A Long-Context Generative Foundation Model Deciphers RNA Design Principles

Yanjie Huang[†1], Guangye Lv[†2], Anyue Cheng[7], Wei Xie[8], Mengyan Chen[1], Xinyi Ma[4], Yijun Huang[3], Yueyang Tang[6], Qingya Shi[9], Zining Wang[10], Junxi Wang[1], Yunpeng Xia[5], Lu Zhao[12], Yifang Cai[12], Jack X. Chen[1], and Shuangjia Zheng[*1,11]

[1]Global Institute of Future Technology, Shanghai Jiao Tong University
[2]School of Cyber Science and Engineering, Shanghai Jiao Tong University
[3]School of Computer Science, Shanghai Jiao Tong University
[4]School of Biomedical Engineering, Shanghai Jiao Tong University
[5]School of Automation and Intelligent Sensing, Shanghai Jiao Tong University
[6]Global College, Shanghai Jiao Tong University
[7]School of Management, Fudan University
[8]School of Data Science, Fudan University
[9]School of Medicine, Tsinghua University
[10]School of Computer Science and Technology, University of Science and Technology of China
[11]School of Artificial Intelligence, Shanghai Jiao Tong University
[12]Lingang Laboratory

[†]These authors contributed equally to this work. [*]Corresponding author.

Abstract

Programmable design of RNA sequences with defined functions remains a central challenge in biology. Despite recent advances, existing RNA generative models lack robust controllable design capabilities and are constrained by short context windows, limiting their capacity to model the complex evolutionary manifold of full-length transcripts. Here, we introduce EVA (Evolutionary Versatile Architect), a long-context generative RNA foundation model trained on over 114 million full-length RNA sequences spanning the full breadth of evolutionary diversity. By integrating a Mixture-of-Experts architecture with an 8,192-token context window, EVA learns a unified, evolution-consistent representation of RNA sequence space that captures both global transcript structure and fine-grained functional elements. EVA establishes a unified framework for a broad spectrum of RNA design tasks within a single architecture, ranging from mutational fitness prediction to context-aware sequence engineering. We demonstrate EVA's versatility through the *de novo* design and targeted optimization of diverse RNA classes, including tRNAs, aptamers, and CRISPR guide RNAs, as well as mRNA and circular RNA therapeutics. In systematic evaluations, EVA achieves state-of-the-art performance across seven of nine established benchmarks, improving structural modeling accuracy by up to an order of magnitude compared to existing approaches. EVA is fully open-source at the following repository: https://github.com/GENTEL-lab/EVA.

[*]Email: shuangjia.zheng@sjtu.edu.cn

1 Introduction

RNA is a key molecular entity that performs essential functions within cells. It carries genetic instructions (mRNA)[1], folds into complex structures to carry out cellular roles (tRNA, rRNA)[2], catalyzes reactions (ribozymes)[2], and regulates gene expression (miRNA, lncRNA, snoRNA)[3,4]. This functional diversity positions RNA at the core of life process, driving evolutionary adaptation and cellular complexity[5]. Consequently, the programmable design of RNA has become a central frontier for both decoding biological logic and engineering next-generation medicines.

The recent expansion of large-scale RNA sequence resources has therefore motivated RNA foundation models that aim to capture these evolution-imposed regularities from data and transfer them across tasks[6–12]. Much of the field has so far emphasized encoder-only, MLM-trained architectures that are particularly effective for representation learning and inference, whereas design-oriented goals are more naturally served by decoder-only generative modeling that can synthesize sequences with explicit control over context and constraints (Supplementary Table S1). Although early decoder-based RNA generators have begun to explore this direction, they typically remain limited in regime, either focusing on selected functional RNA classes [13–15] or operating with context lengths that are short relative to full transcripts [10,16,17], and the effective training signal can be further diluted by corpus heterogeneity, including datasets that interleave RNA and DNA. Moreover, in contrast to genome foundation models that have demonstrated scalability in the DNA domain [18,19], RNA modeling has yet to exhibit comparable scaling behavior, likely due to weaker sequence conservation and the fragmented nature of available RNA datasets. These challenges motivate the development of scalable, long-context generative RNA foundation models capable of capturing the complex evolutionary manifold of full-length transcripts to enable controllable design across diverse RNA classes and taxa.

Herein, We present EVA (Evolutionary Versatile Architect), a long-context generative RNA foundation model trained on 114 million full-length RNA sequences (231.3B nucleotides) spanning structural, regulatory, coding, and viral RNAs across the tree of life. Built on a 1.4B-parameter decoder-only Transformer with a Mixture-of-Experts backbone and an 8,192-token context window, EVA jointly optimizes autoregressive generation and masked infilling, enabling both *de novo* sequence synthesis and context-constrained redesign. Explicit conditioning on RNA type and taxonomic lineage allows it to disentangle universal RNA grammar from lineage-specific adaptations.

Across mutation-effect prediction, conditional generation, and functional RNA design tasks, EVA consistently outperforms prior RNA foundation models, supporting controllable design across 11 major RNA classes without task-specific fine-tuning. It achieves state-of-the-art zero-shot fitness prediction for ncRNAs and mRNAs, and extends to DNA and protein in exploratory settings. Sampling from its evolution-consistent RNA manifold yields sequences that are structurally plausible yet evolutionarily novel, with a 10-fold improvements in structural modeling accuracy. To support the community, we release the full training corpus (OpenRNA v1), EVA model checkpoints at all scales, detailed training protocols, and an interactive web server. By integrating scalable generative modeling, high-performance predictive scoring, and interpretable latent representations within a single framework, EVA provides a foundation for systematic RNA exploration and design across evolutionary diversity.

OpenRNA v1 dataset: https://huggingface.co/datasets/GENTEL-Lab/OpenRNA-v1-114M

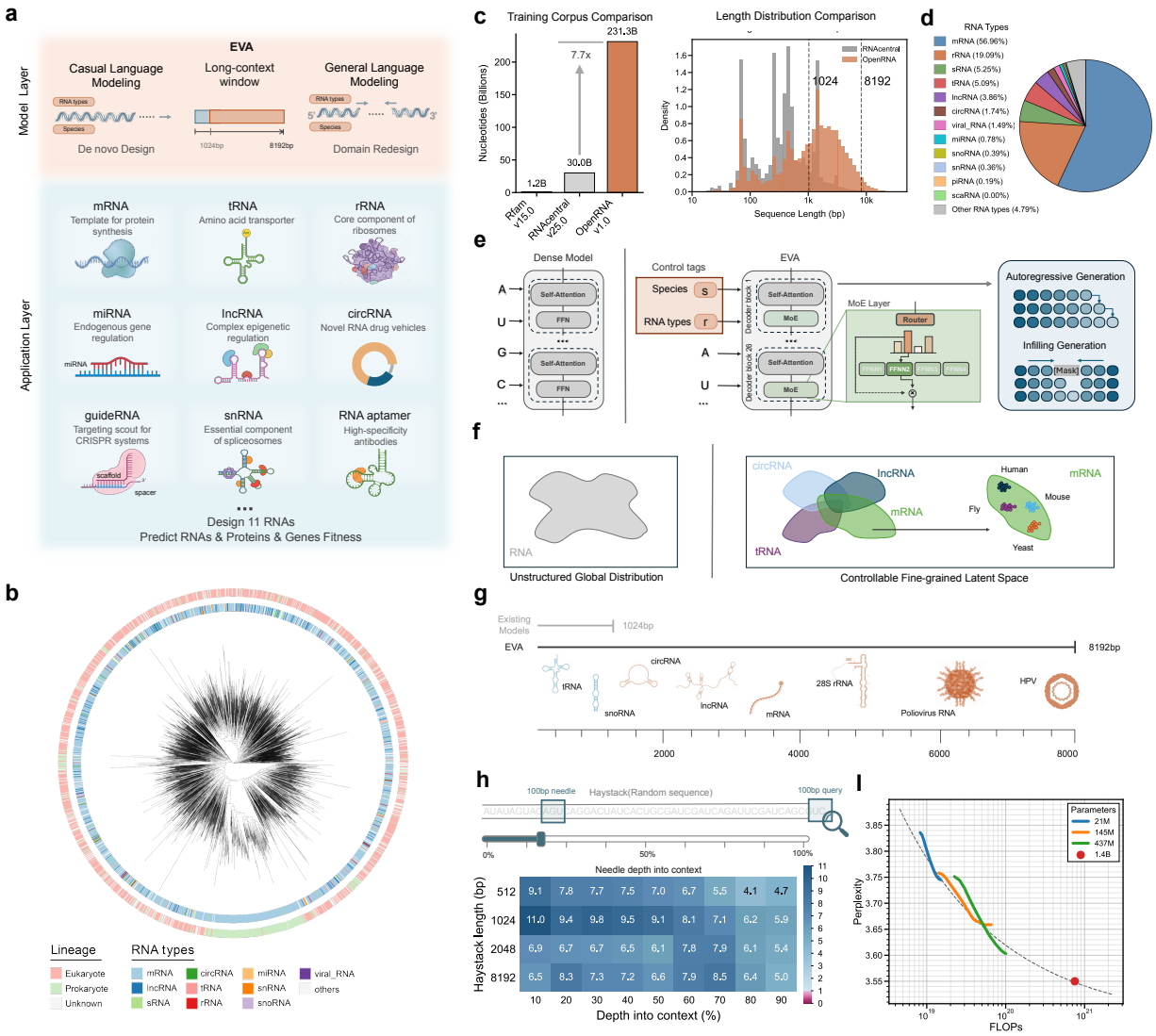

Figure 1: **Overview of EVA data, model, and training.**
(a) Overview of the EVA model capabilities, and representative applications. (b) OpenRNA v1 comprises 114M curated full-length RNA sequences spanning all domains of life;Phylogenetic tree construction of training data. (c) Comparison of training data scale with prior work (left), and sequence-length distributions together with the number of full-length RNAs covered by different context windows (right). (d) RNA type distribution in OpenRNA v1. (e) EVA model architecture and key differences from prior RNA language models: Mixture-of-Experts (MoE) architecture, support for both CLM and GLM generation, and conditioning on RNA type and taxonomic labels. (f) Illustration of the unstructured global distribution learned by conventional RNA LMs versus the controllable fine-grained latent space learned by EVA. (g) Comparison of context window lengths between EVA (8,192 bp) and prior RNA language models (1,024 bp), highlighting broader RNA design coverage enabled by long context. (h) Long-context "needle-in-a-haystack" evaluation demonstrating EVA's ability to recall long-range dependencies under an 8,192-token context window. (i) Scaling behavior showing decreasing perplexity with increasing model size, indicating that larger models better capture RNA sequence grammar.

2 Results

2.1 EVA model architecture, training procedures, data, and evaluations

EVA scales to 1.4 billion parameters to unify the previously disparate domains of non-coding RNA (ncRNA) and messenger RNA (mRNA) modeling. Equipped with an 8,192-token context window, EVA serves as a versatile engine that supports both Causal Language Modeling (CLM) and Generalized Language Modeling (GLM), enabling controllable generation conditioned on RNA family and taxonomic lineages (Figure 1A). To power this system, we introduce OpenRNA v1, a comprehensive RNA atlas totaling 114 million curated full-length sequences. Spanning diverse taxonomic groups, the dataset exhibits a broad distribution of sequences across the tree of life (Figure 1B), with phylum-level coverage across both prokaryotic and eukaryotic lineages (Figure S1).

To capture the hierarchical grammar of full-length RNAs, we adopted an 8,192-token context window, an 8-fold increase over conventional RNA models, expanding the designable RNA space (Figure 1C). This extended context enables the model to leverage our large-scale corpus effectively, capturing long-range dependencies across diverse architectures and covering over 95% of known full-length transcripts (Figure 1D). The dataset integrates 15 distinct RNA categories, providing a comprehensive representation of the RNA landscape (Figure 1E).

To model this diversity, EVA employs a decoder-only Transformer with a Mixture-of-Experts (MoE) backbone (Figure 1F). MoE architectures have demonstrated superior parameter efficiency and performance over dense Transformers in protein language models[20,21]; EVA is the first RNA foundation model to adopt this design, and we validate its advantage empirically (Supplementary Table S5). However, training at this scale reveals that RNA sequences exhibit lower evolutionary conservation and a more severely imbalanced cluster size distribution than proteins (Figure S2C), causing standard training to overfit on abundant sequences while neglecting rare functional families (Figure S3A). We therefore implemented an evolutionary conservation-based sampling strategy (see Methods), which rebalances RNA-type abundances (Figure S2D) and improves model generalization (Figure S3A), with square-root sampling achieving the best balance between coverage and cluster down-weighting (Figure S2C, S3B).

