nature machine intelligence

Article

https://doi.org/10.1038/s42256-026-01295-9

NucleicBERT interprets RNA sequence space through self-supervised language modelling

Received: 14 December 2025

Accepted: 23 July 2026

Published online: 03 September 2026

Check for updates

Utkarsh Upadhyay[1], Julian Herold[3], Markus Götz[2,3] & Alexander Schug[1,3]

Much of the human genome's non-protein-coding fraction acts directly through RNA, yet the structural and functional roles encoded in these sequences remain poorly understood. Applying deep learning is hindered by scarce RNA structural data and it remains unclear what biological constraints such models can recover directly from the abundant RNA sequences alone. Here, to address these challenges, we developed NucleicBERT, a self-supervised masked-language model that learns contextual representations from single sequences without evolutionary information. Explainable artificial intelligence analyses show that the model organizes RNA sequences in latent space and encodes structural properties indicating that biologically meaningful constraints are learned from sequence correlations alone. When fine-tuned for downstream structural and functional tasks, NucleicBERT requires only single sequences while matching or exceeding current RNA prediction models. This alignment-free framework addresses the scarcity of annotated 3D RNA data while providing a rapid, computational complement to experimental techniques. By bridging abundant unlabelled sequence data with scarce structural annotations, NucleicBERT advances RNA structure prediction and informs how large language models encode biological information.

RNA plays essential roles in cellular processes ranging from genetic information transfer to forming the structural and catalytic core of the ribosome[1]. Noncoding RNAs (ncRNAs) regulate gene expression[2,3], yet the functionality of many remains poorly characterized. This knowledge gap represents both a fundamental scientific challenge and an opportunity for therapeutic development. In particular, RNA-targeted drug discovery[4,5] would benefit from improved structure–function predictions, as bacterial drug resistance[6] frequently involves ribosomal targets.

While RNA structure is critical for understanding function, experimental structure determination remains technically challenging and low-throughput. Although NMR spectroscopy and X-ray crystallography[7,8] provide detailed structural information, these approaches are labour-intensive, requiring substantial time and resources. Despite recent advances in cryo-electron microscopy (cryo-EM)[9], experimental bottlenecks persist, underscoring the need for scalable computational alternatives.

The success of deep learning in protein-structure prediction[10,11] has inspired similar efforts for RNA. However, RNA structure prediction faces a fundamental data scarcity problem: while sequencing technologies have generated millions of RNA sequences, the corresponding structural data remains limited. This sequence–structure gap represents a major obstacle in computational structural biology.

Current computational approaches largely rely on evolutionary information extracted from multiple sequence alignments (MSAs). Direct coupling analysis (DCA)[12–14] pioneered the use of coevolutionary signals for protein-structure prediction, with subsequent adaptation to RNA[15]. Supervised methods such as RNAContact[16], CoCoNet[17] or the transformer-based model BARNACLE[18] build on these foundations to predict structural interactions from sequences. However,

[1]Jülich Supercomputing Centre, Forschungszentrum Jülich, Jülich, Germany. [2]Helmholtz AI, Munich, Germany. [3]Present address: Scientific Computing Center, Karlsruhe Institute of Technology, Karlsruhe, Germany. e-mail: alexander.schug@kit.edu

these approaches face practical limitations: evolutionary analysis is computationally expensive, and high-quality MSAs are unavailable or shallow for many RNA families. Recent RNA models have improved performance on specific tasks, yet most remain constrained by data dependencies and narrow task design.

Here, we present NucleicBERT, a large language model designed to overcome these limitations by learning directly from single RNA sequences. Trained on 30 million ncRNA sequences using masked language modelling, NucleicBERT achieves competitive or superior performance across diverse structural and functional prediction tasks without relying on evolutionary information. Based on the bidirectional encoder representations from transformers (BERT) architecture[19], our model treats nucleotides as tokens and RNA sequences as sentences, enabling it to capture long-range and context-dependent relationships through self-attention mechanisms (Fig. 1; see the 'Pretraining architecture and dataset' section in the Methods). This design enables large-scale, alignment-free learning and generalization across diverse RNA types and prediction tasks.

While recent RNA language models such as RNA-FM[20] and RiNALMo[21] emphasize large-scale pretraining and predictive performance, NucleicBERT is designed to probe what biological constraints can be learned from RNA sequence alone. Through single-sequence self-supervision and systematic interpretability and perturbation analyses, we link learned representations to known structural and evolutionary constraints, shifting the focus from model scaling to understanding how language models encode biologically meaningful information.

A critical challenge in applying deep learning to biological problems is model explainability. To address this, we perform a systematic explainable artificial intelligence (AI) analysis to understand the biological knowledge NucleicBERT captures, a level of interpretability assessment not previously conducted for RNA language models. Through saliency mapping, attention-weight profiling and perturbation-based sensitivity analysis, we demonstrate how the model identifies structurally and functionally important sequence regions, thereby providing mechanistic insight into its internal representations and decision-making process.

We make three primary contributions: (1) NucleicBERT achieves competitive or superior performance compared with specialized tools across a range of structural and functional prediction tasks; (2) its modular, transformer-based architecture enables straightforward adaptation to new downstream applications; and (3) by incorporating explainable AI techniques, NucleicBERT goes beyond black-box modelling to provide transparent interpretation of its learned representations. In addition, NucleicBERT uses a direct, lightweight sequence-embedding scheme that keeps the model simple while supporting broad downstream utility.

This establishes NucleicBERT as a single-sequence model for RNA, unifying structure and function prediction within a single interpretable framework that can accelerate both fundamental discovery and therapeutic development.

Results

Here, we present the results for NucleicBERT, evaluate whether pretraining captures task-relevant RNA sequence information, describe the performance of our model on multiple downstream tasks and analyse the explainability of this AI model. Across the downstream tasks, we report NucleicBERT under two training regimes: full fine-tuning, in which all backbone and head parameters are updated jointly, and linear probing, in which the pretrained backbone is frozen and only a task-specific head is trained. We also show the performance of a randomly initialized model under both regimes. This disentangles two effects that have not previously been discussed: how much downstream performance comes from the quality of pretrained representations, and how much comes from end-to-end task adaptation. Parameter counts in Table 1 are taken from the respective original publications; thermodynamic and small-convolutional neural network (CNN) baselines are included for reference but their parameter counts are not directly comparable to those of the language models.

Pretraining performance validation

To validate the effectiveness of our pretraining strategy, we conducted a comprehensive evaluation of the masked language modelling task on a subset of our validation data. NucleicBERT is trained to predict the identity of nucleotides that have been randomly masked out of RNA sequences

$$\mathcal{L}_{\mathrm{MLM}} = \sum_{i \in M} \log p(x_i | x_{\notin M}), \tag{1}$$

where for a randomly generated mask M that includes certain positions i in the sequence x, the model predicts the identity of the nucleotides x_i from the unmasked sequence $x_{\notin M}$ (see the 'Pretraining architecture and dataset' section in the Methods).

Our pretrained model achieved a mean accuracy of 0.831 on the masked token prediction task, representing a substantial improvement over the non-pretrained model baseline. This substantial performance gain demonstrates that our model captures complex sequence dependencies through the pretraining process (Fig. 2a).

We evaluated model performance using perplexity, a standard metric for language models that measures the model's uncertainty when predicting tokens. Perplexity is defined as the exponential of the negative log-likelihood of the sequence. For masked language models such as NucleicBERT, we compute pseudo-perplexity[22] as

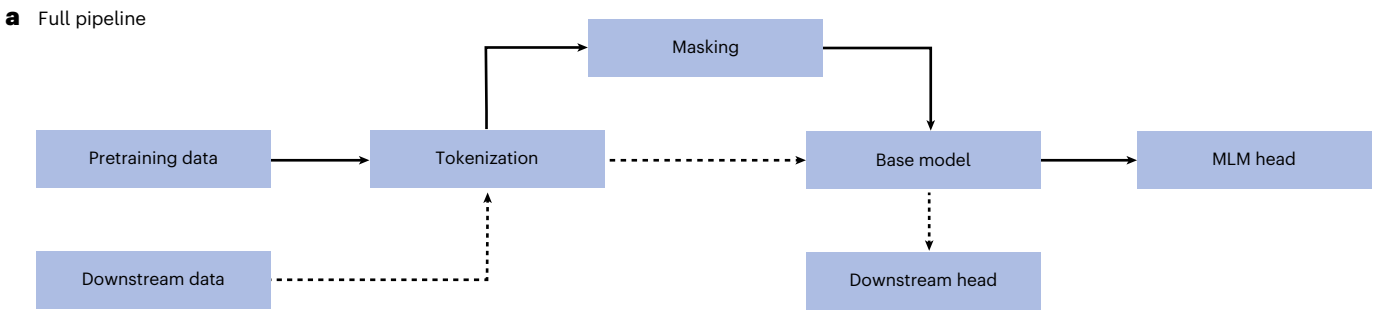
$$\mathrm{PPL}_{\mathrm{pseudo}}(x) = \exp\left(-\frac{1}{L}\sum_{i=1}^{L} \log p(x_i | x_{\neq i})\right), \tag{2}$$

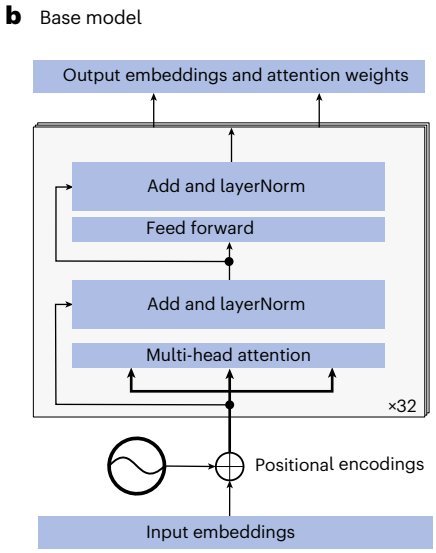
where L is the length of the input sequence and $p(x_i|x_{\neq i})$ represents the probability of nucleotide x_i given all other nucleotides in the sequence. Lower perplexity values indicate that the model confidently assigns high probabilities to correct predictions, while higher perplexity suggests greater uncertainty. Our model demonstrates a strong calibrated confidence relationship between perplexity and accuracy.

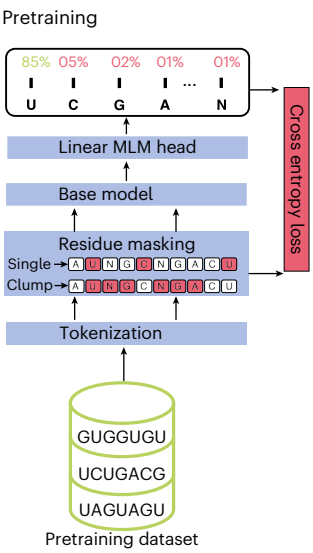
The high accuracy combined with the strong perplexity–accuracy correlation establishes that our model successfully learned to distinguish between probable and improbable RNA sequence patterns. These pretraining results provide empirical support for our approach and establish confidence in the quality of learned representations that we apply to subsequent structural prediction tasks and explainable AI analyses.

