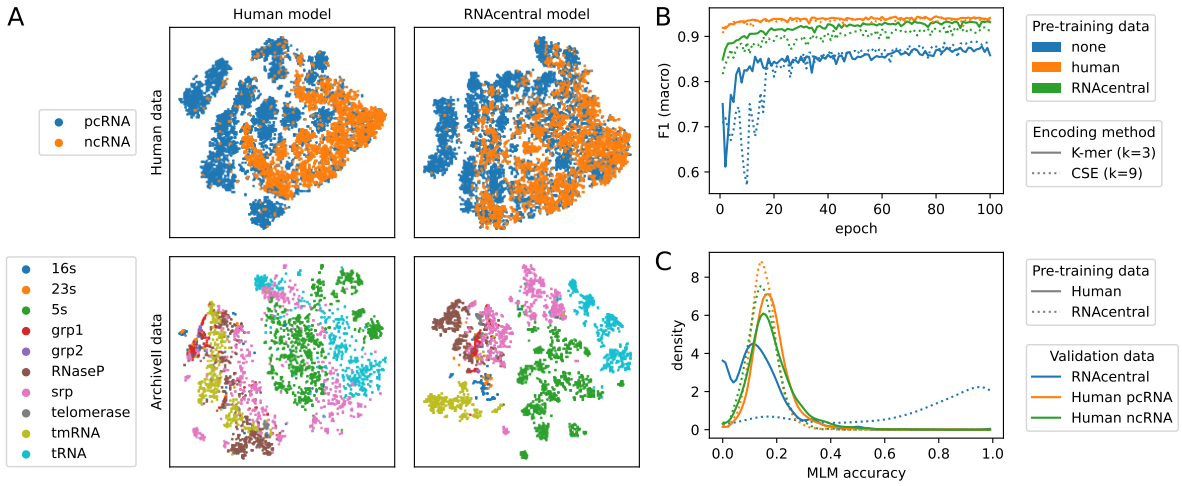

Figure 3: The effect of pre-training data on lncRNA-BERT. A: T-SNE visualizations of the embedding spaces of the human validation set and the ArchiveII dataset after pre-training lncRNA-BERT with 3-mer tokenization on human data or on multi-species data from RNAcentral. The human model better distinguishes pcRNA from ncRNA than the RNAcentral model, while the latter generates an improved separation between structural families in ArchiveII. B: Learning curve of the macro-averaged F1-score during 100 training epochs on the lncRNA classification task, for different pre-training configurations and encoding methods. The models benefit from pre-training, specifically on human data, leading to faster convergence and increased performance. C: Density plot of the MLM accuracy per sequence (% correctly predicted tokens) for lncRNA-BERT with 3-mer tokenization when pre-trained/evaluated on human pcRNA/ncRNA data or cross-species ncRNA from RNAcentral. The RNAcentral model achieves a high MLM accuracy (mean: 70%) on the RNAcentral validation set but performs worse than the human model on lncRNA from the human validation set (mean: 15% versus 18%). The models slightly favor pcRNA over ncRNA.