To fully leverage this curated data while enabling controllable generation, we devised a two-stage training curriculum that maintains both CLM and GLM objectives throughout. In the pre-training stage, we condition the model solely on RNA-type tags, which we found to significantly enhance the quality of generated sequences across all criteria (GC content validity, sequence completeness, and low repetitiveness) by providing explicit family cues that help disentangle the unstructured global distribution (Figure S5). This allows the model to learn fundamental RNA syntax and structural motifs without being overwhelmed by fine-grained taxonomic details. In the subsequent mid-training stage, we introduce lineage (taxonomic) tags. This sequential integration of conditional information is critical; we found that introducing species-level constraints too early leads to overfitting, whereas our staged approach enables the model to first internalize universal RNA grammar before refining its latent space for species-specific adaptability (Figure S3C). This strategy enables EVA to unify de novo generation and targeted region redesign within a single framework, transitioning from a generic global distribution to a controllable, fine-grained latent space tailored to specific biological contexts (Figure 1G).

We observe consistent scaling behavior across all trained model sizes (∼21M to 1.4B), where perplexity decreases consistently with capacity, indicating progressively better capture of RNA grammar (Figure 1I). Hyperparameter exploration further reveals that mid-training lineage conditioning further reduces perplexity by disentangling species-specific constraints from universal RNA grammar (Figure S4B). Finally, using a "needle-in-a-haystack" evaluation, we demonstrate that EVA reliably

detects perturbations as subtle as 1% introduced at arbitrary positions within sequences up to 8,192 tokens, confirming robust long-range dependency recall across its entire context window (Figure 1H).

2.2 EVA predicts mutational effects across RNA, DNA, and protein

Language models have been shown to capture fitness landscapes by assigning higher likelihood to functional sequences and lower likelihood to deleterious variants[18,22,23]. Since EVA is trained on full-length RNA sequences, including coding RNAs that simultaneously encode both transcript-level and protein-level constraints, its manifold naturally reflects selection pressures operating across multiple molecular scales. We therefore evaluate EVA likelihood as a unified scoring function across these modalities.

We benchmark EVA on diverse deep mutational scanning (DMS) assays covering both structured ncRNAs and mRNAs. For each assay, we compute the Spearman rank correlation between per-variant EVA log-likelihood scores and experimentally measured fitness effects. Across structured ncRNA families, including ribozymes, aptamers, and tRNAs, EVA achieves a mean Spearman correlation of 0.40, establishing state-of-the-art performance. (Figure 2B). For mRNA assays, EVA attains a mean Spearman correlation of 0.31, surpassing prior RNA language models as well as dedicated codon optimization models, demonstrating sensitivity to translation efficiency and RNA stability constraints (Figure 2C). In both cases, EVA outperforms prior RNA/Codon language models trained on smaller or less phylogenetically diverse datasets, suggesting that scale and taxonomic breadth contribute to learning generalizable fitness landscapes.

Since RNA is transcribed from DNA, EVA can also model mutational perturbations in gene regions. We formulate gene essentiality prediction as a likelihood-change assay, whereby five tandem premature stop codons are inserted near the $5'$ end of each coding sequence, and the resulting drop in EVA log-likelihood relative to wildtype is used to infer whether the disrupted gene is functionally essential. On eukaryotic benchmarks, EVA surpasses Evo 7B despite being trained solely on RNA; on prokaryotic benchmarks, EVA's 21M-parameter model even outperforms Evo2 1B—a DNA foundation model more than 50× larger (Figures 2D-E). These results reveal that transcriptomic sequences inherently encode the evolutionary signatures of gene functionality, allowing EVA to infer gene essentiality without direct exposure to genomic DNA.

Although EVA is trained exclusively on RNA, coding sequences directly determine the encoded protein, enabling EVA to capture evolutionary constraints relevant to protein function. On protein DMS benchmarks, EVA likelihood-based scoring is computed by conditioning on the mRNA type token, achieving a mean Spearman correlation of 0.46 and matching or exceeding specialized protein language models such as ESMC 300M on several datasets (Figure 2F). These results suggest that coding RNA sequences carry evolutionary fitness constraints that partially overlap with those learned by protein-centric models, offering preliminary evidence that RNA can serve as a cross-modal representation bridging transcriptomic and proteomic fitness landscapes.

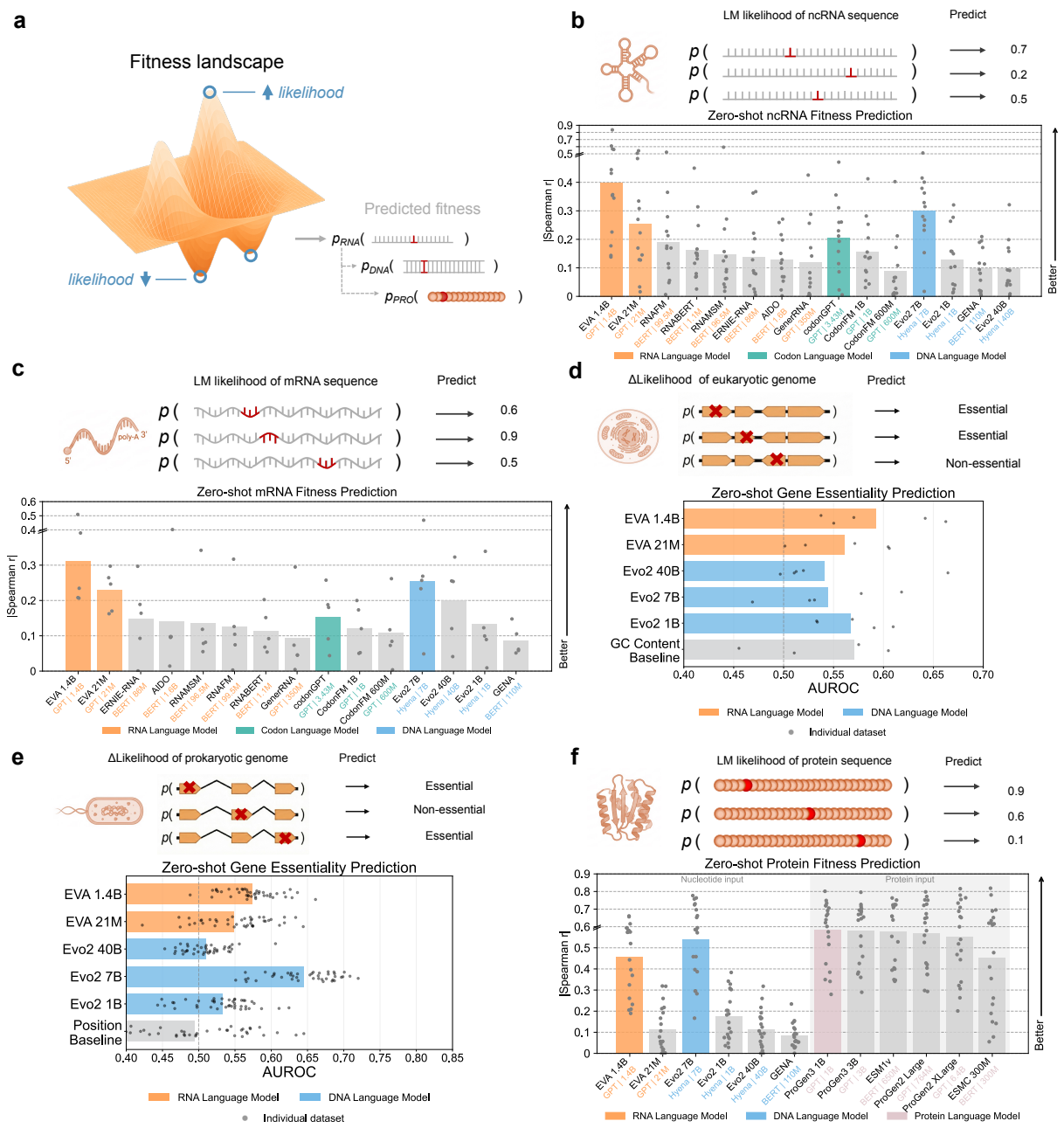

Figure 2: **Zero-shot fitness prediction across RNA, DNA, and protein.** (a) Conceptual overview of using EVA's zero-shot sequence likelihood to approximate fitness landscapes and mutation effects for RNA, gene regions, and proteins. (b)(c) Benchmark results for zero-shot fitness prediction on ncRNA (b) and mRNA (c) sequences, evaluating the correlation between EVA likelihood scores and experimentally measured fitness across a diverse collection of deep mutational scanning (DMS) assays. Performance is reported as Spearman's rank correlation (ρ); points indicate individual assays. (d)(e) Benchmark results on prokaryotic (d) and eukaryotic (e) gene essentiality prediction by inserting premature stop codons near the $5'$ end of coding sequences and quantifying the EVA log-likelihood drop to infer essentiality. (f) Benchmark results for zero-shot fitness prediction on protein, evaluating the correlation between likelihood scores and experimentally measured fitness. Nucleotide-input EVA predictions are compared against protein language model baselines.

2.3 EVA supports controllable RNA generation via CLM and GLM

EVA supports two complementary generation modes, Causal Language Modeling (CLM) for autoregressive sequence generation and Generalized Language Modeling (GLM) for masked infilling, both of which can be used with or without RNA-type and taxonomic conditioning tokens, enabling a spectrum from unconstrained sampling to fine-grained controllable design.

To quantify generation quality, we apply a lightweight filter to remove invalid samples based on GC content (30–70%), low-repeat content (<10%), and proper termination (see Methods). Incorporating RNA-type tags at pretraining markedly improves sample validity: the GC content pass rate increases from 47% without RNA-type pretraining ($n = 2{,}000$) to 97% for unconditional generation ($n = 982$) and 96% for conditional generation ($n = 3{,}625$), while repeat and termination filters reach near-perfect pass rates (≥99%) across all conditioned settings (Figure S5). These results demonstrate that RNA-type conditioning at pretraining improves the quality and biological validity of generated sequences.

We compare CLM and GLM in an unconditioned setting (Figure 3A). CLM spans both high-similarity and more divergent sequences relative to natural neighbours, reflecting the model's ability to both reproduce and extend the natural manifold; GLM achieves higher reconstruction similarity by leveraging richer flanking sequence context. When conditioned on RNA-type tokens, GLM infilling achieves median reconstruction similarity of around 0.5 across all 11 RNA classes (Figure 3B), demonstrating that both modes benefit from structured conditioning. Unlike conventional RNA language models that learn an unstructured global distribution, EVA steers sampling toward specific RNA classes and host organisms–generalizing class-conditioned generation[24] to RNA across all 11 major RNA classes.

To benchmark generation quality, we compare EVA with GenerRNA[15]—currently the only open-source RNA generative model; RNAGenesis[16] has not released model checkpoints and thus precludes direct comparison. Computing KL divergence on three secondary-structure feature groups (GC content & MFE, loop & helix composition, and base-pairing & stem statistics) across all 11 RNA classes, EVA achieves more than 13-fold lower KL divergence than GenerRNA, with particularly large improvements for piRNA, snRNA, snoRNA, and tRNA (Figure 3D; Supplementary Table S11). We further compare generated sequences with natural RNAs across sequence-space and biophysical statistics, including UMAP embedding, length distribution, MFE, GC content, and secondary-structure features; RNA-type conditioning yields closer agreement with natural distributions than unconditional generation (Figure 3E). This reflects the model's ability to capture family-specific sequence and structural grammars—for example, the high GC content and stable folds of rRNAs, or the short length and minimal structure of miRNAs. Extended distributional comparisons across all 11 RNA classes are provided in Figure S6.

After fixing RNA type (e.g., mRNA), we further condition generation on species tags. Restricting to mRNA, EVA achieves more than 30-fold lower KL divergence than GenerRNA across six representative species (*C. elegans*, *C. griseus*, *H. sapiens*, *D. melanogaster*, *M. musculus*, and *R. norvegicus*), demonstrating that species-aware conditioning enables accurate modeling of organism-specific RNA sequence and structural landscapes (Figure 3G; Supplementary Table S12). Distributional comparisons across sequence and structural metrics confirm that species conditioning further improves agreement between generated and natural sequences across *C. elegans*, *C. griseus*, and *H. sapiens* (Figure 3H). Extended results across additional RNA classes and species are provided in Figure S7.