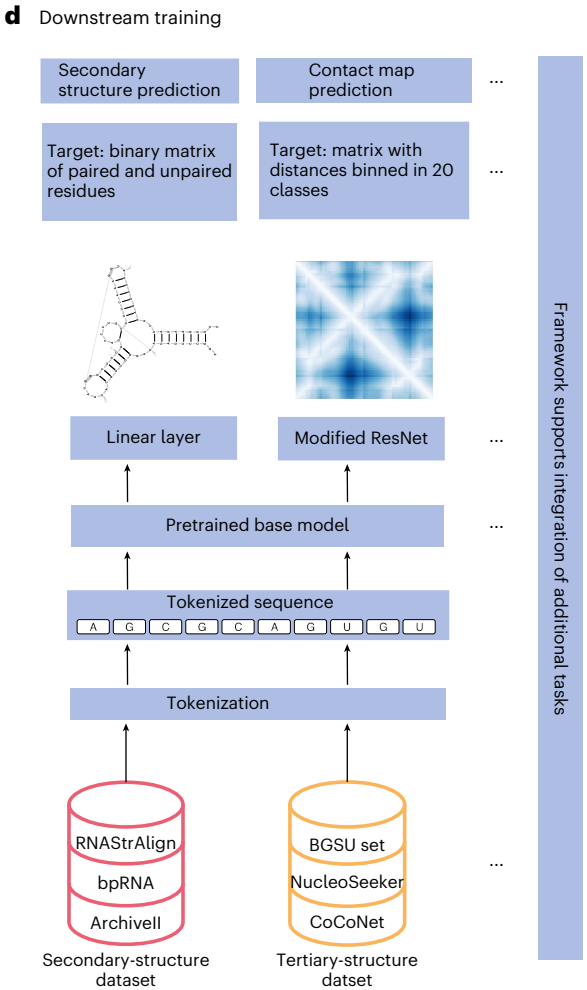
Fig. 1 | NucleicBERT architecture and training pipeline. a, The full pipeline showing the complete workflow from data preprocessing through final predictions. **b**, The base model architecture features transformer layers with multi-head attention and positional encodings to process sequence information. **c**, The pretraining uses masked language modelling on large datasets to learn basic nucleic acid patterns and relationships. **d**, The downstream training fine-tunes the pretrained model for specific tasks such as secondary-structure prediction and contact map prediction. We show only two tasks for brevity, and adding new tasks is straightforward. **e**, An example contact map prediction from the fine-tuned model, with long-range precision (>24 residues) at top-L/5, top-L/2 and top-L given above the panel. **f**, An example secondary-structure prediction, showing the reference structure (left) alongside the model prediction for the same sequence (right). **g**,**h**, Model explainability through saliency analysis reveals which input positions are most influential for predictions (**g**), and attention maps (**h**) capture biologically meaningful patterns such as base-pairing boundaries and long-range dependencies.

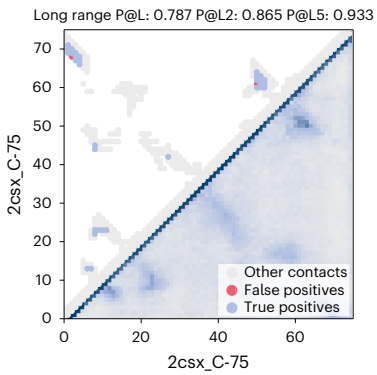
https://doi.org/10.1038/s42256-026-01295-9

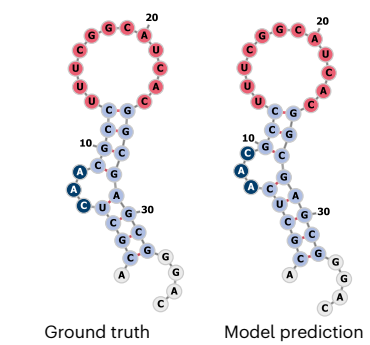


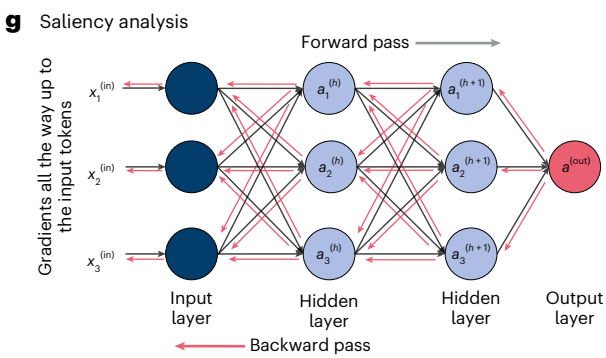


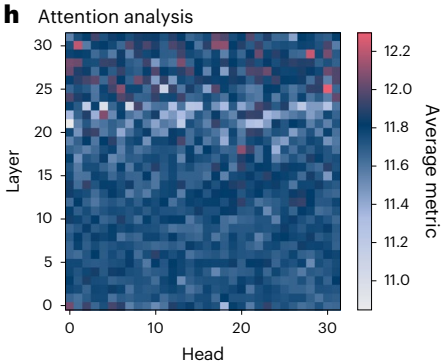












https://doi.org/10.1038/s42256-026-01295-9

Table 1 | NucleicBERT performance comparison across multiple downstream tasks

Secondary structure prediction performance							
Method	**Parameters**	**ArchiveII600**			**TS0**		
		Precision ↑	**Recall ↑**	**F1 ↑**	**Precision ↑**	**Recall ↑**	**F1 ↑**
RNAfold[28]	–	0.565	0.627	0.592	0.494	0.631	0.536
E2Efold[30]	0.72M	0.738	0.665	0.690	0.140	0.129	0.130
MXfold2[29]	0.80M	0.788	0.760	0.768	0.519	0.646	0.558
RNABERT[31]	0.48M	0.634	0.649	0.641	0.435	0.527	0.477
RNA-FM[20]	99.5M	0.752	0.737	0.744	0.518	0.620	0.564
RNAErnie+[32]	105M	0.886	**0.870**	**0.875**	0.575	**0.678**	0.622
NucleicBERT, fine-tuned	404M	**0.915**	0.848	0.872	**0.718**	0.610	**0.649**
NucleicBERT, linear probe	404M	0.819	0.539	0.581	0.622	0.283	0.342
Random init, fine-tuned	404M	0.640	0.503	0.522	0.285	0.184	0.204
Random init, linear probe	404M	0.027	0.003	0.005	0.021	0.002	0.004

Contact map prediction performance (long-range, ≥24 residue separation)				
Method	**Parameters**	**Top-*L*/5 ↑**	**Top-*L*/2 ↑**	**Top-*L* ↑**
RNA-FM[20]	99.5M	0.163	0.148	0.142
RiNALMo[21]	650M	0.508	0.445	0.389
NucleicBERT, fine-tuned	404M	**0.616**	**0.549**	**0.460**
NucleicBERT, linear probe	404M	0.573	0.493	0.423
Random init, fine-tuned	404M	0.472	0.422	0.355
Random init, linear probe	404M	0.199	0.173	0.157

Splice-site prediction accuracy across species (mean of donor and acceptor)						
Model	**Params**	**Zebrafish ↑**	**Fly ↑**	**Worm ↑**	**Plant ↑**	**Average ↑**
Spliceator[36]	–	0.935	0.929	0.916	0.929	0.927
SpliceBERT[38]	19.4M	0.957	0.946	0.934	0.936	0.943
DNABERT[37]	86M	0.951	0.931	0.909	0.909	0.925
RiNALMo[21]	650M	**0.972**	0.945	0.926	0.939	0.945
NucleicBERT, fine-tuned	404M	0.964	**0.949**	**0.935**	**0.944**	**0.948**
NucleicBERT, linear probe	404M	0.444	0.459	0.423	0.469	0.449
Random init, fine-tuned	404M	0.832	0.863	0.848	0.841	0.846
Random init, linear probe	404M	0.349	0.404	0.255	0.322	0.333

For NucleicBERT and a randomly initialized backbone with identical architecture, we report two training regimes: 'fine-tuned', where all parameters are updated jointly, and 'linear probe', where the backbone is frozen and only the task-specific head is trained. Parameter counts are taken from the respective original publications. Bold values indicate the method achieving the best result for each metric. Downstream benchmark values for NucleicBERT and random-init controls are single-run evaluations on fixed test sets; replicate standard errors are therefore not reported. F1, F1-score.

PHATE analysis of RNA sequence embeddings

To investigate the information captured by NucleicBERT's embeddings, we used PHATE[23], a diffusion-based dimensionality reduction method designed to preserve local and global structure in high-dimensional data. We projected pooled embeddings of seven RNA functional categories from three sources: the pretrained NucleicBERT model, a randomly initialized NucleicBERT model with identical architecture and a 6-mer frequency representation that serves as a parameter-free baseline. The 6-mer representation is the L1-normalized count vector over all 4,096 hexamer tokens of each sequence (see the 'Pretraining architecture and dataset' section in the Methods). Figure 2b shows the three projections side by side. The pretrained model produces more clearly separated RNA-family structure than the random-init and 6-mer baselines, including separation of similarly sized but functionally distinct piRNAs and siRNAs. The randomly initialized model produces a uniform circular distribution with mixed categories, an expected geometry for high-dimensional random vectors under diffusion-based reduction[23]. The 6-mer projection produces a sparse star-shaped layout in which class identity is largely lost. To move from visual inspection to a quantitative comparison, we computed k-nearest neighbours (kNN) classification accuracy ($k = 15$) directly in the embedding spaces (not on the 2D projections), where each sequence's RNA family is predicted by majority vote among its 15 nearest neighbours under Euclidean distance (Extended Data Table 1). Across all seven classes ($n = 7{,}000$), pretrained NucleicBERT shows robust performance: the advantage over the baselines persists when the analysis is restricted to the five mid-sized classes (lengths in [50, 500]) and when classes are explicitly length-matched. The randomly initialized backbone's non-trivial accuracy is an interesting observation. Two simple length-and-composition baselines explain most of it: a length-only kNN reaches 0.592 ± 0.057, and a length plus ACGU-composition kNN reaches 0.746 ± 0.017, both close to the random-init backbone. This is consistent with recent work showing that untrained transformers act as nonlinear random feature extractors that preserve compositional and length statistics of their inputs[24,25]. In our setup, mean-pooling over 32 layers and all valid positions makes the random-init embedding effectively a function of sequence length, GC content and a random nonlinear mixture of token statistics, all of which differ systematically across RNA families