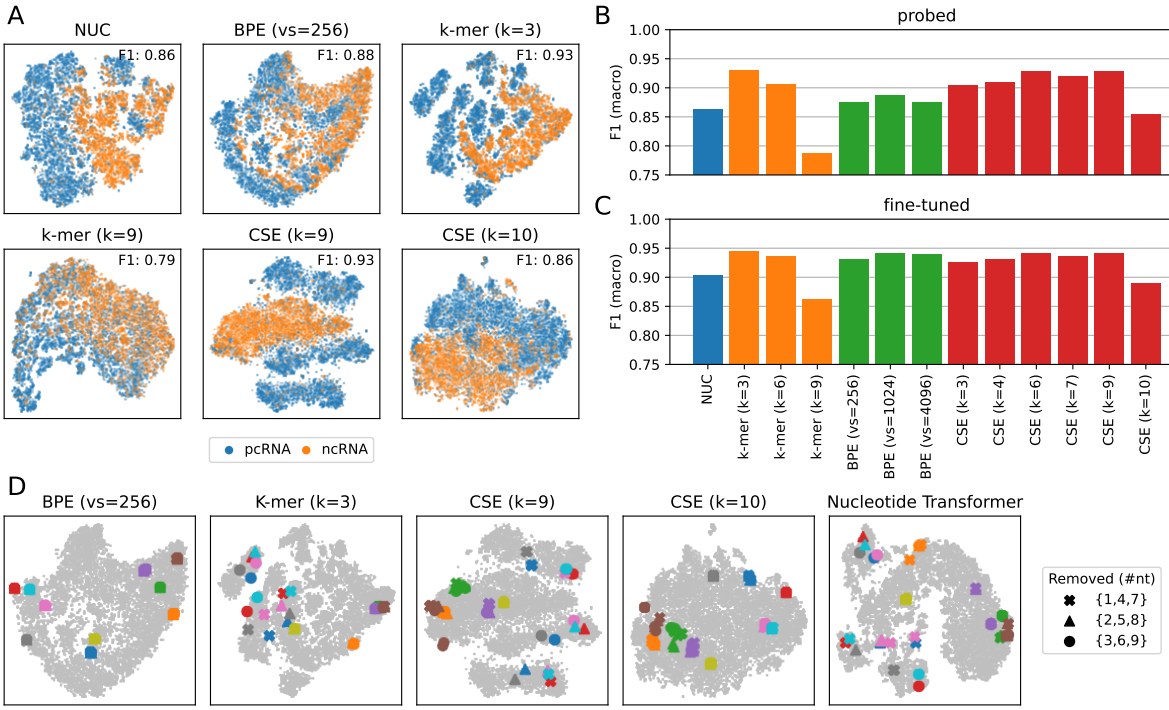

Figure 4: Embedding spaces and performance of lncRNA-BERT for different encoding methods. Mean pooling is used to retrieve sequence-level embeddings. A: T-SNE visualizations of the embedding spaces of the validation set after pre-training models with different encoding methods. K-mer Tokenization with $k = 3$ and CSE with $k = 9$ lead to the best distinction between pcRNA and lncRNA, reflected in the achieved F1-score when probed (0.93). CSE improves upon K-mer for $k = 9$. B-C: Macro-averaged F1-scores on the lncRNA classification task for different encoding methods, using probing (B) or fine-tuning (C). K-mer Tokenization with $k = 3$ leads to the highest performance for both probing (0.93) and fine-tuning (0.94). D: T-SNE visualizations of the embeddings of models with different encoding methods, which change depending on the reading frame of the input. Assessed by removing up to 9 nucleotides from the start of 10 randomly selected sequences (indicated by color) in the validation set. The sequence-level embedding does not change for BPE, and CSE when $k = 10$. Sequence embeddings jump across three different coordinates when using K-mer Tokenization or CSE with k divisible by 3.

Our results reveal a potential weakness of existing RNA language models, which use RNAcentral as the main data source, making them less suitable for pcRNA/lncRNA classification.

3.3 Convolutional Sequence Encoding Improves Pre-Training on Longer RNA Sequences

We pre-trained, probed, and fine-tuned BERT models with different encoding methods (Fig.4), and identified CSE ($k = 9$) and 3-mer tokenization as the most suitable for coding potential classification.

Of all encoding methods that achieve a large sequence length reduction ($\geq$ 6 times), CSE leads to the most effective models. This is reflected in the embedding spaces (Fig.4A) as well as in the obtained F1-scores after probing (Fig.4B). Specifically, CSE and K-mer Tokenization obtain a probing F1-score of 0.93 versus 0.91 for $k = 6$, and 0.93 versus 0.79 for $k = 9$, respectively, on the validation dataset. CSE's fine-tuning scores are also higher for these configurations (Fig.3C). The reduced performance of K-mer Tokenization can be explained by the limitations of using a large token vocabulary (e.g. 4^9 for 9-mers), which include reduced sampling efficiency [23] and a high embedding layer parameter count (e.g., $4^9 \times d_{model}$ for 9-mers). By contrast, CSE sees k-mers as combinations of nucleotides and maintains a fixed number of parameters ($4 \times d_{model} \times k$) by learning important patterns from the data. BPE encodes sequences into fully independent tokens, such as the k-mer approach, contributing to why it is outperformed by CSE for $vs \in [256, 1024, 4096]$ when probed.

Although superior for long sequences, CSE is outperformed by K-mer and BPE when the token size is small and fine-tuning is allowed. In Fig.4C, CSE with $k = 3$ results in a fine-tuning F1-score of 0.93,

while the 3-mer and BPE for $vs = 256$ and $vs = 1024$, achieve 0.94, 0.93, and 0.94 respectively. Shorter tokens enable attention at a higher resolution, thereby improving contextualized embeddings. Seeing input positions as predefined, independent tokens also helps the model discriminate between sequences. In comparison, CSE must learn important patterns before BERT can condition on them, making the model less stable than when using a tokenizer. Issues specific to tokenizers, such as sampling efficiency and parameter count blow-ups, are less prevalent for smaller vocabularies. This specifically applies to fine-tuning, during which the model can prioritize important tokens over less important tokens. Consequently, BPE and K-mer tokenization show a larger performance gain from probing to fine-tuning compared to CSE.