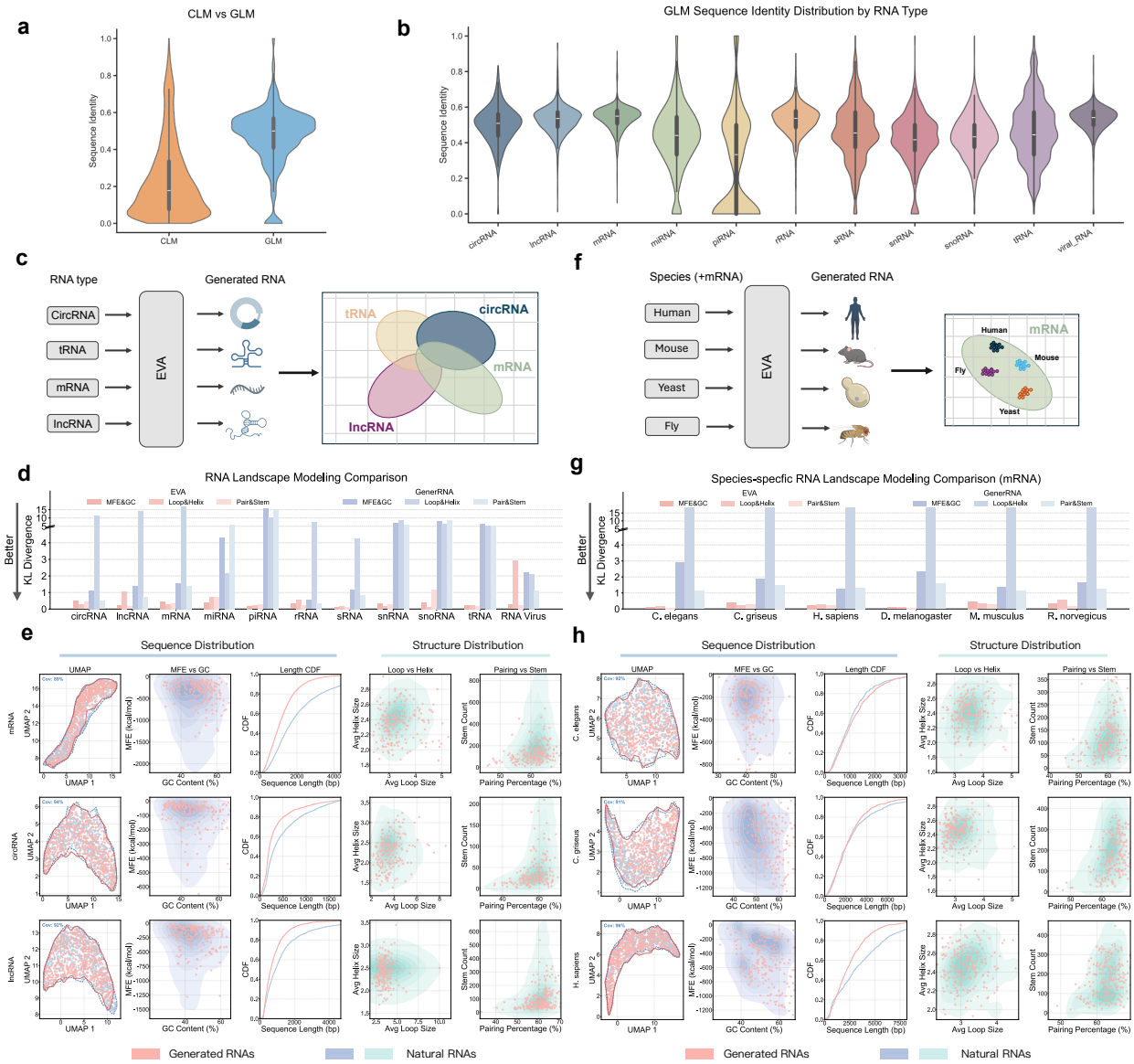

Figure 3: **Controllable RNA generation with EVA across diverse RNA families and species.**
(a) Violin plots of sequence similarity between generated sequences and their closest natural neighbours for unconditional generation with CLM, and reconstruction similarity for GLM; CLM spans both high-similarity samples and more divergent, novel sequences, whereas GLM yields higher reconstruction due to access to richer flanking sequence context. (b) GLM infilling performance across 11 RNA classes, quantified as similarity between the masked fragment and the generated fragment. (c) Schematic of conditional generation controlled by RNA-type tags. (d) Distributional comparisons between RNA-type conditioned samples and natural RNAs across sequence-space and biophysical/structural statistics, including UMAP embedding, length, minimum free energy (MFE), GC content, and secondary-structure features (stem/helix/loop and base-pairing statistics). (e) Schematic of conditional generation controlled by taxonomic tags. (f) Species-conditioned generation within generated mRNAs and natural sequences across three representative species (*C. elegans*, *C. griseus*, and *H. sapiens*) in both sequence, biophysical and structural metrics; results across additional species and RNA classes are provided in Figure S7.

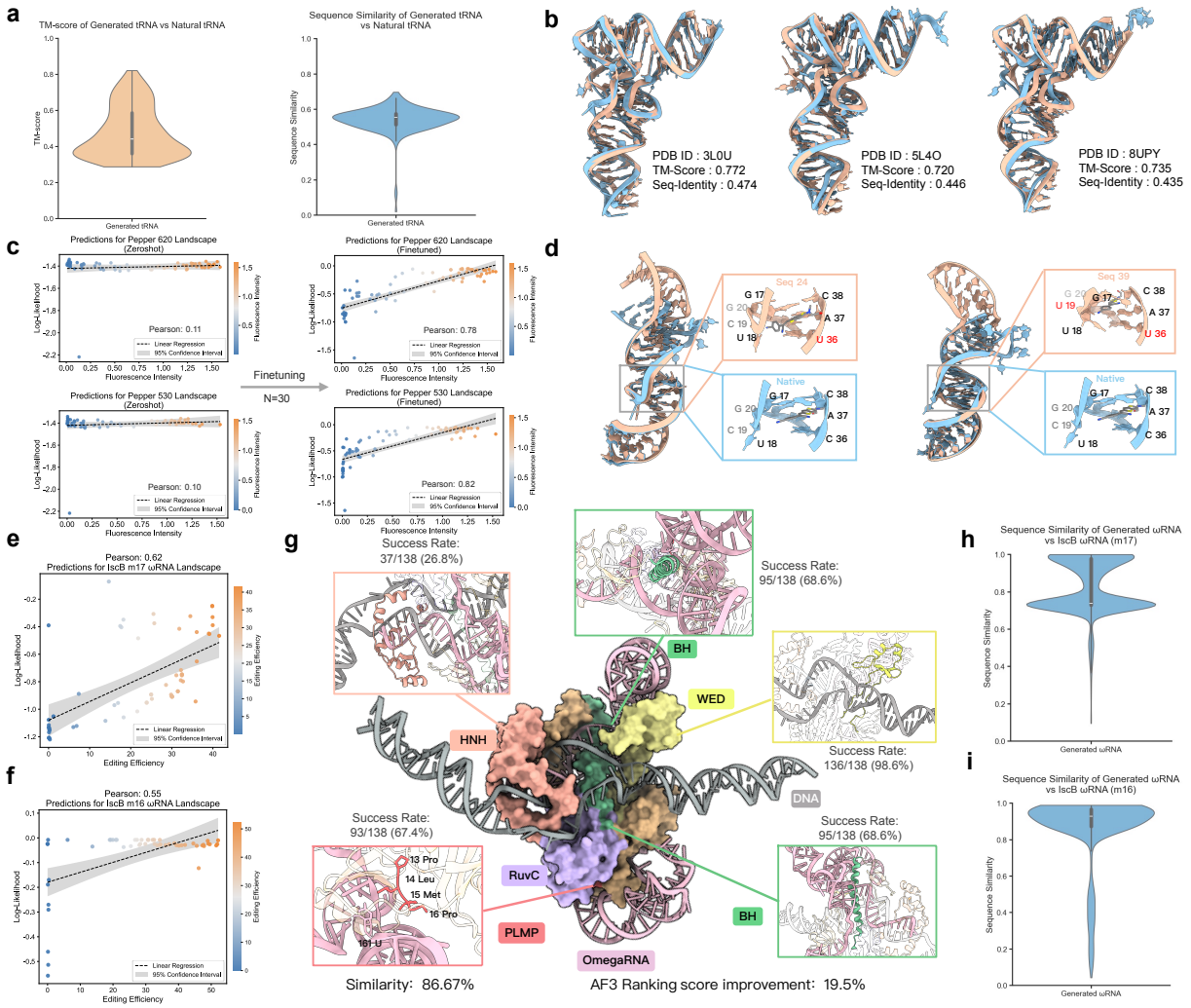

Figure 4: **De novo design of tRNAs, RNA aptamers, and CRISPR gRNAs.** (a) Zero-shot de novo generation of tRNAs showing high structural similarity to natural tRNAs, quantified by TM-score, while maintaining sequence novelty relative to known tRNAs. (b) Representative structural alignments between AlphaFold3-predicted structures of generated tRNAs and experimentally resolved natural tRNAs, illustrating preservation of the canonical L-shaped fold despite low sequence identity. (c) Prediction of fluorescence intensity for RNA aptamers after low-data fine-tuning (n = 30). Left: binding to HBC530; right: binding to HBC620. Model log-likelihood scores are compared against experimentally measured fluorescence. (d) Structural comparison between de novo generated aptamers (blue) and wildtype structures, showing preservation of the ligand-binding pocket and overall fold despite sequence mutations, consistent with learned mutational-structural relationships. (e,f) Prediction of editing efficiency for ISCB omegaRNAs after fine-tuning on m16 (n = 42) and m17 (n = 44) sequence sets, respectively. (g) AlphaFold3-predicted complex structure of ISCB protein, de novo designed m16 omegaRNA, and target DNA. Success is defined by satisfying a set of domain-level structural criteria relative to wildtype (see Methods). (h,i) Violin plots showing sequence similarity distributions between de novo generated omegaRNAs and the corresponding wildtype sequences for m17 (h) and m16 (i). Generated sequences span a biologically reasonable range of divergence while remaining structurally plausible.

2.4 EVA generates novel RNAs with structural fidelity and extends to unseen RNA classes via fine-tuning

Having established that EVA's conditional generation recapitulates natural sequence and secondary structure distributions, we next examine whether the model also captures tertiary structural information. tRNA is an informative test case, as its L-shaped fold is structurally well-defined, functionally critical, and must be encoded entirely in sequence.

Structural similarity was assessed by TM-score between AlphaFold3-predicted structures and all experimentally resolved tRNA structures from RCSB; sequence novelty was quantified as maximum sequence identity against 10,000 randomly sampled tRNAs from training data. Across the generated ensemble, structural similarity to natural tRNAs and sequence divergence from known tRNAs are largely decoupled (Figure 4A), with generated sequences maintaining a mean TM-score of 0.47 even as sequence identity to training tRNAs drops to approximately 0.55, consistent with EVA capturing fold-level constraints independently of specific sequences. Structural alignment of selected generated sequences against their closest experimentally resolved counterparts in RCSB illustrates this at the level of individual examples (Figure 4B), where representative cases achieve a mean TM-score of approximately 0.74 at an average sequence identity of only 0.45, demonstrating that overall backbone topology is preserved despite substantial sequence divergence from any known tRNA. Together, these observations suggest that training on evolutionarily diverse sequences implicitly encodes fold-level structural constraints, consistent with the notion that EVA's learned distribution approximates an evolution-shaped manifold in which sequence novelty and structural viability coexist.

While EVA supports conditional generation for 11 major RNA classes, many valuable RNAs remain outside this scope. RNA aptamers are synthetically engineered molecules absent from nature, and guide RNAs from metagenomic CRISPR-Cas systems represent highly specialized, application-specific sequences. To enable design beyond pre-trained families, we reserved a blank conditioning token to support fine-tuning and conditional generation for user-defined RNA types.

We fine-tuned EVA on a small set of high-fluorescence aptamer sequences ($n = 30$), using sequence-level self-supervised language modeling to steer the model's generative distribution toward the functional aptamer sequence space. After fine-tuning, EVA log-likelihood scores correlate with experimentally measured fluorescence intensity for aptamers binding to both HBC530 and HBC620 ligands, with Spearman correlation improving from 0.1 (zero-shot) to 0.8 (Figure 4C). Position-wise comparison of mutation enrichment confirms that the fine-tuned model preferentially generates the same nucleotide substitutions that experimentally confer improved fluorescence activity, as revealed by alignment with the DMS fitness landscape (Figure S8A, left). Generated aptamers show mutation counts within a biologically reasonable range relative to wildtype (Figure S8A, right). De novo generated aptamers show preserved ligand-binding pocket geometry despite sequence mutations in predicted structures (Figure 4D), providing an in silico indication of structural plausibility.