https://doi.org/10.1038/s42256-026-01295-9

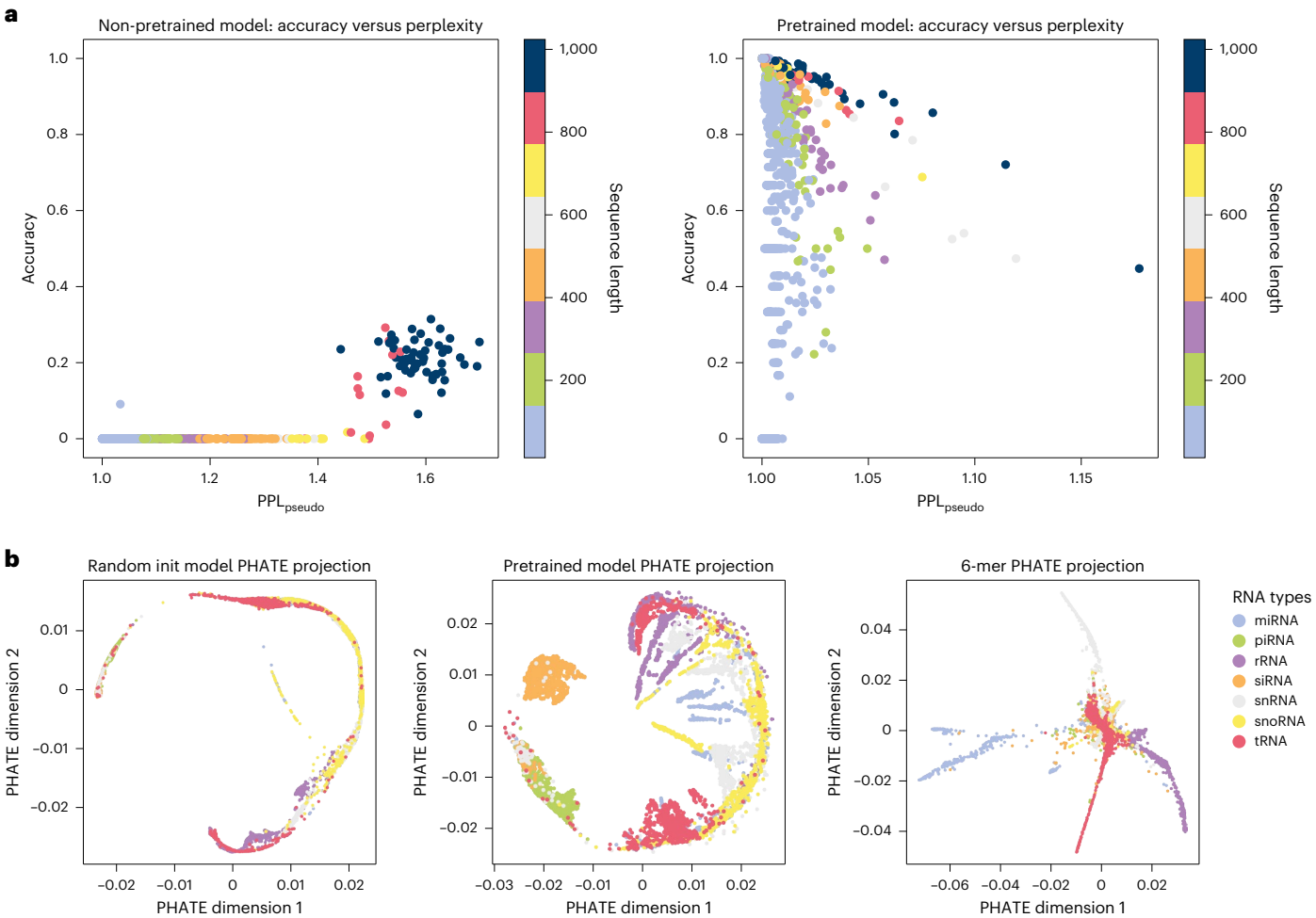


Fig. 2 | Pretraining validation and unsupervised biological discovery. **a**, NucleicBERT pretraining validation through an accuracy–perplexity relationship. Scatter plots comparing masked token prediction accuracy versus pseudo-perplexity for the non-pretrained model (left) and the pretrained model (right). Points are coloured by RNA sequence length. The non-pretrained model shows no meaningful relationship between perplexity and accuracy, with most predictions clustering near 0% or 25% accuracy. This distribution arises because the non-pretrained model consistently predicts a single token, resulting in accuracy that directly reflects the frequency of that token in the masked positions. Short sequences predominantly show 0% accuracy owing to insufficient masked tokens to reliably sample this frequency, while longer sequences approach 25% accuracy as they converge to the correct distribution. However, the pretrained model demonstrates a strong negative correlation between perplexity and accuracy, indicating calibrated confidence where low perplexity (high confidence) corresponds to high accuracy. This relationship validates that our pretraining process successfully learned meaningful RNA sequence representations beyond random guessing. **b**, PHATE projections of RNA sequence embeddings from three sources, coloured by RNA family (labels assigned for visualization only and not used during pretraining or projection). Left: a randomly initialized NucleicBERT. Middle: a pretrained NucleicBERT. Right: a 6-mer frequency representation. Quantitative kNN accuracies (computed in the original embedding spaces, not on the 2D projections shown) are reported in Extended Data Table 1.

(siRNAs are short and uniform-length, rRNAs are long and GC-rich). The relevant comparison is therefore not pretrained versus random-init in absolute terms, but the gap between them under length-matched conditions, where trivial features are controlled out. This supports the conclusion that masked language modelling produces representations encoding RNA-family-relevant sequence regularities beyond length and composition.

Performance on downstream tasks

Secondary-structure prediction. To evaluate NucleicBERT's capacity for secondary-structure prediction, we conducted comprehensive benchmarking against established baseline methods using two widely adopted datasets in the RNA structure prediction community. Our evaluation employed the ArchiveII600[26] dataset, a subset of ArchiveII containing 3,975 RNA structures from ten RNA families with sequence lengths under 600 base pairs, and the TS0 dataset from bpRNA-1m[27], comprising 1,305 test structures with lengths ranging from approximately 100 to 500 base pairs and 80% sequence identity clustering to ensure diversity (see the 'Secondary-structure architecture and dataset' section in the Methods). Table 1 presents the comparison of NucleicBERT against several state-of-the-art methods, including traditional thermodynamic approaches (RNAfold[28]), recent deep learning models (MXfold2[29] and E2Efold[30]) and contemporary RNA language models (RNABERT[31], RNA-FM[20] and RNAErnie+[32]). We used the published numbers to ensure identical metrics.

NucleicBERT demonstrates excellent performance across both evaluation datasets while using a simpler post-processing method (see the 'Secondary-structure architecture and dataset' section in the Methods). We have $\mathcal{O}(n^2)$ time complexity versus $\mathcal{O}(n^3)$ for dynamic programming-based methods[33], full GPU parallelization without CPU-bound bottlenecks, and end-to-end learning that eliminates the need for thermodynamic parameter integration.

The linear-probing rows show the value of pretraining signal. With the backbone frozen and only the head trained, NucleicBERT reaches

an F1 of 0.581 on ArchiveII600, while a randomly initialized backbone under the same protocol achieves an F1 of 0.005. The gap between these two rows is direct evidence that masked language modelling objective produces representations that already encode relevant features, before any task-specific gradient update. Full fine-tuning then adds a further 29 absolute F1 points on ArchiveII600, indicating that some structural information still has to be learned through downstream training.

On the more challenging TS0 dataset, NucleicBERT shows a similar performance. The consistent performance advantage across diverse RNA families and sequence lengths demonstrates the robustness of the learned representations and their transferability to downstream structural prediction tasks.

Distance and contact map prediction. To assess NucleicBERT's capability for distance and contact map prediction, with the latter resulting from thresholding the former, we conducted extensive evaluations on datasets derived from high-resolution RNA structures. Such a prediction represents a fundamental step towards understanding RNA tertiary structure, as it captures long-range nucleotide interactions and serves as a proxy for 3D structure prediction.

We evaluated NucleicBERT on two complementary datasets: a NucleoSeeker-generated[34] dataset containing 408 RNA structures and a Bowling Green State University (BGSU)[35] representative dataset comprising 467 structures (see the 'Tertiary-structure prediction architecture and dataset' section in the Methods).

In Table 1, we compare NucleicBERT against current state-of-the-art RNA language models, including RiNALMo and RNA-FM. NucleicBERT outperformed the evaluated baselines. We calculate the top-L metrics for ≥24 residue separation, as this represents the most challenging aspect of this task, and these interactions are often critical for tertiary-structure formation but cannot be inferred from local sequence patterns.

NucleicBERT outperformed its random-init counterpart by roughly 11 absolute Top-L points under fine-tuning (0.460 versus 0.355) and by roughly 27 points under linear probing (0.423 versus 0.157). End-to-end adaptation also helps in both regimes, but it helps less when the starting representation is already informative: for Top-L/5, the gap between linear probing and full fine-tuning is only 0.616 versus 0.573 for pretrained NucleicBERT, compared with 0.472 versus 0.199 for random init. This pattern supports the interpretation that the pretrained representation already contains contact-relevant information. Accordingly, the downstream head requires less task-specific adaptation than when trained on a randomly initialized backbone.

Splice-site prediction. When genes are first copied from DNA to RNA, they contain both protein-coding regions (exons) and noncoding segments (introns). RNA splicing removes introns and joins exons together to produce mature mRNA. Their precise prediction is critical for genome annotation and functional genomic studies.

To assess NucleicBERT's capability in this functional prediction task, we evaluated its performance on splice-site classification using established benchmark datasets (see the 'Splice-site prediction architecture and dataset' section in the Methods). Table 1 presents the comparison of NucleicBERT against established splice-site prediction methods across four different species: zebrafish, fly, worm and plant. The benchmark includes both traditional machine learning approaches (Spliceator[36]) and recent deep learning models (DNABERT[37], SpliceBERT[38] and RiNALMo[21]).

We make an interesting observation between linear probing (0.449 average) and full fine-tuning (0.948 average) because the gap is wider here than for any other downstream task. Two properties of the splice-site benchmark explain this. First, the task is abundantly labelled: each species comes with 10,000 positive and 10,000 negative 600-nucleotide windows centred on a GT or AG dinucleotide (see the 'Splice-site prediction architecture and dataset' section in the Methods), which is enough labelled data for even a 404M-parameter random-init backbone to reach 0.846 average accuracy when fine-tuned end-to-end. The headroom for pretraining to add to fine-tuned performance is therefore narrow. Second, it is a simpler binary task where the model has to decide whether a single fixed-window sequence contains a splice site, a binary judgement that is well-supported by the abundant labelled data.

Fitness prediction. To further assess NucleicBERT's capacity for capturing RNA sequence–function relationships, we implemented a fitness prediction task using a comprehensive mutational dataset[39] of RNA sequences (see the 'Fitness prediction architecture and dataset' section in the Methods). Here, fitness refers to the self-cleavage activity of CPEB3 ribozyme variants. Specifically, fitness measures the proportion of RNA molecules that successfully cut their phosphodiester backbone. This measure provides a quantitative assessment of how sequence mutations affect catalytic efficiency.

We benchmarked NucleicBERT against RiNALMo[21] and RNA-FM[20], alongside a randomly initialized NucleicBERT and a Hamming ball-query baseline (see the 'Fitness prediction architecture and dataset' section in the Methods). Under full fine-tuning all three pretrained backbones converge to high test-set R^2 vlaues (NucleicBERT 0.994, RiNALMo 0.991 and RNA-FM 0.996); the random-init backbone reaches 0.914, and the ball-query baseline shows 0.669. Pretraining yields roughly an 8.5-fold reduction in epochs to a useful R^2 over random initialization (Fig. 3e,f).

We also ran a frozen-probing variant, in which only the regression head is trained. Here, all four models give $R^2 \leq 0$. The value of pretraining on this dataset is therefore reflected in fine-tuning convergence speed and final accuracy, rather than in zero-shot or linear-probe performance.

The pretrained models demonstrate (Fig. 3) consistently excellent performance across all mutation values. By contrast, the non-pretrained models exhibit a different pattern; they perform worse on sequences with few mutations but improve as the number of mutations increases. Notably, the training data fitness variance decreases sharply as mutation count increases, starting at ~0.16 for single mutations and approaching zero for highly mutated sequences, suggesting that sequences with many mutations converge towards similar (probably low) fitness values. This makes higher-order predictions easier.

Explainable AI analysis

Understanding decision-making through saliency analysis. We conducted a comprehensive saliency analysis by backpropagating loss gradients[40,41] to identify input sequence positions that contribute most strongly to model predictions. This analysis was performed across three key tasks: masked token prediction, secondary-structure prediction and contact map prediction, revealing distinct patterns that provide insights into the model's learned representations.

For the masked token prediction task (Fig. 4a), saliency analysis confirmed that our model correctly focuses on the prediction objective[42,43]. Saliency values consistently showed pronounced spikes at positions containing `[MASK]` tokens, with the highest saliency scores occurring precisely at masked positions across sequences. This pattern validates that our masking strategy effectively directs the model's focus to the prediction tokens while incorporating relevant local context, confirming that the pretraining objective is being optimized as intended and that the model has learned to utilize positional relationships for token prediction.