Another factor that affects classification performance is the sequence length coverage of each encoding method (Fig.1). A longer coverage allows the model to consider a larger part of the sequence in its predictions, explaining why 3-mer tokenization is superior to NUC, as shown in Fig.4. An extended context length may also hinder the model from learning important local signals such as CSE $k = 10$. The context length of a model with 3-mer tokenization ($3 \times 768 = 2304$) was shown to be sufficient for classifying coding potential. The biological implication here is that the coding potential of long RNA transcripts is usually inferrable from the first 2304 nucleotides. This, in combination with its small vocabulary, explains why a model with 3-mer tokenization achieves a similar fine-tuning performance to CSE with $k = 9$ (0.944 versus 0.941), despite having a smaller context length.

3.4 Three-Base Periodicity in Sequence Encoding Method Benefits Performance and Affects Embedding Space

Encoding methods that align with the three-base periodicity of coding RNA have been shown to better distinguish pcRNAs from lncRNAs, owing to their sensitivity to biological reading frames. For example, CSE with $k \in [6, 9]$ improves upon $k \in [7, 10]$. However, there is also a negative effect of the reading frame sensitivity of the three-base periodic encoding methods as illustrated in Fig.4D, in which the same pcRNA sequence could jump between the three groups when observed in different reading frames. While input position embeddings are reading frame dependent (particularly for K-mer and CSE), the aggregated (mean) embedding at the sequence level should remain the same, as the sequence's meaning is preserved. Fig.4D shows that this applies to BPE or CSE with $k = 10$, but not for K-mer or CSE with k as a multiple of 3. Moreover, the figure shows that the Nucleotide Transformer, which uses 6-mer tokenization, suffers from the same issue. This effect was not observed in the non-coding sequences.

These observations demonstrate how biological reading frames in the data affect the behavior of three-base periodic models, which appear to favor in-frame signals. These signals are easier to learn, because their periodic organization aligns with the input positions of the model. For example, from 'ACT TGA ACT' it is easier to learn that 'TGA' follows 'ACT', compared to learning that 'GAA' follows 'CTT'. We expect the presence/absence of such easily recognizable signals to cause the observed shifts between the embeddings in Fig.4D. Three-base periodicity is less evident in lncRNA, causing a lower MLM accuracy (Fig.3) but a better embedding consistency than pcRNA. The frameshift sensitivity is mitigated by setting k to a value not divisible by 3, which breaks up the three-base periodicity.

3.5 Embedding Spaces of pcRNA/lncRNA Data Reveal Differences Between NLMs

We compared lncRNA-BERT to previously released NLMs by visualizing the embedding spaces of the validation set in Fig.5. lncRNA-BERT is pre-trained on human pcRNA and lncRNA, in contrast to other RNA NLMs, which use RNAcentral as the main data source. Excluded from Fig.5 are ERNIE-RNA, RNABERT, RNAErnie, and RNA-FM [26–29]. These methods were only trained on non-coding RNAcentral data, making them less suitable for inference on pcRNAs and lncRNAs (Fig.3). BiRNA-BERT and RiNALMo also utilize the RNAcentral dataset, but augment it with data from RefSeq [41] and Rfam/Ensembl [30]. Nevertheless, the embedding space of BiRNA-BERT did not separate pcRNA from ncRNA as clearly as lncRNA-BERT. Only the RiNALMo model (650M parameters) generated a clearer distinction, but was 7.6× larger than lncRNA-BERT (85M parameters) and was pre-trained with a limited context length (< 1024 nt).

Fig.5 shows the competitiveness of lncRNA-BERT with other NLMs on this dataset and task, while also being the most parameter-efficient. Interestingly, DNA NLMs such as DNABERT-2 and Nucleotide Transformer (v2) show some potential for RNA classification. This indicates that DNA models may be generalized to RNA tasks, although not as well as to RNA-specific models.