IscB is an ancestral CRISPR-associated transposon protein that pairs a compact protein with an large guide RNA, the ωRNA[25]. A central engineering challenge is to reduce ωRNA size while preserving target-site recognition, enabling more efficient viral delivery. We applied the same fine-tuning strategy using high-editing-efficiency sequences from two guide RNA variants. After fine-tuning, EVA log-likelihood correlates with editing efficiency for both variants (Figure 4E, 4F), and more than half of de novo generated sequences are shorter than the respective wildtype (Figure S8B). Structural plausibility was assessed using five domain-level geometric criteria derived from published IscB structural analyses[25,26] (Table S13); designs satisfying all criteria are flagged as in silico candidates for experimental validation (Figure 4G). Generated ωRNAs span a biologically reasonable range of sequence identities relative to wildtype (Figure 4H, 4I). These results demonstrate that EVA adapts to RNA classes absent from its training distribution using minimal data.

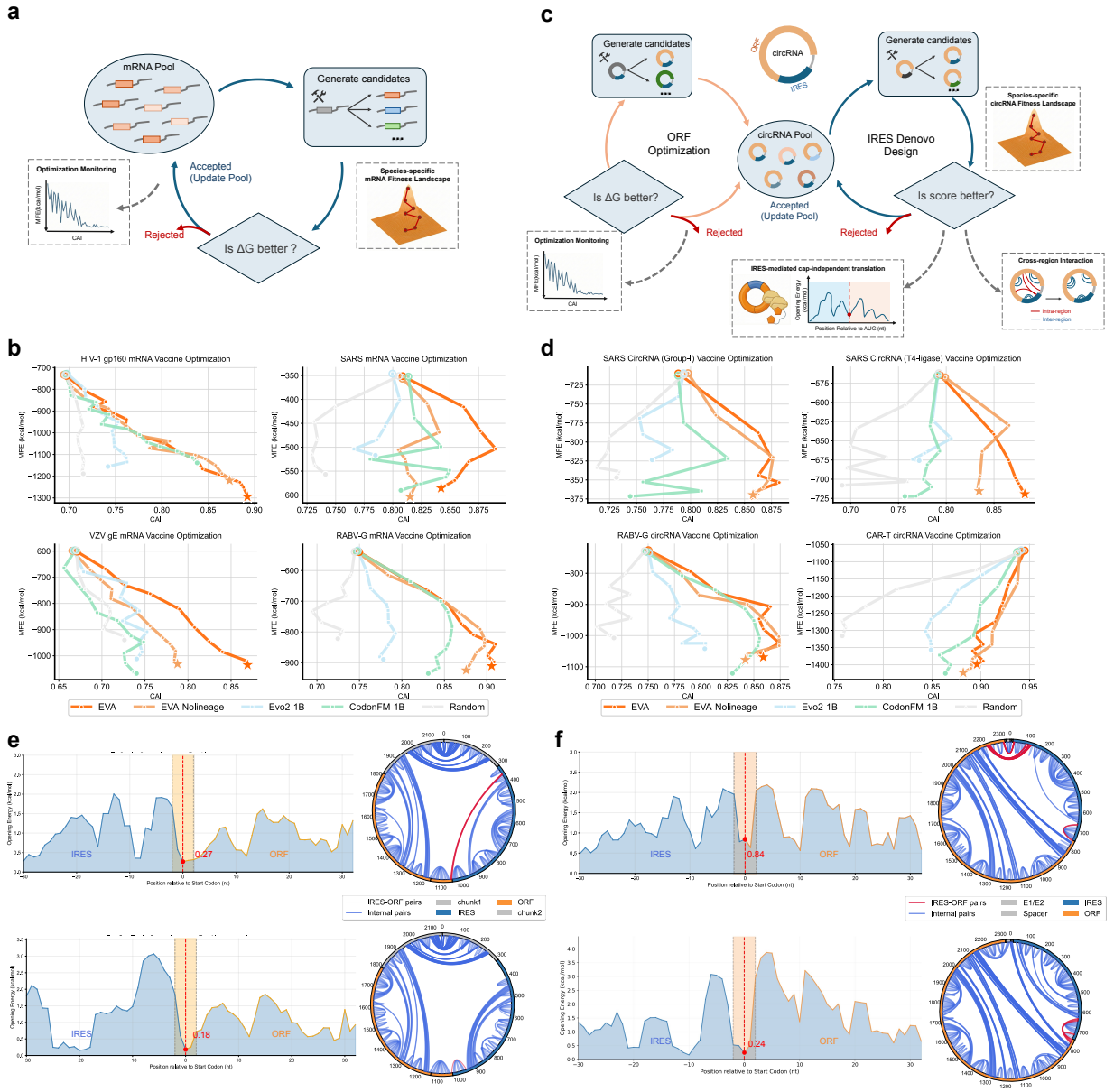

Figure 5: **mRNA and circRNA vaccine design with EVA.**
(a) Conceptual overview of mRNA codon optimization with EVA. (b) Benchmark on four mRNA vaccine systems (SARS-CoV-2 Spike, VZV gE, RABV-G, HIV) comparing CDS optimization in terms of MFE and CAI against Evo2-1B, CodonFM-1B, and a Random (MFE-only) baseline, all conditioned on the mRNA type and *Homo sapiens*. (c) Conceptual overview of circRNA IRES redesign via EVA GLM, where the IRES region is masked and regenerated conditioned on flanking sequences. (d) Benchmark on four circRNA vaccine systems (SARS: Group-I, SARS: T4 ligase, rabies, CAR-T) for ORF codon optimization, same baselines and species conditioning as (b). (e)(f) Biophysical properties of EVA-redesigned IRES sequences versus the original CVB3 IRES for SARS and rabies circRNA vaccines, including ribosome accessibility and inhibitory long-range interactions.

2.5 EVA enables mRNA and circRNA vaccine design

We next assess whether EVA can be used as a practical design engine for RNA vaccines. RNA vaccines have emerged as a transformative therapeutic modality, yet their design remains challenging as efficacy depends on multiple sequence-level properties–codon usage, folding stability, and translational efficiency–that must be jointly optimized. Circular RNAs have attracted increasing interest as vaccine platforms in recent years, owing to their covalently closed topology, which confers markedly greater resistance to exonucleolytic degradation compared to linear mRNA and can extend the duration of antigen expression in vivo[27,28]. However, circRNA vaccine design presents an additional engineering challenge beyond ORF codon optimization. Because the molecule lacks a $5'$ cap and poly(A) tail, translation must be entirely driven by an internal ribosome entry site (IRES), making IRES design a critical determinant of translational efficacy[27,28].

The standard pipeline for mRNA codon optimization jointly optimizes minimum free energy (MFE) for RNA folding stability and the Codon Adaptation Index (CAI) for host codon usage compatibility[29]. However, CAI treats each codon position independently and relies on fixed frequency tables, lacking the ability to model broader sequence context or organism-specific translational dynamics[30–36]. We therefore replace CAI with EVA log-likelihood under joint RNA-type and species conditioning, which captures host-organism compatibility in a context-aware, unified score while retaining MFE as the folding objective (Figure 5A). We perform in silico codon optimization on four mRNA vaccine systems (SARS-CoV-2 Spike, Varicella-Zoster Virus (VZV) gE, RABV-G, and HIV) and benchmark against Evo2-1B[19] and CodonFM-1B[32] as baselines. EVA-guided synonymous search improves MFE while simultaneously increasing CAI compared to both baselines; joint RNA-type and species conditioning further boosts performance by aligning the optimization trajectory with the class- and organism-specific sub-manifold (Figure 5B). Detailed numerical results for all four vaccine systems and all baselines are provided in Supplementary Table S5.

Circular RNA vaccines have attracted growing interest as a next-generation platform due to their covalently closed topology, which resists exonucleolytic degradation and prolongs antigen expression relative to linear mRNA. We apply the same EVA-guided codon optimization pipeline (MFE + EVA log-likelihood, conditioned on the circRNA type and *Homo sapiens*) to circRNA ORF optimization (Figure 5C). We evaluate four circRNA systems (SARS: Group-I, SARS: T4 ligase, rabies, and CAR-T) against the same baselines (Evo2-1B[19] and CodonFM-1B[32]) and observe consistent improvements in both CAI and MFE, with additional gains when conditioning on the target species (Figure 5D). Detailed ORF optimization results for all circRNA systems are provided in Supplementary Table S6.

Beyond ORF optimization, circRNA translation depends entirely on the Internal Ribosome Entry Site (IRES). De novo IRES design offers the potential for reduced immunogenicity, improved circularization compatibility, and enhanced cell-type specificity compared to fixed viral elements such as CVB3[27].

We address this by leveraging EVA's GLM capability, masking the IRES region and conditioning generation on the flanking ORF and cyclization sequences (circRNA type, *Homo sapiens*), the model enables de novo design of IRES within its molecular context (Figure 5C). Designed IRES candidates were evaluated on three biophysical criteria (ribosome accessibility, inhibitory long-range interactions, and MFE; see Methods). Across both SARS and rabies circRNA systems, EVA-generated IRES candidates outperform the original CVB3 IRES across all three biophysical metrics. For SARS circRNA, inhibitory long-range interactions are reduced by 58% and ribosome accessibility is increased by 33% (Figure 5E). For rabies circRNA, inhibitory long-range interactions are reduced by 27% and ribosome accessibility is increased by 71% (Figure 5F).

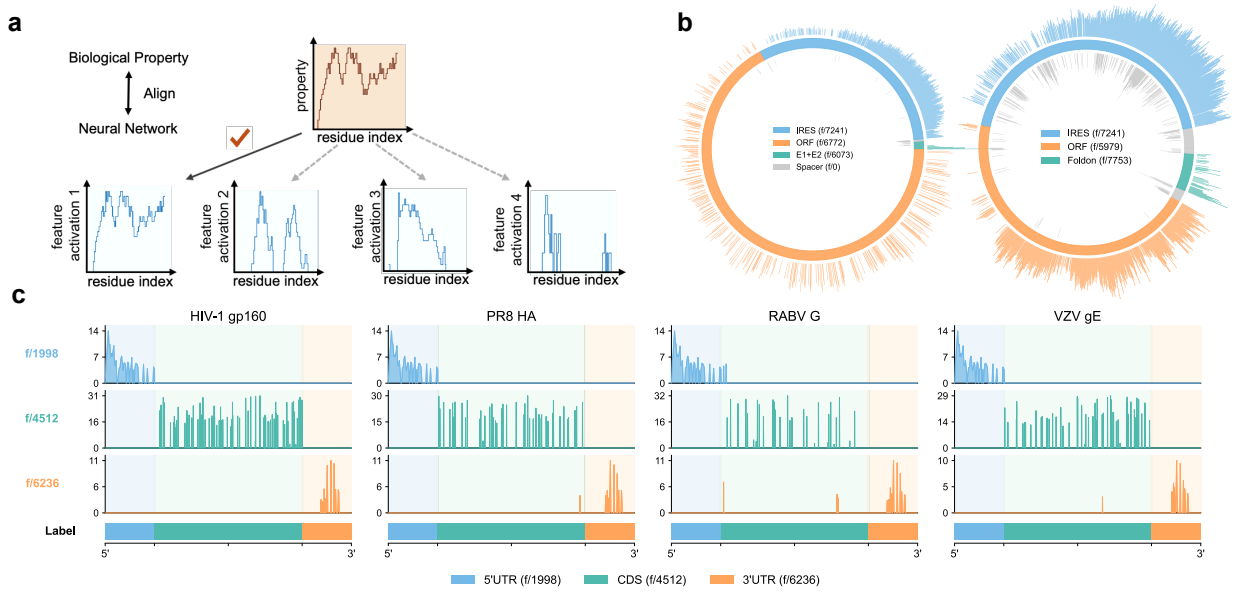

Figure 6: **Interpretability analysis revealing hierarchical RNA features learned by EVA.** (a) Schematic illustrating how model neurons are mapped to biological concepts using sparse autoencoders, revealing that EVA can learn structure- and function-related biological knowledge from sequence alone. (b) Activations of features associated with IRES, ORF, and related functional elements in circRNA vaccines (left: RABV-G; right: SARS-CoV-2). (c) Activations of features associated with 5′UTR, CDS, and 3′UTR in mRNA vaccines (HIV, PR8 HA, RABV-G, and VZV gE).