For secondary-structure prediction (Fig. 4b), saliency analysis revealed a more complex relationship between the model's focus and structural elements. While we observed only weak correlations

https://doi.org/10.1038/s42256-026-01295-9

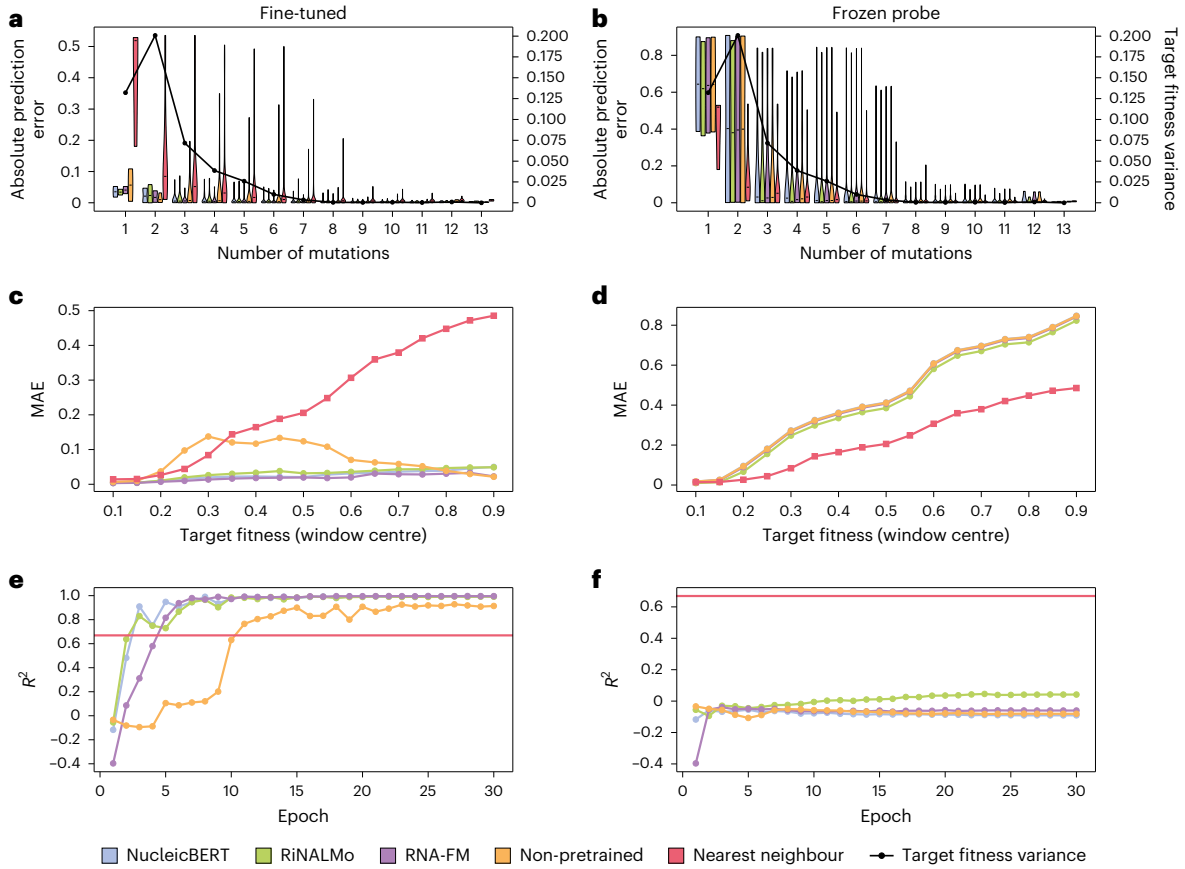


Fig. 3 | Fitness prediction on the CPEB3 ribozyme mutagenesis dataset. **a**,**b**, The absolute prediction error by mutation count for five models, with target fitness variance overlaid (right axis, black line), under full fine-tuning, in which all parameters are updated (**a**), and under the frozen probe, in which the backbone is fixed and only the regression head is trained (**b**). **c**,**d**, The mean absolute error across the fitness value range (sliding window) under full fine-tuning (**c**) and the frozen probe (**d**). **e**,**f**, The R^2 progression across 30 training epochs under full fine-tuning (**e**) and the frozen probe (**f**); the nearest-neighbour ball-query baseline ($R^2 = 0.669$) is shown as a horizontal reference in both. Pretrained backbones are NucleicBERT, RiNALMo [21] and RNA-FM[20]; the non-pretrained backbone is a randomly initialized NucleicBERT.

between saliency and base-pairing status, the model showed heightened importance at structurally important positions. Notably, the strongest saliency spikes occurred at transition points between paired and unpaired regions, suggesting that our model learned to identify structural boundaries and junction points that are critical for RNA folding[44].

The most compelling evidence for structural learning emerged from contact map prediction analysis (Fig. 4g–i), where we observed pronounced positive correlations between contact density and saliency. Positions with higher contact degrees consistently received greater importance from the model. The correlation strength for contact prediction suggests that our model's attention mechanisms are particularly well-suited for capturing long-range interactions and three-dimensional structural features. Collectively, these saliency analyses demonstrate that our RNA language model has developed a hierarchical understanding of RNA structure, from local sequence patterns during pretraining to complex structural elements in downstream tasks.

Assessing sequence authenticity through perturbation sensitivity. Beyond understanding how the model makes decisions, we evaluated NucleicBERT's capacity to distinguish authentic RNA sequences from artificially perturbed variants. This sequence authenticity assessment serves dual purposes: as a sensitivity analysis revealing the model's learned constraints, and as a biologically meaningful downstream task for validating sequence integrity in real-world applications (see the 'Shuffled sequence detection architecture and dataset' section in the Methods).

The model demonstrated remarkable sensitivity to sequence integrity, exhibiting a strong negative correlation between shuffle percentage and sequence authenticity classification ($r = -0.951$, $P = 4.04 \times 10^{-11}$). Performance remained robust for minimally perturbed sequences, maintaining >85% classification accuracy for shuffle levels below 10%. However, a pronounced performance decline occurred around the 20–25% shuffle threshold (Fig. 4g–i).

When perturbations were restricted to base-paired residues, the model exhibited an even stronger performance correlation ($r = -0.989$, $P = 2.43 \times 10^{-17}$), suggesting heightened sensitivity to disruptions in structurally important regions. Notably, even under complete base-pair shuffling (100%), the model maintained classification accuracy above 55%, indicating that it integrates both local sequence context and long-range structural constraints in its authenticity assessment (Fig. 4g–i).

Natural RNA families tolerate an average sequence variation of 26.9% ± 10.4% (see the 'Shuffled sequence detection architecture and dataset' section in the Methods), closely matching our shuffle detection performance. The convergence between the model's performance threshold (~25% shuffle) and the natural sequence variation boundary (26.9%) provides compelling evidence that NucleicBERT has internalized biologically relevant constraints on functional RNA sequence space (Fig. 4g–i), while complementary work by Lambert et al.[45] uses generative models to chart ribozyme diversity and functionality.

https://doi.org/10.1038/s42256-026-01295-9

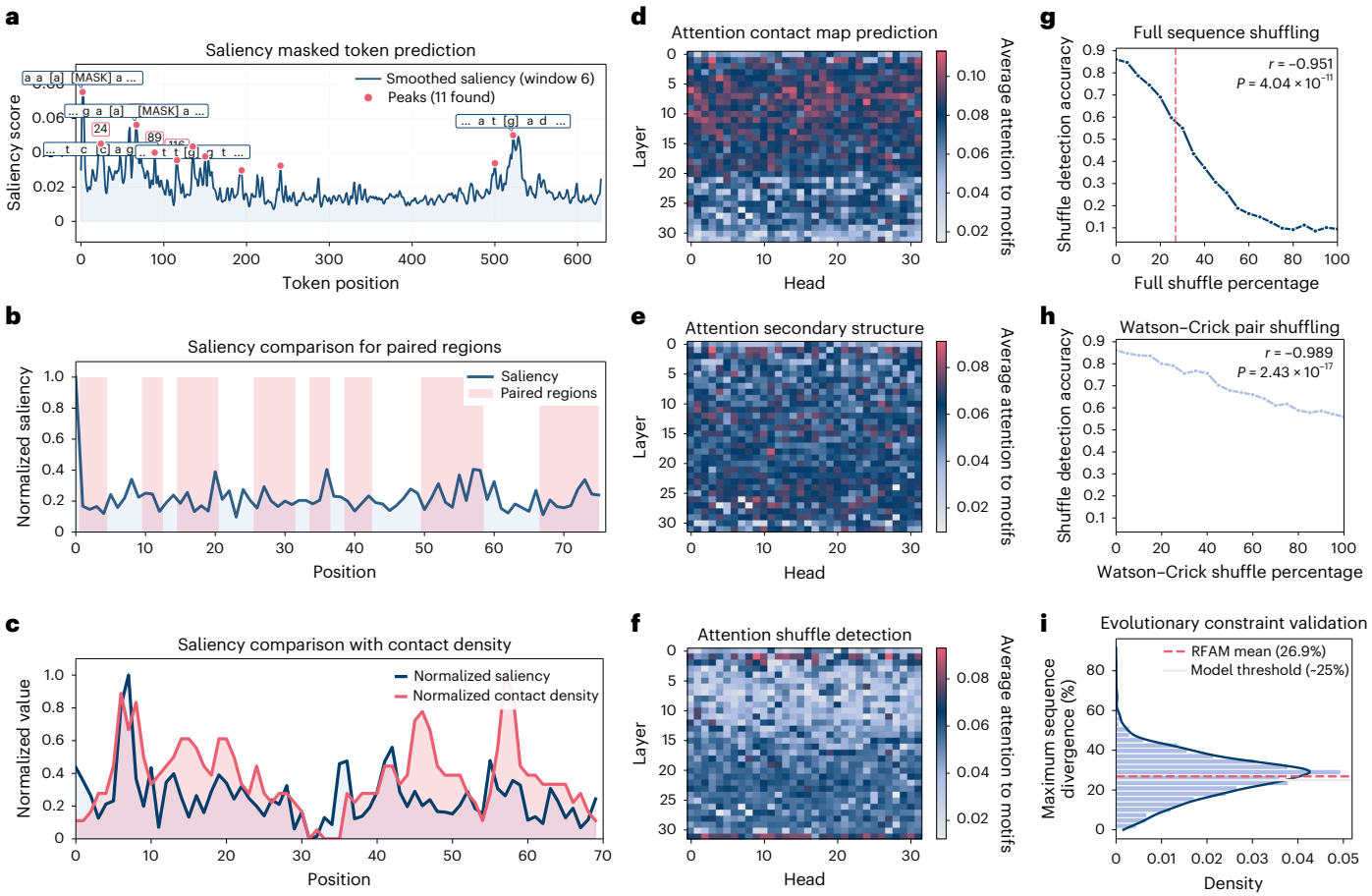


Fig. 4 | Explainable AI analysis and biological validation of NucleicBERT. **a**, The saliency analysis for masked token prediction; peaks correspond to masked positions, confirming the model correctly focuses on the prediction objective. **b**, A saliency comparison for secondary-structure prediction; saliency elevates at transition boundaries between paired and unpaired regions rather than at base-paired positions per se. **c**, A saliency comparison with contact density ($r = 0.387$), indicating the model consistently prioritizes positions with higher contact degrees. **d**–**f**, Task-specific attention patterns showing normalized attention weights by layer (*y* axis) and head (*x* axis) for contact map prediction (**d**), secondary-structure prediction (**e**) and shuffle detection as a non-structural control (**f**). Structural tasks engage distinct early-to-middle-layer pathways; the shuffle baseline does not. **g**, The shuffle-detection accuracy under full-sequence shuffling; the dashed red line marks the natural Rfam divergence mean (26.9%). **h**, The shuffle-detection accuracy under Watson–Crick pair-specific shuffling, demonstrating heightened sensitivity to disruptions in structurally important regions. Associations in **g** and **h** were assessed by two-sided Pearson correlation ($n = 21$ shuffle levels); no multiple-comparison adjustment was applied. **i**, The distribution of maximum sequence divergence across 2,244 Rfam families (≥5 sequences each); the model's ~25% performance threshold matches the natural-divergence boundary of 26.9% ± 10.4%, indicating that NucleicBERT has internalized biologically relevant sequence constraints.

Attention patterns reveal structural learning. To investigate how NucleicBERT processes different types of structural information, we analysed normalized self-attention patterns across layers and heads for three distinct downstream tasks: contact map prediction, secondary-structure prediction and shuffle detection as a non-structural control task. Following established protocols for biological sequence transformers[46], we computed normalized attention weights to ensure meaningful cross-task comparison. The attention heat maps reveal task-specific patterns that provide insights into the model's internal representations of RNA structural complexity.

The three tasks demonstrate hierarchical attention specialization. Contact map prediction (Fig. 4d) exhibits pronounced attention patterns, concentrated in early-to-middle layers (0–15), with scattered patterns in the deeper layers. This pattern suggests distributed processing of spatial relationships essential for tertiary-structure formation[47]. Secondary structure prediction (Fig. 4e) shows heightened attention in the initial layers (1–2), with lower attention in the middle layers and the distributed pattern in the deeper layers. The deepest layer shows high attention again[48]. Shuffle detection (Fig. 4f) exhibits a uniform attention pattern, providing a crucial baseline indicating minimal specialization for sequence-level discrimination.