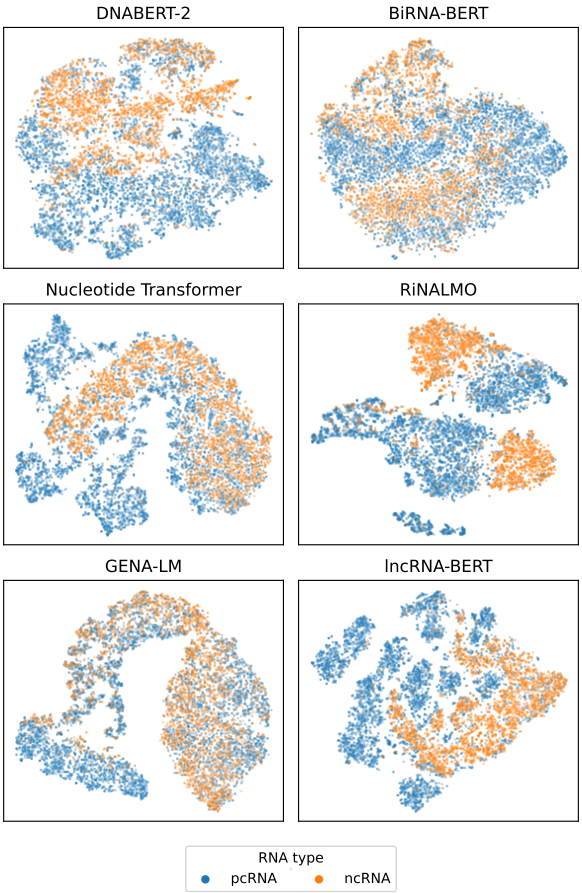

Figure 5: T-SNE visualizations of the embedding spaces of the validation set, generated by different NLMs. RiNALMo generates the clearest distinction between coding and non-coding transcripts, followed by lncRNA-BERT and DNABERT-2. BiRNA-BERT, Nucleotide Transformer (v2, multi-species), and GENA-LM (base, T2T) differentiate the two types of RNA less clearly.

4 Discussion

The human RNA language model proposed in this work, lncRNA-BERT, demonstrates state-of-the-art performance in classifying RNAs as coding or long non-coding (Table2). Pre-training on human pcRNA/lncRNA from GENCODE, RefSeq, and NONCODE leads to a clear distinction between the two classes, showing that an NLM can learn coding potential from sequence data without relying on target labels (Fig.3A). LncRNA-BERT stands apart from other RNA language models through its longer context length, smaller model size, and the use of human pre-training data, which enhances classification performance compared to models pre-trained with RNAcentral (Fig.3B).

A performance plateau may have been reached for binary coding potential classification, which could explain why lncRNA-BERT did not consistently outperform existing methods. Even a simple logistic regression algorithm (CPAT) can reach an F1-score of 0.89 on GENCODE/RefSeq (Fig.2), indicating the triviality of this task. At the same time, the problem is complicated by the false assumption of an unambiguous separation between pcRNAs and lncRNAs. Some lncRNAs are known to contain short ORFs that encode functional micro-peptides [10], whereas some pcRNAs can have lncRNA-like regulatory functions [11, 12]. Hence, it is impossible to obtain 100% accuracy without overfitting to the human annotation system. Here, a method such as LncADeep benefits from incorporating protein alignment data in its predictions, which is also dependent on currently available knowledge. In contrast, lncRNA-BERT distinguishes coding from non-coding RNA without access to annotations, paving the way for a more nuanced view of the two RNA classes in future work.

We demonstrated that the choice of encoding method and pre-training data is critical for lncRNA-BERT's performance. This highlights two key areas for potential improvements in NLMs in future studies. Regarding encoding methods, each have a different effect on the obtained sequence coverage, resolution, efficiency, and information retention. NUC represents nucleotides individually rather than combining them into single input positions, resulting in a demanding attention operation and a limited context length. While context length may be extended through architectural advancements, such as Attention with Linear Biases [23, 41] and Flash Attention [23, 30], adopting a more compressive encoding method is still advisable. BPE reduces the sequence length more efficiently than NUC and K-mer, but leads to reduced probing performance. This indicates that BPE models are less capable of recognizing coding potential after pre-training, in comparison to using K-mer or CSE. We attribute this to BPE's frequency-based vocabulary, which may not be biologically relevant and breaks up the three-base periodicity. Inconsistent token size also introduces problems for other fine-tuning tasks that require nucleotide resolution. Of all encoding methods, CSE enables the strongest compression by using learnable convolutions to recognize complex patterns from RNA sequences. Despite this, 3-mer tokenization obtains similar performance, indicating that a context length of 2,304 nucleotides is sufficient for recognizing the coding potential of most RNAs. lncRNA-BERT with CSE may potentially lead to superior performance when fine-tuned for other tasks that require longer context lengths. Ultimately, the choice of encoding method depends on the characteristics of a specific problem and dataset, and identifying a universal solution requires further evaluation on diverse fine-tuning tasks.