2.6 Feature interpretation in EVA from neurons to biological properties

Despite EVA's strong performance across prediction and generation tasks, it remains important to ask what the model has actually learned from RNA sequences. Large language models are often characterized as opaque, yet mechanistic interpretability methods offer tools to examine their internal representations directly. We apply sparse autoencoders (SAEs)[37] to decompose the dense, distributed activations of EVA's residual stream into a high-dimensional set of sparse features, each of which can be examined for biological correspondence (Figure 6A). SAEs were trained on layer-13 activations extracted from one million RNA sequences spanning diverse RNA types and taxonomic groups (see Methods).

To assess whether learned features correspond to known RNA biology, we identify features whose activation is systematically enriched over specific functional sequence elements. In circRNA vaccine sequences (RABV-G and SARS-CoV-2), individual SAE features activate selectively over IRES and ORF regions (Figure 6B), suggesting that the model has formed internal representations that track these regulatory and coding boundaries. In mRNA vaccine sequences (HIV, PR8 HA, RABV-G, and VZV gE), three distinct features consistently demarcate the 5′UTR, CDS, and 3′UTR across all four constructs, despite substantial differences in their coding sequences (Figure 6C). This cross-construct consistency indicates that these features reflect generalizable positional and functional logic rather than sequence-specific memorization, suggesting that EVA has learned not only the identity of individual functional elements but also their characteristic spatial arrangement within a transcript.

At a higher organizational level, SAE features cluster according to the biological domain of origin: features preferentially activated by eukaryotic sequences and those preferentially activated by

prokaryotic sequences occupy spatially separated regions in the UMAP projection of SAE decoder directions (Figure S10). Individual experts within EVA's Mixture-of-Experts backbone further exhibit non-uniform activation profiles across RNA types (Figure S10), consistent with a degree of functional specialization emerging from training. Together, these observations suggest that EVA's internal representations are organized along biologically meaningful axes, from domain-level taxonomy to transcript-level functional architecture.

3 Discussion

EVA demonstrates that large-scale generative learning on evolutionary RNA sequence diversity yields a unified framework supporting both sequence scoring and controllable design. By training on 114 million full-length RNA sequences with structured conditioning on RNA type and taxonomy, EVA learns an evolution-consistent representation in which sequence likelihood reflects evolutionary plausibility, enabling zero-shot mutation-effect prediction across RNA, DNA gene regions, and proteins without task-specific retraining. Conditioned generation across 11 RNA classes, enabled in part by an 8,192-token context window that accommodates full-length transcripts beyond the reach of prior models, produces sequences that recapitulate natural length, folding energy, and structural distributions while remaining evolutionarily novel, and practical applications spanning tRNA design, aptamer engineering, CRISPR guide RNA generation, and mRNA/circRNA vaccine optimization demonstrate the breadth of this capability.

A notable finding is that, enabled by EVA's mixture-of-experts architecture and joint RNA-type and taxonomic conditioning that disentangles the heterogeneous RNA sequence landscape, the learned representations are interpretable and structurally organized. Sparse autoencoder analysis reveals that individual features localize to functional elements, including 5′UTRs, CDSs, 3′UTRs, IRESs, and ORFs, in a manner that generalizes across diverse constructs, indicating that EVA has leanred the functional architecture of RNA from sequence alone. Complementarily, experts in the MoE backbone exhibit RNA-type-specific activation patterns, and features segregate by biological domain in the SAE latent space, together suggesting that EVA's internal organization mirrors the hierarchical structure of RNA biology.

The current work is limited to computational evaluation; wet-lab validation of designed sequences is ongoing. EVA also operates at the sequence level and does not explicitly model three-dimensional structure, which may limit applicability to tasks requiring precise conformational control. Additionally, the granularity of RNA type classification warrants further refinement, and the range of RNA classes supported for controllable generation without fine-tuning remains an area for expansion.

Looking forward, we see two directions of particular interest. First, scaling the model architecture and extending the context window could enable design at the level of complete RNA systems, moving from individual functional elements toward more complex assemblies such as RNA viral genomes; recent work demonstrating the generative design of viable bacteriophages from genome language models provides an encouraging proof of concept[38]. Second, the interpretability analyses presented here establish a foundation for a complementary goal: not merely identifying which internal features correspond to biological concepts, but actively steering those features to guide generation toward desired functional outcomes[39–41]. Together, these directions point toward generative models that are both more capable and more controllable, advancing the generative design of increasingly complex RNA systems as a step toward generative biology.

4 Methods

4.1 Model Training

EVA is trained on RNA sequences from OpenRNA v1 using a two-stage curriculum that combines Causal Language Modeling (CLM) and Generalized Language Modeling (GLM) objectives throughout both stages, while progressively introducing richer conditioning information. In Stage 1 (pre-training), the model is conditioned on RNA-type tags and learns universal RNA sequence grammar. In Stage 2 (mid-training), taxonomic lineage tags are additionally introduced, enabling species-aware generation. We trained EVA across a range of model scales on 16 NVIDIA H200 141 GB GPUs across 2 nodes, and observe consistent perplexity reduction with increasing capacity. Benchmarks and evaluations are conducted on the smallest (∼21M parameters) and largest (1.4B parameters) checkpoints; all intermediate checkpoints are publicly released. All experiments use a context length of 8,192 tokens.

4.1.1 Model Architecture

EVA is a decoder-only Transformer with an 8,192-token context window. It incorporates rotary positional embeddings (RoPE, $\theta = 10^5$), multi-head attention (MHA), and pre-layer RMSNorm normalization. Feed-forward layers are replaced by a Mixture-of-Experts (MoE) backbone implemented via MegaBlocks, with 8 experts per layer of which 2 are activated per token (top-2 routing) and a capacity factor of 1.5. An auxiliary load-balancing loss with coefficient $\lambda_{\text{aux}} = 0.01$ is added to prevent expert collapse. Full architecture configurations and training hyperparameters for all four model scales are reported in Supplementary Table S4.

4.1.2 Training Objectives

Mixed CLM and GLM pre-training. During training, each example is assigned one of two objectives sampled per sequence: (i) **Causal Language Modeling (CLM)**, in which the model performs standard autoregressive next-token prediction over the full sequence; or (ii) **Generalized Language Modeling (GLM)**, in which one or more contiguous spans are replaced by unique sentinel tokens (`<span_N>`) and the model predicts the masked tokens conditioned on the surrounding context and the corresponding sentinel, enabling fill-in-the-middle infilling. The GLM infilling objective generalizes the fill-in-the-middle (FIM) paradigm originally introduced for code language models[42,43] and subsequently adopted by protein language models such as xTrimoPGLM[44] and ProGen3[21]; we adapt this technique to RNA sequence modeling. Following ablation results from ProGen3[21], we use a CLM:GLM mixing ratio of 2:1 ($p_{\text{GLM}} = 0.333$). For GLM examples, the total coverage ratio of masked tokens is drawn from $\{0.15, 0.25, 0.50, 0.80\}$ with probabilities $(0.28, 0.30, 0.28, 0.14)$; individual span lengths follow a mixture of Gaussian distributions (mean/std: 10/5, 20/10, 50/20 tokens) with mixing weights $(0.3, 0.5, 0.2)$, capped at 10 spans per sequence and with no overlap between spans.

The composite training loss is:

$$\mathcal{L} = \mathcal{L}_{\text{LM}} + \lambda_{\text{aux}} \mathcal{L}_{\text{aux}}, \tag{1}$$

The language modeling loss $\mathcal{L}_{\text{LM}}$ is the mean cross-entropy over all non-padding target positions $\mathcal{S}$ (positions where the label is not the ignore index -100):

$$\mathcal{L}_{\text{LM}} = -\frac{1}{|\mathcal{S}|} \sum_{t \in \mathcal{S}} \log p_\theta(x_t \mid x_{<t}), \tag{2}$$

where for CLM examples $\mathcal{S}$ spans all sequence tokens, and for GLM examples $\mathcal{S}$ contains only the masked span tokens (all other positions carry label -100 and are excluded from the loss).

The auxiliary load-balancing loss $\mathcal{L}_{\text{aux}}$ follows the Switch Transformer formulation[45], encouraging uniform expert utilization across the batch:

$$\mathcal{L}_{\text{aux}} = E \sum_{i=1}^{E} f_i \cdot P_i, \tag{3}$$

where E is the number of experts, $f_i = \frac{1}{T} \sum_t \mathbf{1}[\operatorname{argmax} \mathbf{r}(x_t) = i]$ is the fraction of tokens routed to expert i, $P_i = \frac{1}{T} \sum_t r_i(x_t)$ is the mean router probability for expert i over all T tokens in the batch, and $\lambda_{\text{aux}} = 0.01$.

Position-Preserving Fuzzy Encoding. For the GLM infilling objective, we adopt a position-preserving fuzzy encoding scheme. Given an RNA sequence with one or more masked spans, each span is replaced in-place by a unique sentinel token (`<span_N>`, $N \in \{0, \ldots, 49\}$); the resulting context (with sentinels) is presented to the model first, terminated by `<eos>`, after which each span's masked content is appended sequentially—each delimited by its corresponding sentinel and a closing `<eos_span>` token.

Position IDs are assigned as follows. In the *context* segment, tokens before the first masked span retain their original sequence positions; each sentinel `<span_N>` receives the position ID of the first masked nucleotide of its span (adjusted by any previously accumulated offset); and all context tokens that follow the k-th span have a cumulative *fuzzy offset* $\Delta_k = \sum_{j=1}^{k} \delta_j$ added to their original positions, where each δ_j is sampled independently from a geometric distribution with mean $0.2 \times L_j$ (L_j = length of span j). In the *infilling* segment (after `<eos>`), each sentinel is placed at the *original* (un-offset) position of the first masked nucleotide; the masked tokens that follow receive positions $p_{\text{start}} + 1$, $p_{\text{start}} + 2$, … (where p_{start} is the sentinel position), and `<eos_span>` is placed one position after the last masked token. Because the fuzzy offsets make the positional gap between a sentinel and the subsequent context tokens ambiguous, the model cannot infer the exact length of the masked span from positional cues alone, and must instead learn to terminate infilling by generating the content-based `<eos_span>` token.

A concrete RNA example is shown below, with two masked spans highlighted. Conditioning prefix tokens (lineage tag, RNA-type tag) precede the shown sequence in actual training inputs but are omitted here for clarity.

Original RNA: 5 A U G C U A G C A U G A U G C 3

Infilling Sequence:

```
<bos_glm>  5  A  U  G  C  <span_3>  A  U  G  <span_1>  3  <eos>
           <span_3> U A G C <eos_span> <span_1> A U G C <eos_span>
```

Position IDs:

```
0   1  2  3  4  5     6     11 12 13     14    20  21
                        6  7  8  9 10  11     13 14 15 16 17  18
```

Fuzzy offsets: $\delta_1 = 1$ after span 1 ($L_1 = 4$, mean $= 0.8$); $\delta_2 = 2$ after span 2 ($L_2 = 4$, mean $= 0.8$).
Context token A after span 1: original pos. 10, displayed as $10 + \delta_1 = 11$. Context token 3: original pos. 17, displayed as $17 + \delta_1 + \delta_2 = 20$.
Infilling tokens retain original positions (no fuzzy applied); sentinels in the infilling section use the un-offset span-start position.

Input format and conditioning tokens. Each sequence is prepended with structured conditioning tokens that encode RNA-type identity (*e.g.*, `<rna_tRNA>`, `<rna_mRNA>`) and strand directionality (`5` or `3`). In Stage 2 (mid-training), a Greengenes-format taxonomic lineage tag is additionally prepended at the outermost level, resulting in inputs of the form:

```
<bos> |lineage; <rna_type>| 5 [nucleotides] 3 <eos>
```

where the lineage string follows the seven-level Greengenes hierarchy (`d__domain; p__phylum; ...; s__species`). For GLM examples, `<bos>` is replaced by `<bos_glm>` and the sequence tokens contain the sentinels and infilling structure described above. Reverse-complement augmentation is applied with probability 0.5 in both stages: a sequence originally read `5 [seq] 3` is randomly swapped to `3 [rev-comp] 5`, augmenting coverage of directional sequence patterns.

4.1.3 Two-Stage Training Curriculum

Stage 1: RNA-type-conditioned Pre-training. In Stage 1, the model is trained on the full 114M-sequence OpenRNA v1 dataset conditioned only on RNA-type tags. The mixed CLM+GLM objective enables the model to simultaneously learn autoregressive generation and span infilling. By withholding fine-grained taxonomic information, Stage 1 focuses the model on universal RNA sequence grammar, including shared structural motifs, canonical base-pairing patterns, and family-level syntax, across all 15 RNA categories.