The attention analysis reveals functional hierarchy within NucleicBERT's architecture. Early layers (0–5) consistently show elevated attention across structural tasks, suggesting these components serve as general RNA feature extractors for local motifs and base-pairing patterns[49]. Middle layers exhibit task-specific modulation: contact map prediction engages specialized high-attention regions while secondary-structure prediction shows correspondingly lower attention in these same regions. The shuffle detection task lacks such specialized patterns, confirming that structural tasks engage distinct neural pathways. The deeper layers (21–29) show uniform attention patterns across all three tasks, suggesting these layers process sequence-level information rather than structural features.

The task-specific specialization mirrors RNA structural biology complexity: tertiary structure requires integration of multiple long-range interactions, secondary structure involves local base-pairing with contextual constraints and sequence discrimination needs only general pattern recognition[20]. The clear differentiation between tasks demonstrates that attention analysis, when properly

https://doi.org/10.1038/s42256-026-01295-9

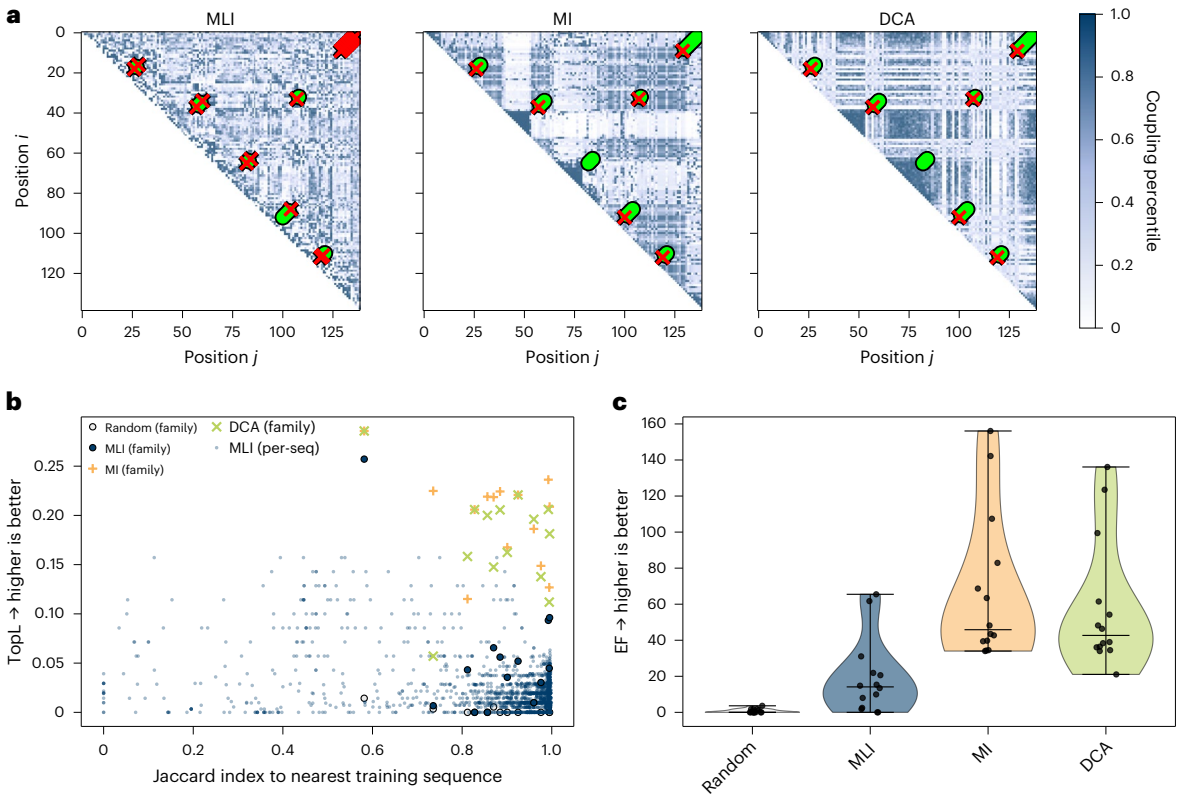


Fig. 5 | Single-sequence MLI recovers coevolution-like signals. a, Example coupling matrices for RF00050 (FMN riboswitch). MLI (left), MI (middle) and DCA (right). Per-panel quantile-normalized. Green markers denote top-L pairs that match the Rfam `SS_cons` reference, and red markers denote top-L misses. MLI uses single-sequence inputs only; MI and DCA use the full MSA. **b**, Per-sequence Top-L stratified by distance to the pretraining corpus. Each small point is one MSA sequence (MLI only) and the larger points are aggregated over MSAs; the x axis is its mean Jaccard index to the nearest sequence in the pretraining corpus ('1' represents the highest similarity). MLI quality is roughly flat in this distance, ruling out memorization as the dominant driver. **c**, The family-level EF distribution across 14 Rfam families. Violins show kernel density estimates of the EF distribution for each method; black points indicate individual Rfam families. Horizontal bars indicate mean values and vertical capped lines indicate the minimum-to-maximum range. MLI is computed from pretrained weights and single-sequence inputs only; MSA averaging is post hoc and serves only as a fair comparator with MI/DCA.

normalized and validated against appropriate controls, provides valuable insights into RNA language model organization and their capacity for learning structural principles.

Coevolution-like couplings from a single sequence. To test whether the pretrained backbone alone, given only single sequences, has internalized pairwise constraints, we extracted a masked-likelihood-influence (MLI) coupling matrix from the pretrained NucleicBERT (see the 'Coevolution analysis from single sequences' section in the Methods). For each sequence, MLI quantifies how strongly masking position j degrades the model's confidence in the native nucleotide at position i, yielding an $L \times L$ coupling matrix. Crucially, the model never sees the MSA: per-sequence matrices are computed independently and only averaged post hoc to produce a family-level estimate comparable with mutual information (MI) and DCA, both of which require an MSA by construction. We benchmarked MLI across 14 Rfam families against the canonical `SS_cons` contact set using Top-L precision, Top-$2L$ precision, AUPRC and the enrichment factor (EF) over the random base rate (see the 'Coevolution analysis from single sequences' section in the Methods and the 'Coevolution per-family metrics' table in the Supplementary Information).

Figure 5 summarizes the results. For RF00050 (FMN riboswitch) MLI surfaces the same anti-diagonal contact ladder as MI and DCA without any alignment input (Fig. 5a). The per-sequence Top-L is approximately stable across the Jaccard index of each query to its nearest neighbour in the MARS pretraining corpus (Fig. 5b), indicating that the recovered signal is not a memorization artefact restricted to families closely matching the training distribution. Across all 14 families, MLI sits well above the seeded-random null on EF (Fig. 5c) but below the MSA-grounded MI/DCA reference. Together, these results show that sequence-only masked language modelling internalizes a measurable, but bounded, fraction of the coevolutionary signal that classical MSA methods extract, supporting the claim that single-sequence foundation models can be treated as approximate, alignment-free coevolution tools when MSAs are shallow or unavailable.

Discussion

NucleicBERT represents a notable advancement in RNA computational biology by achieving competitive performance across both structural and functional prediction tasks using only single-sequence inputs. Unlike previous specialized RNA tools that excel in specific domains, NucleicBERT demonstrates consistent performance across secondary-structure prediction, contact map prediction and splice-site prediction, suggesting that our pretraining strategy captures fundamental principles governing RNA sequence–structure–function relationships. The modular architecture further enables new tasks to be incorporated with modest additional compute, extending the framework's utility beyond the tasks demonstrated here.

The elimination of MSA requirements addresses a critical limitation in the field. MSA-based methods face computational bottlenecks and limited applicability to novel RNAs or RNA families with

shallow MSAs. Our single-sequence approach maintains competitive accuracy while offering practical advantages.

Following fine-tuning, NucleicBERT develops task-specific specialization that mirrors RNA structural complexity. Layerwise attention and saliency patterns consistently highlight regions critical for structure formation, with distinct activation pathways emerging for tertiary versus secondary-structure tasks, indicating that downstream training reshapes the model's internal representations towards known structural determinants. Critically, the pretrained backbone already encodes biologically meaningful constraints before any task supervision is applied. The convergence between NucleicBERT's shuffle detection threshold (~25%) and the natural sequence variation boundary in Rfam families (26.9%) demonstrates that self-supervised training on raw sequences internalizes evolutionary constraints without explicit annotation. This separation between what pretraining contributes and what fine-tuning adds is made explicit by the linear-probing experiments: frozen pretrained embeddings substantially outperform frozen random-init embeddings across all structural tasks, while the fine-tuning gap is largest where labelled data is scarce and smallest where it is abundant.

The PHATE projections and MLI results provide two complementary windows into the geometry of the pretrained representations. The kNN analysis, computed in the original embedding spaces under length-matched conditions, confirms that pretrained embeddings organize RNA families into functionally coherent regions beyond what sequence length or nucleotide composition alone can explain. The MLI analysis goes further: by measuring how masking one position degrades the model's confidence at another, we show that the pretrained backbone recovers a measurable fraction of pairwise coevolutionary signal from single sequences, without access to any MSA. Across 14 Rfam families, MLI consistently sits above a random null and below the MSA-grounded references of MI and DCA. Together, these findings suggest that sequence-only masked language modelling internalizes a structured model of RNA sequence space that partially recapitulates what classical coevolutionary analysis requires deep alignments to extract.

In particular, NucleicBERT delivers competitive or improved performance compared with other methods across diverse RNA tasks, establishing it as a unifying framework with implications for, for example, RNA-targeted drug discovery. The model's explainability features enable integration with experimental workflows, where predictions can guide hypothesis formation and experimental design. As high-throughput measurements of RNA structure and function expand, pretrained representations such as those from NucleicBERT may form a computational basis for systematic exploration of sequence–function space, linking abundant sequence data to scarce functional annotations, thus concurrently advancing both RNA biology and therapeutic design.

Methods

Pretraining architecture and dataset

Our model consists of 32 transformer layers, each with 32 attention heads and an embedding dimension of 1,024. We employed learned positional encodings to capture position-specific information.

For tokenization, we trained a byte pair encoding tokenizer with a clump size of 1, effectively creating a character-level tokenization strategy in which each nucleotide or ambiguity character becomes its own token. This approach maintains maximal granularity without any merge operations, which is well-suited to the relatively small RNA alphabet.

The resulting vocabulary size is 25. It comprises the four standard RNA nucleotides (A, U, G and C), 16 ambiguity characters and five special tokens. The ambiguity codes extend beyond conventional IUPAC representations and were empirically determined by identifying all unique non-standard characters present in our training dataset. The special tokens serve distinct roles: [PAD]is used for padding sequences within a batch, [UNK] denotes unknown or out-of-vocabulary characters, [MASK] is used during masked language modelling, [CLS] is prepended to each sequence to aggregate global context and [SEP] separates input segments when required for downstream tasks.

The model was pretrained using a masking strategy adapted from BERT-style language modelling[19], combining both token-level and subsequence-level masking to encourage learning of local and higher-order sequence patterns. For token-level masking, 15% of tokens in each input sequence were selected at random. Of these, 80% were replaced with the `[MASK]` token, 10% were substituted with a random token from the vocabulary and 10% were left unchanged. In parallel, a subsequence masking strategy was employed, in which contiguous spans of four to eight tokens were randomly selected and masked as a group. This span-based approach draws on principles from SpanBERT[52] and is intended to promote learning of biologically meaningful RNA motifs, which often appear as short subsequences, but did not observe any downstream performance improvement.

To train NucleicBERT, we obtained ncRNA sequences from the MARS database[53]. It contains ~1.7 billion sequences in FASTA format, but as we wanted to train our model only on ncRNA sequences, we extracted all sequences with the ncRNA keyword in their description and obtained ~30 million sequences. Overall, 80% of this data was used in training, while the remaining was used in validation. We trained this 404-million-parameter model using 192 A100-40-GB GPUs with a linear learning rate schedule and a batch size of 16. To ensure stable gradients and prevent overfitting, we maintained a dropout rate of 0.1 and clipped gradient values to 0.001. Each epoch of training required approximately 1 h and 20 min, and the model was trained for 300 epochs. Pretraining accuracy plateaued at 83.1% on the validation dataset. Initially, we also tested a smaller 101-million-parameter model, but the pretraining accuracy stagnated at 52%, pointing us towards a bigger model to capture complex patterns in RNA sequence data.