Regarding pre-training data, an NLM requires human pcRNA/lncRNA to achieve maximum performance in distinguishing the two classes, as shown in Fig.3 and Fig.5. This demonstrates the necessity of a sufficient amount of task-specific pre-training data. Nevertheless, we anticipate that lncRNA-BERT and other RNA language models may be improved by increasing the dataset size when a proper class balance and sequence diversity are ensured. Adding genetic variation has helped DNA language models generalize over subtle signals between different individuals and phylogenetic signals between species. For example, pre-training the Nucleotide Transformer with data from the 1000 Genomes Project enables it to perform variant prioritization [25], and GENA-LM can be used for taxonomic classification after pre-training on multi-species data [24]. In contrast, our results show that using the 37M multi-species RNAcentral dataset as the only pre-training source does not improve lncRNA-BERT, but results in a bias towards RNA types other than pcRNA and lncRNA (Fig.3C). Algorithms such as CD-HIT and MMSeqs2 can remove redundant sequences from datasets like RNAcentral, which may be the key to ensuring proper data balance and diversity. RiNALMo implements this approach, but excludes pcRNA [30]. Hence, pre-training a model with an efficient encoding method on a well-balanced dataset encompassing all types of RNA may yield the next generation of RNA NLMs.

Data Availability Statement

The data supporting the findings of this study are openly available from GitHub at `https://github.com/luukromeijn/lncRNA-Py`, GENCODE (v46) at `https://www.gencodegenes.org/`, RefSeq (v225) at `https://www.ncbi.nlm.nih.gov/refseq/`, NONCODE (v6) at `http://v6.noncode.org/`, and RNA-central (v24) at `https://rnacentral.org/`.

Disclosure of interest

All authors declare that they have no conflicts of interest.

Funding

There was no external sponsorship and funding to this research.

References

[1] M. Cesana, D. Cacchiarelli, I. Legnini, T. Santini, O. Sthandier, M. Chinappi, A. Tramontano, and I. Bozzoni, "A long noncoding rna controls muscle differentiation by functioning as a competing endogenous rna," *Cell*, vol. 147, no. 2, p. 358–369, Oct. 2011. [Online]. Available: http://dx.doi.org/10.1016/j.cell.2011.09.028

[2] S. Li, Q. Xiong, M. Chen, B. Wang, X. Yang, M. Yang, Q. Wang, Z. Cui, and F. Ge, "Long noncoding rna hotair interacts with y-box protein-1 (ybx1) to regulate cell proliferation," *Life Science Alliance*, vol. 4, no. 9, p. e202101139, Jul. 2021. [Online]. Available: http://dx.doi.org/10.26508/lsa.202101139

[3] C. Fan, R. González-Prieto, T. B. Kuipers, A. C. O. Vertegaal, P. A. van Veelen, H. Mei, and P. ten Dijke, "The lncrna lets1 promotes tgf-–induced emt and cancer cell migration by transcriptionally activating a tr1-stabilizing mechanism," *Science Signaling*, vol. 16, no. 790, Jun. 2023. [Online]. Available: http://dx.doi.org/10.1126/scisignal.adf1947

[4] L. Zhao, J. Wang, Y. Li, T. Song, Y. Wu, S. Fang, D. Bu, H. Li, L. Sun, D. Pei, Y. Zheng, J. Huang, M. Xu, R. Chen, Y. Zhao, and S. He, "Noncodev6: an updated database dedicated to long non-coding rna annotation in both animals and plants," *Nucleic Acids Research*, vol. 49, no. D1, p. D165–D171, Nov. 2020. [Online]. Available: http://dx.doi.org/10.1093/nar/gkaa1046

[5] B. A. Sweeney, A. I. Petrov, C. E. Ribas, R. D. Finn, A. Bateman, M. Szymanski, W. M. Karlowski, S. E. Seemann, J. Gorodkin, J. J. Cannone, R. R. Gutell, S. Kay, S. Marygold, G. dos Santos, A. Frankish, J. M. Mudge, R. Barshir, S. Fishilevich, P. P. Chan, T. M. Lowe, R. Seal, E. Bruford, S. Panni, P. Porras, D. Karagkouni, A. G. Hatzigeorgiou, L. Ma, Z. Zhang, P.-J. Volders, P. Mestdagh, S. Griffiths-Jones, B. Fromm, K. J. Peterson, I. Kalvari, E. P. Nawrocki, A. S. Petrov, S. Weng, P. Bouchard-Bourelle, M. Scott, L. M. Lui, D. Hoksza, R. C. Lovering, B. Kramarz, P. Mani, S. Ramachandran, and Z. Weinberg, "Rnacentral 2021: secondary structure integration, improved sequence search and new member databases," *Nucleic Acids Research*, vol. 49, no. D1, p. D212–D220, Oct. 2020. [Online]. Available: http://dx.doi.org/10.1093/nar/gkaa921