Stage 2: Lineage-conditioned Mid-training. In Stage 2, training resumes from the Stage 1 checkpoint, with only model weights loaded; the optimizer state and learning-rate scheduler are fully reset. The key change is the introduction of Greengenes-format phylogenetic lineage tags as additional conditioning prefixes. To prevent catastrophic forgetting of the universal RNA grammar learned in Stage 1, 20% of training examples are kept in the pre-training (no-lineage-prefix) format, while 80% use the full lineage-conditioned format. The mixed CLM+GLM objective is retained. This staged approach, which learns RNA-type-level grammar first and subsequently incorporates species-specific constraints, proves critical; empirically, introducing lineage conditioning from the outset leads to overfitting, whereas the two-stage curriculum achieves substantially lower perplexity and better generation quality (Figure S4B).

4.1.4 Data Sampling Strategy

A major challenge in training RNA foundation models is the extreme imbalance and low evolutionary conservation of RNA sequences compared to proteins. The natural RNA landscape follows a heavy-tailed distribution, dominated by a few massive clusters of housekeeping RNAs (e.g., rRNAs, tRNAs) while containing numerous small clusters of diverse, functional non-coding RNAs. We found that direct training on this raw distribution causes the model to severely overfit to abundant clusters while neglecting rare families. To overcome this, we implemented an **evolutionary conservation-based sampling strategy**. Sequences were first clustered by similarity to estimate the evolutionary density of each family. During training, sampling probability was re-weighted in inverse proportion to the square root of cluster size, effectively down-sampling abundant housekeeping RNAs and up-sampling rare functional RNAs. This strategy flattens the training distribution, ensuring the model captures the full diversity of the RNA landscape. Comparative training dynamics confirm that this approach prevents overfitting to dominant clusters, including rRNA and tRNA, and yields substantially improved model generalization compared with uniform or inverse-linear sampling (Figure S2C, S2D).

4.1.5 Optimization and Hyperparameters

All models are trained with the AdamW optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.95$, $\varepsilon = 10^{-8}$, weight decay $= 5 \times 10^{-6}$). The learning rate follows a linear warm-up schedule (3,000 steps in Stage 1; 1,000 steps in Stage 2), followed by cosine annealing to a minimum of 10% of the peak learning rate. Based on systematic hyperparameter search, we use a peak learning rate of 1×10^{-4} (Figure S4A), notably smaller than typical values for DNA or protein language models. The effective batch size is 256 sequences (4 sequences per device $\times$ 16 gradient accumulation steps $\times$ 4 data-parallel replicas). All models are trained in BF16 mixed precision.

4.1.6 Scaling Law Analysis

Scaling laws describe how model performance improves with compute and can guide optimal allocation of pre-training resources between model size and data[46,47]. To characterize EVA's scaling behavior, we fit a Kaplan-style power law[46] to the final validation perplexities of four model scales (21M, 167M, 470M, and 1.4B active parameters). Training FLOPs F are computed based on active parameters (i.e., the subset of parameters activated per forward pass in the MoE architecture), following standard practice for sparse models. We fit the power law in log space, modeling $L = \ln(\text{PPL})$ as a function of F:

$$L(F) = A \cdot F^{-\alpha} + C, \tag{4}$$

fitted via nonlinear least squares using `scipy.optimize.curve_fit`. The fitted parameters are $A = 75867.99$, $\alpha = 0.3211$, and $C = 1.2444$, yielding a smooth power-law decrease in perplexity(Figure 1H).

4.1.7 Distributed Training Infrastructure

The 1.4B model employs a hybrid parallelism strategy across 16 H100 141 GB GPUs (2 nodes, 8 GPUs each). **Expert parallelism** (EP = 4) shards the 8 MoE experts across 4 GPUs via All-to-All communication using the MegaBlocks framework; each GPU holds 2 experts and processes tokens routed to its local experts. Orthogonally, **data parallelism** (DP = 4) replicates each EP group across 4 independent data-parallel ranks, with gradients of non-expert parameters synchronized via All-Reduce across the DP group after each gradient-accumulation cycle. The resulting EP$\times$DP $=$ 4$\times$4 $=$ 16 GPU layout achieves near-linear throughput scaling while keeping per-device peak memory within the 141 GB limit.

4.2 OpenRNA v1 training data

Existing large-scale nucleotide databases such as MARS[17] aggregate extensive sequence collections but intermix RNA and DNA data; models trained on such mixed-modality corpora have been shown to underperform relative to those trained on curated, modality-specific data[10], motivating the construction of a dedicated, high-quality, full-length RNA corpus. To build open v1, we surveyed 19 public RNA resources and integrated 15 databases into a unified atlas containing 114M curated full-length RNA sequences. More than 90% of sequences are annotated with both RNA type and taxonomic labels. OpenRNA v1 covers 15 RNA categories, including mRNA, lncRNA, circRNA, snoRNA, snRNA, sRNA, tRNA, rRNA, piRNA, miRNA, YRNA, scaRNA, ribozyme, viral RNA and vaultRNA.

4.2.1 Data curation

OpenRNA v1 was built by integrating sequences from eight primary sources: NCBI Nucleotide and Virus databases (NT & Virus; 56.2M sequences after filtering, 49.25%), RNACentral Consortium aggregating Rfam, GtRNAdb and partner databases (33.4M, 29.25%), Ensembl vertebrate genomes Release 114 (15.7M, 13.75%), three merged circRNA databases (circBase, circAtlas; 2.1M, 1.87%), SILVA high-quality ribosomal RNA alignments (0.55M, 0.48%), NONCODE long non-coding RNAs (0.22M, 0.19%), piRNAdb (42K, 0.04%), and a collection of organism-specific databases including WormBase, FlyBase, snoDB, and miRBase (5.9M, 5.17%). Sequences were first filtered to remove redundant entries arising from cross-database overlap, low-complexity sequences (e.g., homopolymer-dominated transcripts), and sequences with ambiguous nucleotide content exceeding 5%. Following filtering, 114M unique full-length RNA sequences were retained (Supplementary Table S2).

Core and Genomic Archives. We utilized RNACentral and Rfam as the structural backbone. For Rfam, we implemented a dual-resolution labeling strategy: sequences with definitive family-level evidence were assigned specific functional labels (e.g., *snRNA*, *snoRNA*), while transcripts lacking high-resolution classification but exhibiting characteristic short-RNA length profiles (50–200 nt) were curated as *sRNA*. This category serves as a reservoir of structural candidates. To ensure broad taxonomic coverage, we integrated Ensembl (341 species) and the NCBI Nucleotide (NT) database. For Ensembl, we filtered out low-complexity regions (e.g., poly-A/T stretches). For the massive NCBI NT collection, we employed `accession2taxid` for precise species mapping and removed sequences with ambiguous taxonomic lineage.

Specialized Functional RNA Enrichment. We enriched the dataset with specialized repositories to capture distinct RNA modalities. For lncRNAs, we integrated NONCODE, LNCipedia, LncRNAWiki, and LncRNADisease; poly(A) tails in LNCipedia entries were computationally trimmed to avoid homopolymer bias. For small RNAs, miRBase and MirGeneDB were unified for miRNAs, explicitly distinguishing mature sequences from precursors and excluding incomplete stem-loop fragments; piRNAdb and snoDB were incorporated to cover piRNAs and snoRNAs, respectively. circRNA sequences (from circBase) were standardized to linear representations for tokenization. rRNA sequences were drawn from SILVA and PR2, with primer sequences, vector contamination, and potential chimeras removed from SILVA entries. Viral RNA coverage was established using NCBI Virus, classifying sequences into six viral RNA types (e.g., dsRNA, ssRNA+, ssRNA−) and filtering fragmented contigs shorter than 20 nt.

Deduplication and Filtering Pipeline. We developed a unified key-value header format to resolve schema discrepancies across the 16 databases. Redundancy was addressed in two stages: intra-database exact duplicates were removed based on SHA256 content hashing, followed by inter-database deduplication using `seqkit` (v2.8.2) with the `rmdup -by-seq` option and the xxHash algorithm for high-efficiency cross-corpus sequence comparison.

Finally, we applied a strict functional filter to retain 15 core RNA categories (e.g., mRNA, rRNA, tRNA, lncRNA, sRNA, viral RNA). Ambiguous categories (e.g., "misc_RNA", "pseudogene") and intermediate processing byproducts (e.g., miRNA loop regions) were excluded. Sequences shorter than 20 nt were also excluded. The final OpenRNA v1 dataset comprises 114,186,538 unique sequences, providing a non-redundant and biologically coherent training set.

4.2.2 Data processing and tokenization

Sequence Clustering and Sampling. To reduce data redundancy and enhance training diversity, we clustered the 114M cleaned sequences using `MMseqs2` with parameters set to 50% sequence identity (`MIN_SEQ_ID=0.5`) and 80% coverage (`COVERAGE=0.8`). This clustering produced 17,350,557 clusters, with 56.19% being singleton clusters and 37.65% being ultra-small clusters (2–10 sequences), indicating substantial redundancy in the data.

To improve exposure of rare RNA types while maintaining data quality, we employed an inverse-sqrt weighted sampling strategy: the sampling probability P_i for a cluster i is inversely proportional to the square root of its size n_i ($P_i \propto 1/\sqrt{n_i}$), enabling the model to effectively learn complex sequence patterns from rare RNA families in the long-tail distribution.

Sequence Tokenization. Each sequence was annotated with two orthogonal labels: (i) **RNA type**, assigned using source database metadata to one of 15 RNA family categories; and (ii) **taxonomic lineage**, mapped to the seven-level Greengenes hierarchy (`d__domain; p__phylum; c__class; o__order; f__family; g__genus; s__species`) via NCBI Taxonomy lookups on the NCBI accession or organism field of each record. Sequences lacking a resolvable taxonomy at any rank were assigned partial lineage strings (with missing levels omitted), retaining the deepest available annotation. More than 90% of sequences received complete RNA-type and at least genus-level lineage annotations.

Nucleotide sequences were tokenized at single-nucleotide resolution using a character-level vocabulary consisting of four canonical RNA nucleotides (A, U, G, C), a `5` token denoting the 5′-to-3′ reading direction, a `3` token denoting the 3′-to-5′ reading direction (reverse complement), special tokens (`<bos>`, `<bos_glm>`, `<eos>`, `<eos_span>`, `<pad>`, `[MASK]`), 15 RNA-type conditioning tokens (one per RNA class), and span sentinel tokens for GLM training. The tokenizer vocabulary has a total size of 114 tokens. Sequences exceeding 8,192 nt were truncated to fit EVA's context window.

4.2.3 Sequence diversity analysis within RNA families

To characterize intra-family sequence conservation across RNA types, we employed the Average Pairwise Identity (API) as the primary metric. Given a family of N sequences, API is defined as:

$$\text{API} = \frac{2}{N(N-1)} \sum_{i<j} \text{Identity}(s_i,\, s_j), \tag{5}$$

where $\text{Identity}(s_i, s_j)$ is the fraction of identical positions between sequences s_i and s_j after global pairwise alignment, computed as the number of matched positions divided by total alignment length. API is a continuous measure in $[0, 1]$ that captures subtle evolutionary trends missed by discrete clustering counts; unlike k-mer similarity or edit distance, it accounts for insertions and deletions through global alignment. High API indicates strong functional constraint; low API reflects greater sequence diversity or relaxed selective pressure.

Prior to API calculation, we applied 90% identity clustering to OpenRNA v1 using MMseqs2 `easy-linclust` (`-min-seq-id 0.9`, `-c 0.8`, `-cov-mode 1`), reducing the 114M sequences to 37,447,972 representatives while preserving family diversity. API was then computed for 11 RNA families: mRNA, miRNA, lncRNA, circRNA, piRNA, sRNA, viral RNA, rRNA, tRNA, snRNA, and snoRNA.

Because the largest family (mRNA, >27M sequences) makes exhaustive $O(N^2)$ pairwise computation infeasible, we adopted a non-parametric bootstrap resampling strategy: in each of $K = 50$ iterations, $n = 500$ sequences were drawn uniformly at random, aligned with MAFFT, and API was

computed for the subsample. The final estimate is the mean over all iterations, with 95% confidence intervals and coefficients of variation reported as uncertainty measures. This reduces complexity from $O(N^2)$ to $O(K \cdot n^2)$. Sensitivity analysis confirmed that $n = 500$, $K = 50$ achieves a 95% CI width of approximately $\pm 1\%$ with coefficient of variation below 5%. API values for all families are reported in Supplementary Table S3 and Figure S2A.