For analysing embeddings obtained from our model, we used the PHATE dimensionality reduction technique. Unlike clustering-focused approaches such as t-SNE[54] or PCA[55], PHATE preserves continuous trajectories and branching structures through a diffusion-based framework that simulates random walks across data manifolds to capture global relationships.

PHATE constructs embeddings through a multi-step process: computing local similarities with α-decaying kernels, creating Markovian transition matrices and raising these matrices to successive powers to simulate longer diffusion processes. This diffusion operation denoises data while learning global relationships between distant points. Rather than using diffusion coordinates directly, PHATE computes an information-geometric 'potential distance' between probability distributions, making it particularly effective at preserving biological transitions.

Secondary-structure architecture and dataset

For secondary-structure prediction, we developed a binary classification framework that determines whether nucleotide pairs form base-pairing interactions. The approach involves attaching a bottleneck residual network architecture to the pretrained NucleicBERT model, which processes pairwise nucleotide representations generated through outer concatenation of the transformer embeddings. Specifically, the representation for nucleotide pair (i, j) is constructed by concatenating the embedding vectors of positions i and j from NucleicBERT's final layer.

The prediction network generates a symmetric probability matrix indicating base-pairing likelihood for each nucleotide pair in the sequence. We optimized the model using a binary cross-entropy loss, computing gradients exclusively for the upper triangular portion of the matrix to account for base-pairing symmetry constraints. Training proceeded for 50 epochs using a learning rate of 1×10^{-5}, and we trained

all model parameters simultaneously without implementing gradual unfreezing strategies.

We converted the predicted probability matrix into discrete secondary structures using a greedy decoding algorithm. This iterative process selects the highest-probability nucleotide pair as base-paired, then removes conflicting pairs from consideration in subsequent iterations. The algorithm enforces biological constraints by excluding non-Watson–Crick pairings and preventing formation of hairpin loops shorter than four nucleotides (constraint $|i-j| \geq 4$). We determined the optimal probability threshold through validation set optimization to achieve balanced pairing ratios.

Performance evaluation incorporated RNA structural flexibility by accepting near-correct predictions as valid. Following established evaluation protocols[56], we considered predictions within one position of the true pairing as correct: for a reference pair (i,j), the predictions $(i \pm 1, j)$ and $(i, j \pm 1)$ were scored as accurate. Final F1 scores represent the average of individual structure-level F1 calculations across the entire test set. Several benchmark datasets are used in this study:

(1) The RNAStrAlign[57] dataset comprises 37,149 RNA structures from eight distinct RNA families, with sequence lengths ranging from approximately 100 to 3,000 base pairs.
(2) The ArchiveII[26] dataset consists of 3,975 RNA structures from ten RNA families, with lengths ranging from approximately 100 to 2,000 base pairs. We apply a cutoff of 600 base pairs to get the standard dataset used in tools such as UFold and RNAErnie.
(3) bpRNA-1m[27], contains a large number of highly similar sequences, thus a sequence identity cutoff of 80% is applied. The dataset is randomly split into three subsets: TR0 (10,814 structures) for training, TV0 (1,300 structures) for validation and TS0 (1,305 structures) for testing.

The model was trained on the RNAStrAlign data and TR0 data combined. We tested the model on ArchiveII600 and TS0 datasets.

Tertiary-structure prediction architecture and dataset

To use the representations learned during the pretraining, we fine-tuned our model to predict distance maps and contact maps for a given RNA sequence.

Distance maps represent spatial pairwise inter-residue interactions, providing an RNA's translationally and rotationally invariant topological representation. For a sequence of length L, a distance map is a matrix $(L \times L)$ representation that delineates pairwise interactions between atoms within a bio-molecular structure. The elements of the matrix correspond to specific residues, and the entry at the intersection of row i and column j indicates the Euclidean distance between them.

Contact maps are a binary representation derived from these distance maps, where a threshold distance is applied to determine whether two nucleotides are in contact. If the distance between nucleotides is below the defined threshold (we used 8 Å) (ref. 15), they are considered to be in contact (represented as 1 or a marked point), while distances above the threshold are considered non-contacts (represented as 0 or empty space). This simplified binary representation captures essential structural information while reducing noise and computational complexity.

The PDB database contains merely 8,446 (as of January 2025) annotated tertiary RNA structures, reflecting the scarcity of such data. Further, a large proportion of these structures are hybrids, have poor resolutions, have small lengths and have high sequence similarity. An unfiltered dataset with redundancy, obtained from the PDB database, can impact the generalizability of the model. We used the BGSU RNA 3D representative dataset (release 3.368, January 2025[35]) as our starting point. We processed this dataset by filtering for structures with lengths between 32 and 1,024 nucleotides, removing redundancy using CD-HIT-EST[58] with an 80% sequence identity threshold, and ensuring structural quality. For each selected structure, we extracted the sequence, resolution and chain information from mmCIF files, creating a curated dataset suitable for model training. We also used NucleoSeeker[34] to create a non-redundant dataset for downstream training using the same steps as described above. Both datasets were used separately for training, and they showed similar performance.

To prepare distance maps, the distances between the heavy atoms of each residue are calculated with every other residue. These distances are assigned into 20 classes such that

$$C(d) = \begin{cases} 0, & \text{if } d < 2 \\ \lfloor d - 2 \rfloor + 1, & \text{if } 2 \leq d \leq 20, \\ 19, & \text{if } d > 20. \end{cases}$$

Many structures in the PDB database have missing residues, which can cause problems in the training because there will be a mismatch in the length of sequences and the shape of the distance map. To solve this, we use PDBFixer[59] to add missing residues and atoms. Further, we use the Chemical Component Dictionary from PDB to replace the remaining non-standard residues.

Last, we remove the trivial distances from the distance maps because they originate from the backbone and do not contain meaningful information. We remove four off-diagonals above and below the main diagonal, including the main diagonal. Data prepared in this manner enable the model to learn long-range interactions.

For the downstream supervised model, we use image segmentation techniques. The attention weights representation from the pretrained model is fed into a ResNet[60] model, which classifies each pixel (element) of the attention weights into the 20 categories.

Contact map prediction presents unique challenges that differentiate it from conventional computer vision tasks. While distance maps can be interpreted as 2D images, making contact prediction superficially analogous to pixel-level image labelling, this analogy is limited and should be applied with caution. Some techniques from image labelling may be transferable, but there are several important distinctions.

First, in typical image classification or segmentation tasks, input images are often resized to a fixed dimension. By contrast, contact maps must preserve the native dimensions of the RNA sequence, as predictions are required for every pair of residues. Second, the input features for contact prediction are substantially more complex, incorporating both sequential and pairwise features, unlike the raw pixel intensities used in standard image tasks. Third, the label distribution in contact maps is highly imbalanced: the proportion of positive contacts is typically less than 2%, creating a considerable challenge for model training[61].

Splice-site prediction architecture and dataset

We formulated splice-site prediction as a binary classification problem, where the model must distinguish between authentic splice sites and non-functional sequences. The evaluation was conducted using carefully curated multi-species datasets that have become standard benchmarks in the community.

For splice site prediction evaluation, we employed the benchmark datasets established in the Spliceator[36] study, which have since been adopted by the RiNALMo framework[21]. These datasets represent a gold standard for multi-species splice-site prediction assessment owing to their rigorous curation and phylogenetic diversity.

The benchmark datasets encompass four evolutionarily diverse species: zebrafish (*Danio rerio*), fruit fly (*Drosophila melanogaster*), nematode worm (*Caenorhabditis elegans*) and thale cress (*Arabidopsis thaliana*). Each benchmark contains 10,000 splice-site sequences and 10,000 non-splice-site sequences, maintaining a balanced 1:1 ratio to

prevent class imbalance bias. The datasets include both canonical and non-canonical splice sites, with non-canonical splice sites of 307 in human, 85 in zebrafish, 120 in fly, 67 in worm and 122 in plant samples.

The original Spliceator datasets were constructed from high-quality genomic sequences obtained from the Ensembl database release 87, with splice sites extracted and flanked by ±300 nucleotide environments to provide sufficient sequence context. A verification process ensured that no sequences containing undetermined nucleotides (noted as 'N') were included in the datasets. Each sequence spans 600 nucleotides total, with the GT (donor) or AG (acceptor) dinucleotide positioned at the central location (positions 301–302).

The negative subset construction employed a heterogeneous approach, incorporating three categories of non-splice-site sequences: randomly selected exon regions, randomly selected intron regions and false-positive sequences containing GT or AG dinucleotides that do not correspond to authentic splice sites. This heterogeneous negative dataset design (termed GS_1 in the Spliceator methodology) was shown to achieve superior performance compared with homogeneous negative datasets containing only false-positive sequences.

The benchmark datasets underwent rigorous quality control procedures on the basis of the G3PO+ multi-species benchmark methodology. Sequences were classified into 'confirmed' (error-free) and 'unconfirmed' (containing potential gene prediction errors) categories through expert-guided comparative sequence analysis. Only 'confirmed' sequences were retained in the final benchmarks to ensure biological authenticity of the splice sites.

To prevent sequence redundancy and potential overfitting, duplicate sequences were systematically removed from each dataset. Sequence similarity analysis revealed that the majority (>90%) of sequences in each benchmark share between 20% and 30% identity for full-length sequences, with higher conservation (20–60% identity) observed in the immediate splice-site vicinity.

For this downstream task, we fine-tuned NucleicBERT using a two-layer sequence-level classification head that processes the model's contextual embeddings to make binary predictions. The fine-tuning process was performed separately for donor and acceptor splice sites to optimize performance for each specific recognition task, following established protocols from both the Spliceator and RiNALMo methodologies.

Shuffled sequence detection architecture and dataset

We assessed NucleicBERT's capacity to identify authentic RNA sequences by systematically introducing controlled perturbations through nucleotide shuffling. Using the same dataset employed for secondary-structure prediction training and testing, we generated shuffled variants with perturbation levels ranging from 0% to 100% in 5% increments. At each shuffle percentage, we measured the fraction of sequences that the model still classified as authentic RNA. In this study, we use the embeddings of the pretrained model, which are passed through a three-layer perceptron to predict if a given sequence is shuffled or not.

To investigate whether the model preferentially attends to structurally critical regions, we performed targeted perturbation analysis focusing exclusively on Watson–Crick base-paired positions identified from known secondary structures. This approach enabled us to assess the model's understanding of the hierarchical constraints that govern RNA folding.

To establish whether the observed performance thresholds reflect genuine biological constraints rather than training artefacts, we analysed sequence variation patterns across natural RNA families. We extracted seed MSAs for 4,178 RNA families from the Rfam database[62] (release 15.0), focusing on families containing at least five sequences to ensure statistical reliability. For each family, we identified a representative sequence by optimizing Levenshtein similarity across all family members and calculated the maximum sequence variation tolerance within each family using normalized edit distance.

Coevolution analysis from single sequences

The pretrained NucleicBERT (no fine-tuning) was probed for coevolution-like couplings using the MLI score. For an input sequence $\mathbf{x} = (x_1, \ldots, x_L)$, MLI quantifies how much the model's predictive log-probability for the native nucleotide at position i changes when position j is additionally masked

$$I_{j\to i}(\mathbf{x}) = \log p_\theta\left(x_i|\mathbf{x}_{\setminus i}\right) - \log p_\theta\left(x_i|\mathbf{x}_{\setminus i,\setminus j}\right), \tag{3}$$

where p_θ is the softmax over the 25-token vocabulary at the masked position and $\mathbf{x}_{\setminus i}$ denotes $\mathbf{x}$ with position i replaced by `[MASK]`. The raw $L \times L$ matrix is symmetrized with zero diagonal,

$$M_{ij}(\mathbf{x}) = \frac{1}{2}\left(I_{j\to i}(\mathbf{x}) + I_{i\to j}(\mathbf{x})\right), \quad M_{ii} = 0. \tag{4}$$

This formulation generalizes the single-mask perplexity used in protein language models[22] to a paired-position perturbation.