[6] L. Kong, Y. Zhang, Z.-Q. Ye, X.-Q. Liu, S.-Q. Zhao, L. Wei, and G. Gao, "Cpc: assess the protein-coding potential of transcripts using sequence features and support vector machine," *Nucleic Acids Research*, vol. 35, no. suppl_2, p. W345–W349, Jul. 2007. [Online]. Available: http://dx.doi.org/10.1093/nar/gkm391

[7] L. Sun, H. Luo, D. Bu, G. Zhao, K. Yu, C. Zhang, Y. Liu, R. Chen, and Y. Zhao, "Utilizing sequence intrinsic composition to classify protein-coding and long non-coding transcripts," *Nucleic Acids Research*, vol. 41, no. 17, p. e166–e166, Jul. 2013. [Online]. Available: http://dx.doi.org/10.1093/nar/gkt646

[8] L. Wang, H. J. Park, S. Dasari, S. Wang, J.-P. Kocher, and W. Li, "Cpat: Coding-potential assessment tool using an alignment-free logistic regression model," *Nucleic Acids Research*, vol. 41, no. 6, p. e74–e74, Jan. 2013. [Online]. Available: http://dx.doi.org/10.1093/nar/gkt006

[9] Y.-J. Kang, D.-C. Yang, L. Kong, M. Hou, Y.-Q. Meng, L. Wei, and G. Gao, "Cpc2: a fast and accurate coding potential calculator based on sequence intrinsic features," *Nucleic Acids Research*, vol. 45, no. W1, p. W12–W16, May 2017. [Online]. Available: http://dx.doi.org/10.1093/nar/gkx428

[10] Y. Pang, C. Mao, and S. Liu, "Encoding activities of non-coding rnas," *Theranostics*, vol. 8, no. 9, p. 2496–2507, 2018. [Online]. Available: http://dx.doi.org/10.7150/thno.24677

[11] A. M. Mustoe, S. Busan, G. M. Rice, C. E. Hajdin, B. K. Peterson, V. M. Ruda, N. Kubica, R. Nutiu, J. L. Baryza, and K. M. Weeks, "Pervasive regulatory functions of mrna structure revealed by high-resolution shape probing," *Cell*, vol. 173, no. 1, pp. 181–195.e18, Mar. 2018. [Online]. Available: http://dx.doi.org/10.1016/j.cell.2018.02.034

[12] J. Li, X. Zhang, and C. Liu, "The computational approaches of lncrna identification based on coding potential: Status quo and challenges," *Computational and Structural Biotechnology Journal*, vol. 18, p. 3666–3677, 2020. [Online]. Available: http://dx.doi.org/10.1016/j.csbj.2020.11.030

[13] S. Liu, X. Zhao, G. Zhang, W. Li, F. Liu, S. Liu, and W. Zhang, "Predlnc-gfstack: A global sequence feature based on a stacked ensemble learning method for predicting lncrnas from transcripts," *Genes*, vol. 10, no. 9, p. 672, Sep. 2019. [Online]. Available: http://dx.doi.org/10.3390/genes10090672

[14] S. Han, Y. Liang, Q. Ma, Y. Xu, Y. Zhang, W. Du, C. Wang, and Y. Li, "Lncfinder: an integrated platform for long non-coding rna identification utilizing sequence intrinsic composition, structural information and physicochemical property," *Briefings in Bioinformatics*, vol. 20, no. 6, p. 2009–2027, Jul. 2018. [Online]. Available: http://dx.doi.org/10.1093/bib/bby065

[15] H. Feng, S. Wang, Y. Wang, X. Ni, Z. Yang, X. Hu, and S. Yang, "Lnccat: An orf attention model to identify lncrna based on ensemble learning strategy and fused sequence information," *Computational and Structural Biotechnology Journal*, vol. 21, p. 1433–1447, 2023. [Online]. Available: http://dx.doi.org/10.1016/j.csbj.2023.02.012

[16] C. Yang, L. Yang, M. Zhou, H. Xie, C. Zhang, M. D. Wang, and H. Zhu, "Lncadeep: anab initiolncrna identification and functional annotation tool based on deep learning," *Bioinformatics*, vol. 34, no. 22, p. 3825–3834, May 2018. [Online]. Available: http://dx.doi.org/10.1093/bioinformatics/bty428

[17] S. T. Hill, R. Kuintzle, A. Teegarden, E. Merrill, P. Danaee, and D. A. Hendrix, "A deep recurrent neural network discovers complex biological rules to decipher rna protein-coding potential," *Nucleic Acids Research*, vol. 46, no. 16, p. 8105–8113, Jul. 2018. [Online]. Available: http://dx.doi.org/10.1093/nar/gky567