4.2.4 Phylogenetic tree construction of training data

To improve ecological coverage, multi-domain RNA representation, and lineage-level balance, a stratified sampling strategy was applied to characterize the ecological landscape of the dataset. Sequence redundancy was first reduced using MMseqs2 clustering at 50% sequence identity, and representative sequences were retained. To exclude biologically and computationally uninformative extremes, sequences outside the 80–4000 nucleotide length range were filtered to satisfy the model's context length constraint. A hierarchical Hybrid Proportional-Minimum Sampling framework was then employed to preserve diversity across RNA functional categories, lineage-level ecological grouping, and sequence length distribution, enabling a multi-dimensional overview of dataset ecology. Ultimately, 1746 sequences were selected for visualization to illustrate the ecological and distributional landscape of the dataset. Phylogenetic overview trees were constructed sequentially using MAFFT for multiple sequence alignment, ClipKit for alignment trimming, and FastTree for tree inference.

4.2.5 UMAP visualization of training data

To visualize the sequence-space diversity of OpenRNA v1, we computed 2-dimensional UMAP embeddings from species-level k-mer frequency profiles ($k = 1$–6) calculated over all RNA sequences. Prior to dimensionality reduction, feature vectors were re-weighted by domain (Archaea $\times 5$, Bacteria $\times 2$, Eukaryota $\times 1$) to compensate for imbalanced species counts, then ℓ_2-normalized. UMAP was run with `n_neighbors`$= 30$, `min_dist`$= 0.8$, `spread`$= 1.5$, and cosine distance. Points are colored at the phylum level (Bacteria: green; Eukaryota: blue; Archaea: red), with color intensity reflecting the log-scaled number of sequences per species. The resulting projection is shown in Supplementary Figure S1B.

4.3 Model Benchmarking and Applications

Through self-supervised learning on RNA sequences alone, EVA learns universal representations that enable a wide range of downstream applications, from zero-shot fitness prediction to controllable generative design.

4.3.1 Zero-shot Fitness Prediction

We benchmark EVA's predictive capabilities across RNA, DNA, and protein modalities, evaluating whether the model's likelihood scores correlate with experimental fitness.

Zero-shot ncRNA fitness prediction. We curated a benchmark of 13 deep mutational scanning (DMS) assays spanning three structured ncRNA classes—ribozymes, RNA aptamers, and transfer RNAs (tRNAs)—to evaluate EVA's zero-shot fitness prediction capability (Supplementary Table S6). Assays were sourced from published DMS studies and our previous experimental work[48], covering diverse functional readouts including catalytic activity, fluorescence, and stability. For each assay, we computed the Spearman rank correlation coefficient (ρ) between per-variant EVA log-likelihood scores and the experimentally measured functional activities.

Zero-shot mRNA fitness prediction. We curated five datasets capturing diverse aspects of mRNA function, including protein-coding fitness, pre-mRNA splicing regulation, and in-cell structural accessibility (Supplementary Table S7). Spearman rank correlation (ρ) between EVA log-likelihood scores and experimental measurements served as the primary evaluation metric, consistent with the ncRNA benchmark.

Zero-shot gene essentiality prediction. To evaluate whether EVA's sequence likelihoods encode information about gene function at the organism level, we assessed its ability to predict gene essentiality in a zero-shot setting, following the methodology of Evo 2[19] and Evo 1[18].

For bacteria, we compiled a dataset of essential and non-essential genes from the Database of Essential Genes (DEG)[49], cross-referenced with NCBI RefSeq genome annotations. Species were retained if they contained at least 10 essential and 10 non-essential annotated genes, yielding 42 bacterial species. For eukaryotes, we assembled essentiality annotations for 5 species (*Arabidopsis thaliana*, *Aspergillus fumigatus*, *Caenorhabditis elegans*, *Schizosaccharomyces pombe*, and *Saccharomyces cerevisiae*) from DEG[49]. The full list of species with their NCBI accession numbers is provided in Supplementary Table S9. Gene sequences were formatted with species-specific taxonomic lineage prefixes and RNA type tokens (e.g., `<bos>|`*lineage*`;<rna_mRNA>|5`*sequence*`3<eos>`).

For each gene, we computed the autoregressive log-likelihood of the wild-type coding sequence under the model:

$$\mathrm{LL}(\mathbf{s}) = \sum_{i=2}^{L} \log\, p(s_i \mid s_1, \ldots, s_{i-1}), \tag{6}$$

where $\mathbf{s} = (s_1, \ldots, s_L)$ is the tokenized input sequence. We then constructed a mutant sequence by inserting five tandem premature stop codons (`UAAUAAUAAUAAUAA`; 15 nt) near the $5'$ end of the coding region (at nucleotide position 12, or at one quarter of the gene length for short genes), and computed its log-likelihood $\mathrm{LL}(\mathbf{s}')$. The essentiality score was defined as the difference in log-likelihood between wild-type and mutant:

$$\Delta\mathrm{LL} = \mathrm{LL}(\mathbf{s}) - \mathrm{LL}(\mathbf{s}'), \tag{7}$$

where a higher ΔLL indicates that the premature stop insertion is more disruptive to the model's sequence likelihood, predicting greater essentiality. Sequences were truncated to a maximum of 8,192 tokens. We computed the area under the receiver operating characteristic curve (AUROC) for each species independently and averaged across species. We compared EVA (1.4B and 21M parameters) against Evo 2 (7B and 1B parameters) and two non-sequence baselines: for bacteria, a positional baseline based on distance from the chromosomal origin of replication (oriC); for eukaryotes, a GC content baseline.

Zero-shot protein fitness prediction. Although trained explicitly on RNA, EVA generalizes to protein fitness prediction. We evaluate this using 20 human protein DMS datasets that provide readily accessible nucleotide and protein sequences: ten from ProteinGym (ERBB2, NPC1, LYAM1, GLPA, PITX2, RBP1, RASH, CBPA2, PIN1, and DNJA1) and ten from DomainOme (Q9HC78, Q9UJQ4 [PF00096_383], Q8N1W1, Q9NU63, Q7Z5Q1, Q9ULZ3, Q13263, Q9UL15, O15151, and Q9UJQ4 [PF00096_627]). Full dataset identifiers and source databases are provided in Supplementary Table S8. We test whether the model captures protein-level evolutionary constraints via its exposure to coding sequences, computing Spearman rank correlation between per-variant EVA log-likelihood scores and experimentally measured fitness effects.

4.3.2 Controllable Generative Design

Benefiting from our proposed training strategy and extensive data, EVA enables controllable rational design of diverse RNAs across all domains of life.

Unconditional Generation. We first evaluate the model's baseline capability to generate valid, diverse RNA sequences without any specific constraints. This tests the model's grasp of general RNA grammar and its ability to cover the natural sequence manifold.

Conditional Autoregressive Generation. EVA supports fine-grained control over generation by conditioning on both **RNA-type tags** and **taxonomic lineage tags**. Conditioning on RNA type ensures the preservation of family-specific structural motifs (e.g., cloverleaf structures for tRNA). Further conditioning on taxonomic lineage enables species-aware design, such as optimizing mRNA codon usage to match the tRNA abundance of a specific host organism. This dual-conditioning capability is critical for practical applications like species-specific mRNA vaccine design.

Targeted Region Redesign (GLM). Using the Generalized Language Modeling (GLM) capability, users can perform targeted redesign of specific RNA regions (e.g., IRES elements) while keeping flanking sequences fixed. This enables modular optimization of functional elements within larger RNA scaffolds.

CLM generation quality: sequence similarity to nearest natural neighbour (Figure 3A). To assess how closely CLM-generated sequences resemble natural RNAs, we searched each generated sequence against the full OpenRNA v1 training corpus using MMseqs2 (nucleotide mode, `-search-type 3`, sensitivity $s = 7.5$, k-mer size 11, spaced k-mer mode, strand-aware, E-value ≤ 10, minimum alignment length 15 nt, up to 20 reported hits per query). For each generated sequence, we retained the database hit with the highest full-length weighted similarity, defined as:

$$\mathrm{sim}_{\mathrm{CLM}}(q, r) = \mathrm{pident}(q, r) \times \mathrm{qcov}(q, r), \tag{8}$$

where $\mathrm{pident}(q, r) \in [0, 1]$ is the local percent sequence identity of the best-scoring alignment and $\mathrm{qcov}(q, r) \in [0, 1]$ is the fraction of the query sequence covered by that alignment. Multiplying identity by coverage penalises hits in which only a short, highly similar sub-region is matched, ensuring that sequences with poor full-length coverage receive low scores even when the aligned portion is highly conserved. If no database hit was found for a query, its similarity was set to 0. The resulting per-sequence similarity values were aggregated into the violin plots shown in Figure 3A.

GLM infilling quality: reconstruction similarity to ground truth (Figure 3B). To quantify how accurately GLM reconstructs masked sequence spans, we compared each model-generated infill against the corresponding ground-truth fragment using global pairwise alignment. Alignments were performed with Biopython's `PairwiseAligner` in global mode (match score $+1$, mismatch score -1, gap-open penalty -2, gap-extend penalty -0.5), implementing the Needleman–Wunsch algorithm. The reconstruction similarity for a generated fragment $\hat{s}$ relative to its ground truth s^* was computed as:

$$\mathrm{sim}_{\mathrm{GLM}}(\hat{s}, s^*) = \frac{\mathrm{identities}(\hat{s}, s^*)}{|s^*|}, \tag{9}$$

where $\mathrm{identities}(\hat{s}, s^*)$ is the number of identical nucleotide pairs in the optimal global alignment and $|s^*|$ is the length of the ground-truth fragment. Normalising by ground-truth length rather than

alignment length ensures that insertions in the generated sequence are penalised. Per-span similarity values were aggregated into the violin plots shown in Figure 3B, stratified by RNA class.

Secondary structure statistics for distributional comparison (Figures 3D, 3F, S6, S7). To compare the biophysical properties of generated sequences against natural RNAs at scale, secondary structures were predicted using RNAfold from the ViennaRNA package (default parameters). RNAfold was chosen over slower folding tools because the distributional comparisons require predicting structures for large numbers of sequences (up to 200 generated sequences and thousands of natural reference sequences per RNA class or species), making computational efficiency essential. For each sequence, RNAfold provides the minimum free energy (MFE) dot-bracket secondary structure and the corresponding MFE value. The following structural statistics were extracted from the predicted dot-bracket string: MFE (kcal/mol), reflecting overall thermodynamic stability; stem/helix fraction, the proportion of nucleotides in Watson–Crick base pairs; loop fraction, the proportion of unpaired nucleotides; and base-pairing rate, the ratio of paired to total nucleotides summarizing global structural compactness. Distributions of these statistics for generated sequences ($n = 200$ per condition) and natural sequences were compared using kernel density estimates and visualized as overlaid density plots, enabling direct assessment of whether EVA's generative distribution recapitulates the structural properties of natural RNAs.

4.3.3 Zero-shot tRNA Generation and Structural Evaluation

To evaluate EVA's ability to capture three-dimensional structural constraints from sequence alone, we performed zero-shot conditional generation of tRNAs by conditioning the model on the tRNA type tag without any structure-based supervision.

For each generated sequence, we predicted its three-dimensional structure using AlphaFold3 and compared it against a curated set of 32 experimentally resolved monomeric tRNA structures retrieved from the RCSB Protein Data Bank (PDB IDs: 1EHZ, 1EVV, 1FIR, 1I9V, 1QZB, 1TN1, 1TN2, 1TRA, 1VTQ, 1YFG, 2K4C, 2TRA, 3A3A, 3BBV, 3CW5, 3CW6, 3L0U, 3TRA, 4P5J, 4TNA, 4TRA, 5L4O, 6CU1, 6TNA, 6UGG, 6Y3G, 7EQJ, 7KJU, 7VNW, 8UPT, 8UPY, 8V1H). Two metrics were computed for each generated sequence. For structural similarity, the AlphaFold3-predicted structure was compared against the same curated set of 32 RCSB tRNA structures and the maximum TM-score was taken. For sequence novelty, we randomly sampled 10,000 tRNA sequences from OpenRNA v1 and computed pairwise sequence identity via Needleman–Wunsch global alignment; the maximum identity across all 10,000 comparisons was taken as the sequence similarity score, with lower values indicating greater novelty relative to the training data. Representative generated sequences were additionally superimposed onto their closest RCSB structural matches for visual inspection of tertiary fold recapitulation.

4.3.4 Fine-tuning for Specialized RNA Types

To extend EVA's applicability to RNA families underrepresented in the training data, we implemented a lightweight fine-tuning strategy that preserves pre-trained RNA knowledge while adapting the generative distribution to user-defined functional RNA classes.