Per-family aggregation. For each Rfam family we computed $M(\mathbf{x}^{(n)})$ for every aligned sequence $\mathbf{x}^{(n)}$ in the post-filtering MSA and averaged over the alignment, $\bar{M} = \mathrm{nanmean}_n M(\mathbf{x}^{(n)})$, projecting per-sequence indices back to alignment columns and treating gap rows/columns as `NaN` so they are ignored by the column-wise mean. The MSA enters only at this averaging step and the model itself sees one sequence at a time, so MLI remains an alignment-free, single-sequence statistic; averaging is purely to obtain a family-level estimate comparable to classical MSA-based couplings.

Average product correction. To remove the dominant column- and row-conservation contribution to coupling magnitudes, we apply 'average product correction'[14] to $\bar{M}$

$$M_{ij}^{\mathrm{APC}} = \bar{M}_{ij} - \frac{\bar{M}_{i\cdot}\,\bar{M}_{\cdot j}}{\bar{M}_{\cdot\cdot}}, \tag{5}$$

where $\bar{M}_{i\cdot}$, $\bar{M}_{\cdot j}$ and $\bar{M}_{\cdot\cdot}$ are means over the upper triangle. Average product correction is applied to MLI (when averaged over MSA, not for single-sequence predictions) and MI, and DCA Frobenius norms are reported uncorrected per convention.

Baselines. MI is computed from one-/two-site frequencies on the MSA with a 0.5 pseudocount and a sequence-reweighting threshold of 0.8 identity for M_{eff}. DCA is the Frobenius norm of the 4×4 inverse-covariance block per pair under a mean-field approximation[14]. The random null is a deterministic seed feeding a symmetric Gaussian $L \times L$ draw scored with the same metrics.

Metrics. All metrics are computed on the upper triangle ($i<j$). Given a reference set of true contacts (Rfam canonical `SS_cons` Watson–Crick pairs), TopL is the precision among the top-L predicted pairs, Top2L is the precision among the top-$2L$ pairs, AUPRC is the area under the precision-recall curve, and EF = TopL/p_0, where $p_0 = |\mathrm{contacts}|/\binom{L}{2}$ is the per-family random base rate. EF normalizes across families with very different base rates and is the metric used in Fig. 5, other metrics are reported per family in Supplementary Table 8.

For family selection, we ran the analysis on 14 Rfam families spanning the a range of sequence lengths L and MSA depths M_{eff}, requiring at least 50 post-filtering sequences, with a gap fraction ≤0.5, making sure that MSA depths are sufficient for MI and DCA to provide fair baselines.

Fitness prediction architecture and dataset

We use the CPEB3 mutation dataset[39], which represents a comprehensive mutagenesis study of RNA sequences that serve as substrates for

cytoplasmic polyadenylation element-binding protein 3 (CPEB3), an RNA-binding protein that regulates gene expression through control of mRNA polyadenylation. This dataset contains fitness landscape data from a high-throughput cleavage assay where systematically mutagenized RNA sequences were evaluated for their susceptibility to cleavage by CPEB3. Each RNA variant was tested across three independent biological replicates, with fitness calculated as the ratio of cleaved to uncleaved RNA molecules, providing a quantitative measure of how effectively each sequence serves as a CPEB3 substrate. The rigorous experimental methodology, including multiple biological replicates and standardized cleavage assays, ensures high-quality fitness measurements suitable for downstream computational analysis and machine learning models designed to predict RNA substrate fitness from sequence information.

For fitness prediction, we employed a task-specific regression head that processes the pooled embeddings from NucleicBERT's final transformer layer. The regression architecture consists of a three-layer perceptron that maps the 1,024-dimensional RNA sequence representations to scalar fitness predictions. We implemented an 80:20 train–test split, ensuring that the evaluation dataset contained sequences spanning the full mutational spectrum while maintaining statistical power for performance assessment.

To establish a biologically informed baseline that captures the assumption that similar fitness values arise from sequences with similar mutation patterns, we develop a ball-query model, which is essentially a *k*-nearest-neighbours model. It measures the distance between sequences using the symmetric distance between mutation sets, and for each target sequence, it finds all reference sequences within a specified cutoff radius. The fitness value is the average fitness of all sequences within the radius.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

Pretraining data: the RNA sequences used for pretraining were obtained from the publicly available MARS database[55], specifically extracting ~30 million ncRNA sequences with the ncRNA keyword. Benchmark datasets: all evaluation datasets used in this study are publicly available: Secondary-structure prediction: ArchiveII[26], bpRNA-1m[27] and RNAStrAlign[59]. Tertiary-structure prediction: PDB structures processed through NucleoSeeker[34] and BGSU RNA 3D representative dataset[35] (release 3.368, January 2025). Splice-site prediction: Spliceator[36] benchmark datasets for zebrafish, fly, worm and plant species. Fitness prediction: CPEB3 mutation dataset as in ref. 39. Shuffle detection: ArchiveII[26], bpRNA-1m[27], RNAStrAlign[57] and Rfam database release 15.0[62]. Coevolution analysis: seed alignments and `SS_cons` contact annotations for 14 Rfam families from the Rfam database release 15.0[62]. Source data are provided with this paper.

Code availability

The code used in this study is available via GitHub at https://github.com/KIT-MBS/NucleicBERT (ref. 63). Model weights are available via Zenodo at https://doi.org/10.5281/zenodo.16989562 (ref. 64). Further processed datasets generated for this study are available upon reasonable request from the corresponding author.

References

1. Schuntermann, D. B., Jaskolowski, M., Reynolds, N. M. & Vargas-Rodriguez, O. The central role of transfer RNAs in mistranslation. *J. Biol. Chem.* **300**, 107679 (2024).
2. Statello, L., Guo, C.-J., Chen, L.-L. & Huarte, M. Gene regulation by long non-coding RNAs and its biological functions. *Nat. Rev. Mol. Cell Biol.* **22**, 96–118 (2021).
3. Mattick, J. S. et al. Long non-coding RNAs: definitions, functions, challenges and recommendations. *Nat. Rev. Mol. Cell Biol.* **24**, 430–447 (2023).
4. Saw, P. E. & Song, E. Advancements in clinical RNA therapeutics: present developments and prospective outlooks. *Cell Rep. Med.* **5**, 101555 https://doi.org/10.1016/j.xcrm.2024.101555 (2024).
5. Tani, H. Recent advances and prospects in RNA drug development. *Int. J. Mol. Sci.* **25**, 12284 (2024).
6. Wu, K. J. Y. et al. An antibiotic preorganized for ribosomal binding overcomes antimicrobial resistance. *Science* **383**, 721–726 (2024).
7. Gauto, D. F. et al. Integrated NMR and cryo-EM atomic-resolution structure determination of a half-megadalton enzyme complex. *Nat. Commun.* **10**, 2697 (2019).
8. Lengyel, J., Hnath, E., Storms, M. & Wohlfarth, T. Towards an integrative structural biology approach: combining Cryo-TEM, X-ray crystallography, and NMR. *J. Struct. Funct. Genomics* **15**, 117–124 (2014).
9. Ma, H., Jia, X., Zhang, K. & Su, Z. Cryo-EM advances in RNA structure determination. *Signal Transduct. Target. Ther.* **7**, 58 (2022).
10. Torrisi, M., Pollastri, G. & Le, Q. Deep learning methods in protein structure prediction. *Comput. Struct. Biotechnol. J.* **18**, 1301–1310 (2020).
11. Qin, Y. et al. Deep learning methods for protein structure prediction. *MedComm Futur. Med.* **3**, e96 (2024).
12. Weigt, M., White, R. A., Szurmant, H., Hoch, J. A. & Hwa, T. Identification of direct residue contacts in protein–protein interaction by message passing. *Proc. Natl Acad. Sci. USA* **106**, 67–72 (2009).
13. Schug, A., Weigt, M., Onuchic, J. N., Hwa, T. & Szurmant, H. High-resolution protein complexes from integrating genomic information with molecular simulation. *Proc. Natl Acad. Sci. USA* **106**, 22124–22129 (2009).
14. Morcos, F. et al. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proc. Natl Acad. Sci. USA* https://doi.org/10.1073/pnas.1111471108 (2011).
15. De Leonardis, E. et al. Direct-coupling analysis of nucleotide coevolution facilitates RNA secondary and tertiary structure prediction. *Nucleic Acids Res.* **43**, 10444–10455 (2015).
16. Sun, S., Wang, W., Peng, Z. & Yang, J. RNA inter-nucleotide 3D closeness prediction by deep residual neural networks. *Bioinformatics* **37**, 1093–1098 (2021).
17. Zerihun, M. B., Pucci, F. & Schug, A. CoCoNet–boosting RNA contact prediction by convolutional neural networks. *Nucleic Acids Res.* **49**, 12661–12672 (2021).
18. Taubert, O. et al. RNA contact prediction by data efficient deep learning. *Commun. Biol.* **6**, 913 (2023).
19. Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proc. 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* Vol. 1 (eds Burstein, J., Doran, C. & Solorio, T.) 4171–4186 (Association for Computational Linguistics, 2019); https://aclanthology.org/N19-1423
20. Chen, J. et al. Interpretable RNA foundation model from unannotated data for highly accurate RNA structure and function predictions. Preprint at https://doi.org/10.48550/arXiv.2204.00300 (2022).
21. Penić, R. J., Vlašić, T., Huber, R. G., Wan, Y. & Šikić, M. RiNALMo: general-purpose RNA language models can generalize well on structure prediction tasks. *Nat. Commun.* **16**, 5671 (2025).
22. Lin, Z. et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science* **379**, 1123–1130 (2023).