[18] A. P. Camargo, V. Sourkov, G. Pereira, and M. Carazzolle, "Rnasamba: neural network-based assessment of the protein-coding potential of rna sequences," *NAR Genomics and Bioinformatics*, vol. 2, no. 1, Jan. 2020. [Online]. Available: http://dx.doi.org/10.1093/nargab/lqz024

[19] J. Baek, B. Lee, S. Kwon, and S. Yoon, "Lncrnanet: long non-coding rna identification using deep learning," *Bioinformatics*, vol. 34, no. 22, p. 3889–3897, May 2018. [Online]. Available: http://dx.doi.org/10.1093/bioinformatics/bty418

[20] J. Meng, Q. Kang, Z. Chang, and Y. Luan, "Plncrna-hdeep: plant long noncoding rna prediction using hybrid deep learning based on two encoding styles," *BMC Bioinformatics*, vol. 22, no. S3, May 2021. [Online]. Available: http://dx.doi.org/10.1186/s12859-020-03870-2

[21] D. Singh and J. Roy, "A large-scale benchmark study of tools for the classification of protein-coding and non-coding rnas," *Nucleic Acids Research*, vol. 50, no. 21, p. 12094–12111, Nov. 2022. [Online]. Available: http://dx.doi.org/10.1093/nar/gkac1092

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017. [Online]. Available: http://arxiv.org/abs/1706.03762

[23] Z. Zhou, Y. Ji, W. Li, P. Dutta, R. Davuluri, and H. Liu, "Dnabert-2: Efficient foundation model and benchmark for multi-species genome," 2023. [Online]. Available: https://arxiv.org/abs/2306.15006

[24] V. Fishman, Y. Kuratov, A. Shmelev, M. Petrov, D. Penzar, D. Shepelin, N. Chekanov, O. Kardymon, and M. Burtsev, "Gena-lm: A family of open-source foundational dna language models for long sequences," Jun. 2023. [Online]. Available: http://dx.doi.org/10.1101/2023.06.12.544594

[25] H. Dalla-Torre, L. Gonzalez, J. Mendoza-Revilla, N. Lopez Carranza, A. H. Grzywaczewski, F. Oteri, C. Dallago, E. Trop, B. P. de Almeida, H. Sirelkhatim, G. Richard, M. Skwark, K. Beguir, M. Lopez, and T. Pierrot, "Nucleotide transformer: building and evaluating robust foundation models for human genomics," *Nature Methods*, Nov. 2024. [Online]. Available: http://dx.doi.org/10.1038/s41592-024-02523-z

[26] W. Yin, Z. Zhang, L. He, R. Jiang, S. Zhang, G. Liu, X. Zhang, T. Qin, and Z. Xie, "Ernie-rna: An rna language model with structure-enhanced representations," Mar. 2024. [Online]. Available: http://dx.doi.org/10.1101/2024.03.17.585376

[27] M. Akiyama and Y. Sakakibara, "Informative rna base embedding for rna structural alignment and clustering by deep representation learning," *NAR Genomics and Bioinformatics*, vol. 4, no. 1, Jan. 2022. [Online]. Available: http://dx.doi.org/10.1093/nargab/lqac012

[28] N. Wang, J. Bian, Y. Li, X. Li, S. Mumtaz, L. Kong, and H. Xiong, "Multi-purpose rna language modelling with motif-aware pretraining and type-guided fine-tuning," *Nature Machine Intelligence*, vol. 6, no. 5, p. 548–557, May 2024. [Online]. Available: http://dx.doi.org/10.1038/s42256-024-00836-4

[29] J. Chen, Z. Hu, S. Sun, Q. Tan, Y. Wang, Q. Yu, L. Zong, L. Hong, J. Xiao, T. Shen, I. King, and Y. Li, "Interpretable rna foundation model from unannotated data for highly accurate rna structure and function predictions," 2022. [Online]. Available: https://arxiv.org/abs/2204.00300

[30] R. J. Penić, T. Vlašić, R. G. Huber, Y. Wan, and M. Šikić, "Rinalmo: General-purpose rna language models can generalize well on structure prediction tasks," 2024. [Online]. Available: https://arxiv.org/abs/2403.00043

[31] A. Frankish, S. Carbonell-Sala, M. Diekhans, I. Jungreis, J. Loveland, J. Mudge, C. Sisu, J. Wright, C. Arnan, I. Barnes, A. Banerjee, R. Bennett, A. Berry, A. Bignell, C. Boix, F. Calvet, D. Cerdán-Vélez, F. Cunningham, C. Davidson, S. Donaldson, C. Dursun, R. Fatima, S. Giorgetti, C. Giron, J. Gonzalez, M. Hardy, P. Harrison, T. Hourlier, Z. Hollis, T. Hunt, B. James, Y. Jiang, R. Johnson, M. Kay, J. Lagarde, F. Martin, L. Gómez, S. Nair, P. Ni, F. Pozo, V. Ramalingam, M. Ruffier, B. Schmitt, J. Schreiber, E. Steed, M.-M. Suner, D. Sumathipala, I. Sycheva, B. Uszczynska-Ratajczak, E. Wass, Y. Yang, A. Yates, Z. Zafrulla, J. Choudhary, M. Gerstein, R. Guigo, T. J. P. Hubbard, M. Kellis, A. Kundaje, B. Paten, M. Tress, and P. Flicek, "Gencode: reference annotation for the human and mouse genomes in 2023," *Nucleic Acids Research*, vol. 51, no. D1, p. D942–D949, Nov. 2022. [Online]. Available: http://dx.doi.org/10.1093/nar/gkac1071

[32] N. A. O'Leary, M. W. Wright, J. R. Brister, S. Ciufo, D. Haddad, R. McVeigh, B. Rajput, B. Robbertse, B. Smith-White, D. Ako-Adjei, A. Astashyn, A. Badretdin, Y. Bao, O. Blinkova, V. Brover, V. Chetvernin, J. Choi, E. Cox, O. Ermolaeva, C. M. Farrell, T. Goldfarb, T. Gupta, D. Haft, E. Hatcher, W. Hlavina, V. S. Joardar, V. K. Kodali, W. Li, D. Maglott, P. Masterson, K. M. McGarvey, M. R. Murphy, K. O'Neill, S. Pujar, S. H. Rangwala, D. Rausch, L. D. Riddick, C. Schoch, A. Shkeda, S. S. Storz, H. Sun, F. Thibaud-Nissen, I. Tolstoy, R. E. Tully, A. R. Vatsan, C. Wallin, D. Webb, W. Wu, M. J. Landrum, A. Kimchi, T. Tatusova, M. DiCuccio, P. Kitts, T. D. Murphy, and K. D. Pruitt, "Reference sequence (refseq) database at ncbi: current status, taxonomic expansion, and functional annotation," *Nucleic Acids Research*, vol. 44, no. D1, p. D733–D745, Nov. 2015. [Online]. Available: http://dx.doi.org/10.1093/nar/gkv1189

[33] L. Fu, B. Niu, Z. Zhu, S. Wu, and W. Li, "Cd-hit: accelerated for clustering the next-generation sequencing data," *Bioinformatics*, vol. 28, no. 23, p. 3150–3152, Oct. 2012. [Online]. Available: http://dx.doi.org/10.1093/bioinformatics/bts565

[34] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, K. Erk and N. A. Smith, Eds. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 1715–1725. [Online]. Available: https://aclanthology.org/P16-1162

[35] L. Romeijn, A. Bernatavicius, and D. Vu, "Mycoai: Fast and accurate taxonomic classification for fungal its sequences," *Molecular Ecology Resources*, vol. 24, no. 8, Aug. 2024. [Online]. Available: http://dx.doi.org/10.1111/1755-0998.14006

[36] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2020. [Online]. Available: https://arxiv.org/abs/2010.11929

[37] S. He, B. Gao, R. Sabnis, and Q. Sun, "Nucleic transformer: Classifying dna sequences with self-attention and convolutions," *ACS Synthetic Biology*, vol. 12, no. 11, p. 3205–3214, Nov. 2023. [Online]. Available: http://dx.doi.org/10.1021/acssynbio.3c00154

[38] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2018. [Online]. Available: https://arxiv.org/abs/1810.04805

[39] A. Li, J. Zhang, and Z. Zhou, "Plek: a tool for predicting long non-coding rnas and messenger rnas based on an improved k-mer scheme," *BMC Bioinformatics*, vol. 15, no. 1, Sep. 2014. [Online]. Available: http://dx.doi.org/10.1186/1471-2105-15-311

[40] M. F. Sloma and D. H. Mathews, "Exact calculation of loop formation probability identifies folding motifs in rna secondary structures," *RNA*, vol. 22, no. 12, p. 1808–1818, Oct. 2016. [Online]. Available: http://dx.doi.org/10.1261/rna.053694.115

[41] M. T. Tahmid, H. S. Shahgir, S. Mahbub, Y. Dong, and M. S. Bayzid, "Birna-bert allows efficient rna language modeling with adaptive tokenization," Jul. 2024. [Online]. Available: http://dx.doi.org/10.1101/2024.07.02.601703