23. Gigante, S., Charles, A. S., Krishnaswamy, S. & Mishne, G. Visualizing the PHATE of neural networks. In *Proc. 33rd International Conference on Neural Information Processing Systems* 1842–1853 (Curran, 2019).
24. Zhong, Z. & Andreas, J. Algorithmic capabilities of random transformers. In *38th Annual Conference on Neural Information Processing Systems* https://openreview.net/pdf?id=plH8gW7tPQ (2024).
25. Li, S., Tong, Y., Wang, H. & Hu, T. Transformers are born biased: structural inductive biases at random initialization and their practical consequences. Preprint at https://doi.org/10.48550/arXiv.2602.05927 (2026).
26. Sloma, M. F. & Mathews, D. H. Exact calculation of loop formation probability identifies folding motifs in RNA secondary structures. *RNA* **22**, 1808–1818 (2016).
27. Danaee, P. et al. bpRNA: large-scale automated annotation and analysis of RNA secondary structure. *Nucleic Acids Res.* **46**, 5381–5394 (2018).
28. Lorenz, R. et al. ViennaRNA Package 2.0. *Algorithms Mol. Biol.* **6**, 26 (2011).
29. Sato, K., Akiyama, M. & Sakakibara, Y. RNA secondary structure prediction using deep learning with thermodynamic integration. *Nat. Commun.* **12**, 941 (2021).
30. Chen, X., Li, Y., Umarov, R., Gao, X. & Song, L. RNA Secondary structure prediction by learning unrolled algorithms. In *International Conference on Learning Representations 2020* https://openreview.net/pdf?id=S1eALyrYDH (2020).
31. Akiyama, M. & Sakakibara, Y. Informative RNA base embedding for RNA structural alignment and clustering by deep representation learning. *NARGenom. Bioinform.* **4**, lqac012 (2022).
32. Wang, N. et al. Multi-purpose RNA language modelling with motif-aware pretraining and type-guided fine-tuning. *Nat. Mach. Intell.* **6**, 548–557 (2024).
33. Lei, G. et al. CPU-GPU hybrid accelerating the Zuker algorithm for RNA secondary structure prediction applications. *BMC Genomics* **13**, S14 (2012).
34. Upadhyay, U., Pucci, F., Herold, J. & Schug, A. NucleoSeeker—precision filtering of RNA databases to curate high-quality datasets. *NAR Genom. Bioinform.* **7**, lqaf021 (2025).
35. Leontis, N. B. & Zirbel, C. L. Nonredundant 3D structure datasets for RNA Knowledge extraction and benchmarking. in *RNA 3D Structure Analysis and Prediction* (eds Leontis, N. & Westhof, E.) 281–298 (Springer, 2012).
36. Scalzitti, N. et al. Spliceator: multi-species splice site prediction using convolutional neural networks. *BMC Bioinformatics* **22**, 561 (2021).
37. Ji, Y., Zhou, Z., Liu, H. & Davuluri, R. V. DNABERT: pre-trained bidirectional encoder representations from transformers model for DNA-language in genome. *Bioinformatics* **37**, 2112–2120 (2021).
38. Chen, K. et al. Self-supervised learning on millions of primary RNA sequences from 72 vertebrates improves sequence-based RNA splicing prediction. *Brief. Bioinform.* **25**, bbae163 (2024).
39. Beck, J. D. et al. Predicting higher-order mutational effects in an RNA enzyme by machine learning of high-throughput experimental data. *Front. Mol. Biosci.* **9**, 893864 (2022).
40. Simonyan, K., Vedaldi, A. & Zisserman, A. Deep inside convolutional networks: visualising image classification models and saliency maps. Preprint at https://doi.org/10.48550/arXiv.1312.6034 (2014).
41. Sundararajan, M., Taly, A. & Yan, Q. Axiomatic attribution for deep networks. In *Proc. 34th International Conference on Machine Learning* Vol. 70 (eds Precup, D. & Teh, Y. W.) 3319–3328 (2017).
42. Rogers, A., Kovaleva, O. & Rumshisky, A. A primer in BERTology: what we know about how BERT works. *Trans. Assoc. Comput. Linguist.* **8**, 842–866 (2020).
43. Jawahar, G., Sagot, B. & Seddah, D. What does BERT Learn about the structure of language? In *Proc. 57th Annual Meeting of the Association for Computational Linguistics* (eds Korhonen, A., Traum, D. & Màrquez, L.) 3651–3657 (Association for Computational Linguistics, 2019).
44. Clauwaert, J., Menschaert, G. & Waegeman, W. Explainability in transformer models for functional genomics. *Brief. Bioinform.* **22**, bbab060 (2021).
45. Lambert, C. N. et al. Exploring the space of self-reproducing ribozymes using generative models. *Nat. Commun.* **16**, 7836 (2025).
46. Dalla-Torre, H. et al. Nucleotide transformer: building and evaluating robust foundation models for human genomics. *Nat. Methods* **22**, 287–297 (2025).
47. Vig, J. et al. BERTology meets biology: interpreting attention in protein language models. Preprint at https://doi.org/10.48550/arXiv.2006.15222 (2021).
48. Song, Z. et al. Attention-based multi-label neural networks for integrated prediction and interpretation of twelve widely occurring RNA modifications. *Nat. Commun.* **12**, 4011 (2021).
49. Brandes, N., Ofer, D., Peleg, Y., Rappoport, N. & Linial, M. ProteinBERT: a universal deep-learning model of protein sequence and function. *Bioinformatics* **38**, 2102–2110 (2022).
50. Alvarez, D. JUWELS Cluster and Booster: exascale pathfinder with modular supercomputing architecture at Juelich Supercomputing Centre. *J. Large-Scale Res. Fac.* **7**, A183 https://doi.org/10.17815/jlsrf-7-183 (2021).
51. Krause, D. & Thörnig, P. JURECA: modular supercomputer at Jülich Supercomputing Centre. *J. Large-Scale Res. Fac.* **4**, A132 https://doi.org/10.17815/jlsrf-4-121-1 (2018).
52. Joshi, M. et al. SpanBERT: improving pre-training by representing and predicting spans. *Trans. Assoc. Comput. Linguist.* **8**, 64–77 (2020).
53. Chen, K., Litfin, T., Singh, J., Zhan, J. & Zhou, Y. MARS and RNAcmap3: the master database of all possible RNA sequences integrated with RNAcmap for RNA homology search. *Genomics Proteomics Bioinformatics* **22**, qzae018 https://doi.org/10.1093/gpbjnl/qzae018 (2024).
54. van der Maaten, L. & Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **9**, 2579–2605 (2008).
55. Maćkiewicz, A. & Ratajczak, W. Principal components analysis (PCA). *Comput. Geosci.* **19**, 303–342 (1993).
56. Mathews, D. H. How to benchmark RNA secondary structure prediction accuracy. *Methods* **162–163**, 60–67 (2019).
57. Tan, Z., Fu, Y., Sharma, G. & Mathews, D. H. TurboFold II: RNA structural alignment and secondary structure prediction informed by multiple homologs. *Nucleic Acids Res.* **45**, 11570–11581 (2017).
58. Li, W. & Godzik, A. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics* **22**, 1658–1659 (2006).
59. Eastman, P. et al. OpenMM 4: a reusable, extensible, hardware independent library for high performance molecular simulation. *J. Chem. Theory Comput.* **9**, 461–469 (2013).
60. He, K., Zhang, X., Ren, S. & Sun, J. Deep residual learning for image recognition. Preprint at https://arxiv.org/abs/1512.03385 (2015).
61. Li, Z., Lin, Y., Elofsson, A. & Yao, Y. Protein contact map prediction based on ResNet and DenseNet. *BioMed Res. Int.* **2020**, e7584968 (2020).
62. Ontiveros-Palacios, N. et al. Rfam 15: RNA families database in 2025. *Nucleic Acids Res.* **53**, D258–D267 (2025).

63. Upadhyay, U. KIT-MBS/NucleicBERT: NucleicBERT. *GitHub* https://github.com/KIT-MBS/NucleicBERT (2026).
64. Upadhyay, U., Herold, J., Götz, M. & Schug, A. NucleicBERT: pre-trained model weights for RNA structure and function prediction. *Zenodo* https://doi.org/10.5281/zenodo.16989562 (2025).

Acknowledgements
We thank S. Kesselheim (JSC/FZJ), J. Wang (JSC/FZJ), C. Faber (JSC/FZJ) and A. Dorn (JSC/FZJ) for valuable discussions.

Author contributions
Conceptualization: U.U. and A.S. conceived and designed the study. Software: U.U. developed the main software package. J.H. and M.G. contributed code for the explainable AI components. Methodology: U.U., J.H. and M.G. provided the analytical framework and method validation. Formal analysis: all authors contributed to the formal analysis. Writing—original draft: all authors contributed to writing the original manuscript. Writing—review and editing: all authors reviewed and approved the final manuscript.

Funding
The authors gratefully acknowledge the Gauss Centre for Supercomputing e.V. (https://www.gauss-centre.eu) for funding this project by providing compute time through the John von Neumann Institute for Computing on the GCS Supercomputer JUWELS[50] and on the supercomputer JURECA[51] at Forschungszentrum Jülich under grants EatsRNA and PROFOUND. A.S. and J.H. recognize support by the Helmholtz Information and Data Science School for Health. M.G., A.S. and J.H. recognize support by the Helmholtz Association's Initiative and Networking Fund under the Helmholtz AI platform grant and the grant ZebraTwin. A.S. recognizes support by the Helmholtz Foundation Model Initiative of the Helmholtz Association under the grants PROFOUND and VirtualCell. The funders had no role in study design, data collection, analysis, decision to publish or preparation of the manuscript. Open access funding provided by Karlsruher Institut für Technologie (KIT).

Competing interests
The authors declare no competing interests.

Additional information
Extended data is available for this paper at https://doi.org/10.1038/s42256-026-01295-9.

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-026-01295-9.

Correspondence and requests for materials should be addressed to Alexander Schug.

Peer review information *Nature Machine Intelligence* thanks the anonymous reviewer(s) for their contribution to the peer review of this work.

https://doi.org/10.1038/s42256-026-01295-9

Extended Data Table 1 | Quantitative validation of PHATE embeddings

Setting	n	Pre-trained	Random Init	6-mer	Length	Length + ACGU%
A. All 7 classes	7000	0.870 ± 0.007	0.765 ± 0.020	0.300 ± 0.008	0.592 ± 0.057	0.746 ± 0.017
B. 5 classes, length $\in [50, 500]$	4102	0.852 ± 0.016	0.690 ± 0.012	0.405 ± 0.006	0.492 ± 0.015	0.677 ± 0.017
C. 5 classes, length-matched	230	0.653 ± 0.023	0.457 ± 0.030	0.308 ± 0.004	0.353 ± 0.035	0.367 ± 0.034

via kNN classification (k = 15, Euclidean distance, computed in the original embedding spaces, not on 2D projections). RNA family is predicted by majority vote among each sequence's 15 nearest neighbours; values are mean accuracy ± s.d. over 5 random splits. Length-only and length+ACGU% baselines test how much of each model's accuracy is recoverable from trivial sequence statistics. Setting C controls these statistics out by length-matching across classes.

nature portfolio

Corresponding author(s): Alexander Schug

Last updated by author(s): May 20, 2026

Reporting Summary

Nature Portfolio wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Portfolio policies, see our Editorial Policies and the Editorial Policy Checklist.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a	Confirmed	
☐	☒	The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement
☐	☒	A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
☒	☐	The statistical test(s) used AND whether they are one- or two-sided *Only common tests should be described solely by name; describe more complex techniques in the Methods section.*
☒	☐	A description of all covariates tested
☒	☐	A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
☒	☐	A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
☒	☐	For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted *Give P values as exact values whenever suitable.*
☒	☐	For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
☒	☐	For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
☒	☐	Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	Please see the data availability statement in the main text, the supplementary information and in particular https://doi.org/10.5281/zenodo.16989562 .
Data analysis	All software code used for data analysis is available at https://github.com/KIT-MBS/NucleicBERT with dependency information provided in the repository.

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Portfolio guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A description of any restrictions on data availability
- For clinical datasets or third party data, please ensure that the statement adheres to our policy

Pretraining data: The RNA sequences used for pretraining were obtained from the publicly available MARS database, from which ~30 million non-coding RNA sequences annotated with the ncRNA keyword were extracted. Benchmark datasets used in this study are publicly available: ArchiveII, bpRNA-1m, and RNAStrAlign for secondary structure prediction; PDB structures processed through NucleoSeeker and the BGSU RNA 3D representative dataset (release 3.368, January 2025) for tertiary structure prediction; Spliceator benchmark datasets for zebrafish, fly, worm, and plant species for splice-site prediction; the published CPEB3 mutation dataset for fitness prediction; and ArchiveII, bpRNA-1m, RNAStrAlign, and Rfam release 15.0 for shuffle detection. Model weights are available at https://doi.org/10.5281/zenodo.16989562 . Further processed datasets generated for this study are available from the corresponding author upon reasonable request.

Research involving human participants, their data, or biological material

Policy information about studies with human participants or human data. See also policy information about sex, gender (identity/presentation), and sexual orientation and race, ethnicity and racism.

Reporting on sex and gender	n/a
Reporting on race, ethnicity, or other socially relevant groupings	n/a
Population characteristics	n/a
Recruitment	n/a
Ethics oversight	n/a

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/documents/nr-reporting-summary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	No statistical method was used to predetermine sample size. Sample sizes were defined by the publicly available datasets used in this computational study.
Data exclusions	No ad hoc data exclusions were applied. Data filtering followed predefined criteria described in the main text and and Supplementary Information
Replication	This study did not include newly performed biological experiments. Replication information for published datasets is described in the main text and Supplementary Information.
Randomization	Where applicable, datasets were randomly split for training, validation, and testing, as described in the main text and Supplementary Information.
Blinding	Blinding was not applicable because this was a computational study using predefined datasets and automated evaluation.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
☒	☐ Antibodies
☒	☐ Eukaryotic cell lines
☒	☐ Palaeontology and archaeology
☒	☐ Animals and other organisms
☒	☐ Clinical data
☒	☐ Dual use research of concern
☒	☐ Plants

Methods

n/a	Involved in the study
☒	☐ ChIP-seq
☒	☐ Flow cytometry
☒	☐ MRI-based neuroimaging

Plants

Seed stocks	n/a
Novel plant genotypes	n/a
Authentication	n/a