Conditioning token. We introduced a dedicated `Y_RNA` conditioning token reserved for user-defined RNA types. During fine-tuning, all sequences in the specialized dataset are assigned this token, enabling the model to associate a distinct generative mode with the target class without interfering with the 11 pre-trained RNA-type distributions.

Fine-tuning datasets. Two case studies were pursued: (i) RNA aptamers, synthetically engineered fluorogenic molecules absent from natural sequence databases, fine-tuned on $n = 30$ high-fluorescence sequences; and (ii) IscB omegaRNAs, CRISPR guide RNAs from a metagenomic Cas-related system, fine-tuned separately on high-editing-efficiency sequences from two guide variants (m16: $n = 42$; m17: $n = 44$).

Training procedure. Starting from the pre-trained EVA checkpoint, we performed continued autoregressive language-model training on each specialized dataset using a reduced learning rate and a small number of epochs, sufficient to shift the generative distribution while retaining general RNA sequence knowledge.

De novo sequence generation. After fine-tuning, sequences were sampled autoregressively conditioned on the `Y_RNA` token. For omegaRNA design, two independent models (one per guide variant) were each used to generate large libraries (m16: $n = 1{,}800$; m17: $n = 1{,}595$). Generated sequence lengths were verified against the expected length range of natural IscB omegaRNAs.

Mutation-count analysis. To characterize the edit distance between generated sequences and wildtype, we counted the number of mutated positions in each generated aptamer sequence ($n = 93$) relative to the wildtype and compared the resulting distribution to that of variants catalogued by deep mutational scanning (DMS). This analysis establishes whether fine-tuned generative models produce sequences with biologically plausible divergence levels, addressing the concern that de novo generation yields sequences too distant from wildtype to retain function.

Structural validation. Representative designed sequences from each case study were submitted to AlphaFold3 for complex structure prediction. For ωRNA designs, ternary complexes comprising the designed guide RNA, IscB protein, and target DNA substrate were predicted using AlphaFold3 in MSA mode with the experimentally resolved IscB–ωRNA–DNA structure (RCSB PDB: 7UTN) as a structural template.

To move beyond global confidence scores, we derived five domain-level geometric criteria from the IscB structural literature[25,26] and applied them as pass/fail filters across all predicted complexes (Supplementary Table S13): The five criteria cover: (1) HNH domain distance to the guide RNA, verifying productive nuclease positioning for target-strand cleavage; (2) contact of all four PLMP residues with the ωRNA (heavy-atom distance < 7 Å), required for complex stabilization; (3) straightness of the bridge helix (BH) α-helix; (4) fraction of the BH surface wrapped by the ωRNA, reflecting scaffold integrity; and (5) WED domain distance to the PAM-proximal DNA phosphate backbone, confirming correct PAM engagement. Designs satisfying all thresholds were shortlisted as structurally viable candidates and visualized from two opposing viewpoints (0° and 180° rotation) to confirm overall fold preservation.

4.4 Design and Optimization of RNA Vaccines

We applied EVA to optimize two major classes of RNA vaccines: linear mRNA vaccines and circular RNA (circRNA) vaccines.

4.4.1 Species-aware mRNA Codon Optimization

We formulated mRNA codon optimization as a synonymous substitution search problem, where the objective is to find the synonymous coding sequence that maximizes a composite score reflecting

both RNA structural stability and host-organism compatibility.

Scoring Signal. Traditional codon optimization relies on the Codon Adaptation Index (CAI) as the sequence-level objective. CAI is computed from per-codon frequency tables and treats each position independently, ignoring global sequence context and the interplay between codon choice, tRNA dynamics, elongation kinetics, and co-translational folding. As an alternative, we use the conditional log-likelihood assigned by EVA under species conditioning. Prepending the target species tag causes EVA to evaluate each sequence against a species-specific sub-manifold carved out of the broader RNA evolutionary manifold. The resulting likelihood captures global sequence context–including 5′UTR and 3′UTR compatibility, structural constraints, and organism-specific sequence preferences–in a single data-driven score, without requiring explicit generation. Our codon optimization framework therefore employs two complementary objectives in sequence: EVA log-likelihood (species-conditioned) guides candidate generation via beam search to confine exploration to the evolutionary manifold, while MFE computed by LinearFold evaluates the thermodynamic stability of the resulting candidates. This sequential two-objective design replaces the conventional CAI + MFE pipeline.

Optimization Algorithm. Codon optimization is performed via a sliding-window beam search combined with simulated annealing over the space of synonymous coding sequences. Starting from an initial set of $P = 8$ candidate CDS sequences (synonymous variants of the target protein), the algorithm iteratively proposes and evaluates local substitutions as follows.

At each iteration, $B = 8$ sequences are randomly drawn without replacement from the population as modification targets (*victims*). For each victim, a contiguous sliding window of $W = 8$ codons is placed at a uniformly random position within the CDS. The upstream context–comprising the 5′UTR and the CDS prefix upstream of the window–is formatted as an EVA conditioning prompt:

```
<bos>|[lineage];[RNA-type]|5'[context sequence]
```

where `[lineage]` encodes the Greengenes-format taxonomic prefix for the target host organism (e.g., `d__eukaryota;p__chordata;...;s__homo_sapiens` for *Homo sapiens*) and `[RNA-type]` specifies the target RNA class token (`<rna_mRNA>` or `<rna_circRNA>`).

EVA is then used to perform a beam search with beam width $k = 10$ over all valid synonymous codon assignments for the W window positions. At each codon step, every synonymous codon for the corresponding amino acid residue is scored by the conditional token log-probability that EVA assigns given the current prefix context; the k candidate codon sequences with the highest cumulative log-probability scores are retained. Beam searches for all B victims within a single iteration are batched onto the GPU and evaluated in a single forward pass to maximize throughput.

After beam search, the full-length sequence (5′UTR + CDS + 3′UTR) for each of the top-k beam candidates is submitted to LinearFold for MFE computation, with up to 64 calculations parallelized across CPU cores. The candidate with the lowest MFE among the k beam entries is selected as the proposal. Acceptance is then determined by the Metropolis criterion: if $\Delta E = \mathrm{MFE}_{\mathrm{proposal}} - \mathrm{MFE}_{\mathrm{current}} < 0$, the proposal replaces the victim unconditionally; otherwise, it is accepted with probability $\exp(-\Delta E/T)$. The annealing temperature T decays exponentially from $T_{\mathrm{init}} = 2.0$ to $T_{\mathrm{min}} = 0.01$ with a per-iteration cooling factor of $\gamma = 0.995$, enabling broad sequence-space exploration early in the run and progressive convergence toward thermodynamically stable solutions. The optimization runs for $N = 400$ iterations; the population member with the lowest MFE at termination is returned as the final optimized sequence.

This two-stage proposal mechanism jointly leverages two complementary signals: EVA beam search confines candidate exploration to synonymous sequences that reside within the evolutionary

manifold defined by the target RNA class and host organism, while MFE-based selection subsequently identifies the most thermodynamically favorable candidate within this evolutionarily constrained set. As an ablation baseline, a *Random (MFE-only)* method replaces EVA beam search with uniform random synonymous codon sampling for the W window while retaining the identical Metropolis MFE acceptance criterion, thereby isolating the contribution of EVA-guided proposals from the annealing procedure.

CAI Computation. The Codon Adaptation Index (CAI) for each optimized sequence is computed post hoc from human codon usage frequencies as an interpretable measure of host-organism compatibility. For each codon, its relative synonymous codon usage (RSCU) weight is normalized against the most frequently used synonymous codon for the same amino acid; CAI is then the geometric mean of these per-codon weights over all codons in the CDS. CAI serves as a secondary evaluation metric alongside MFE, providing a direct measure of alignment with host translation machinery that is independent of the EVA-guided optimization objective.

Evaluation Metrics. Optimized sequences were evaluated on two metrics: Codon Adaptation Index (CAI), measuring alignment with host codon usage bias, and Minimum Free Energy (MFE), computed by ViennaRNA to assess RNA folding stability (lower MFE indicates more stable secondary structure). We benchmarked this approach on four vaccine candidates: SARS-CoV-2 Spike, Varicella-Zoster Virus (VZV) gE, RABV-G, and HIV envelope protein.

4.4.2 ORF Optimization and IRES Generation for circRNA Vaccines

For circRNA vaccines, which lack $5'$ caps and poly(A) tails, design involves two distinct tasks: optimizing the Open Reading Frame (ORF) and designing the Internal Ribosome Entry Site (IRES) to drive cap-independent translation.

ORF Optimization. Similar to linear mRNA, we optimized the circRNA ORF via synonymous substitution. We evaluated performance on four systems: SARS-CoV-2 (Group-I intron and T4 ligase cyclization methods), Rabies, and CAR-T circRNAs.

De Novo IRES Redesign via GLM. We leveraged EVA's Generalized Language Modeling (GLM) capability for fill-in-the-middle design of IRES elements.

Task formulation. The original viral IRES region (CVB3) in each circRNA construct is treated as a masked span, while the flanking ORF and permuted-intron-exon (PIE) cyclization sequences are held fixed as left and right context. The GLM input to EVA is constructed by placing a `[MASK]` placeholder at the IRES position and conditioning on the circRNA type and *Homo sapiens* lineage tags. EVA then samples new IRES candidates autoregressively, conditioned on both flanking sequence contexts, enabling the model to co-optimize the IRES with its molecular environment rather than designing it in isolation. Multiple independent samples are drawn for each circRNA system (SARS and Rabies) to generate a candidate library.

Biophysical evaluation. Generated IRES candidates are evaluated on three structure-based metrics using ViennaRNA and LinearFold: ribosome accessibility (unpaired probability of the AUG start-codon region, with higher values indicating a more accessible entry site); inhibitory long-range interaction reduction (fraction of IRES nucleotides forming base pairs with distal ORF or cyclization sequences, where fewer contacts facilitate ribosome recruitment); and MFE of the full circRNA construct (more negative values reflect greater global stability). Designed IRES sequences are benchmarked against the original CVB3 IRES on all three metrics.

4.5 Model Interpretability with Sparse Autoencoders

4.5.1 Sparse Autoencoder Training

To decompose dense, distributed representations into interpretable features, we trained sparse autoencoders (SAEs) on model internal activations, following recent advances in mechanistic interpretability[37]. We trained SAEs on two EVA model scales (167M and 1.4B parameters) using distinct architectures, both on layer 13 activations extracted from one million RNA sequences sampled from a curated multi-species corpus. Layer 13 was selected as an intermediate layer where representations transition from local sequence features to higher-order biological abstractions[18].

ReLU SAE with L_1 sparsity (167M model). For the 167M-parameter model (hidden dimension $d = 448$), we trained a ReLU-based SAE following the InterPLM framework[50]. The encoder maps the centered input to an overcomplete latent space of dimensionality $D = 3{,}584$ (8× expansion):

$$\mathbf{f} = \mathrm{ReLU}(\mathbf{W}_{\mathrm{enc}}\,(\mathbf{x} - \mathbf{b}_{\mathrm{pre}}) + \mathbf{b}_{\mathrm{enc}})\,, \tag{10}$$

and the decoder reconstructs the input as $\hat{\mathbf{x}} = \mathbf{W}_{\mathrm{dec}}\,\mathbf{f} + \mathbf{b}_{\mathrm{pre}}$. The training loss is:

$$\mathcal{L} = \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 + \lambda\,\|\mathbf{f}\|_1\,, \tag{11}$$

with $\lambda = 0.06$ linearly warmed up over the first 5% of training steps. The model was optimized with ConstrainedAdam[50], which projects gradients to maintain unit ℓ_2-norm on decoder columns, preventing sparsity minimization via decoder norm scaling. Learning rate: 2×10^{-4} with 1,000 warmup steps, constant phase, and linear decay from 80% of training. Dead neurons (inactive for >2,500 consecutive steps) were periodically resampled every 5,000 steps. Training ran for 150,000 steps with a batch size of 16,384 tokens.

BatchTopK SAE (1.4B model). For the 1.4B-parameter model (hidden dimension $d = 1{,}024$), we adopted a BatchTopK SAE architecture[51] enforcing hard sparsity constraints. The encoder maps the centered input to $D = 8{,}192$ latent dimensions (8× expansion). A BatchTopK operation retains only the top $k \times B$ entries across the flattened $B \times D$ pre-activation matrix (batch size B, $k = 32$), enforcing a fixed average sparsity of 32 active features per token. The training objective is:

$$\mathcal{L} = \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 + \alpha \left\| \mathbf{W}_{\mathrm{dec}}^{(\mathrm{dead})} \mathrm{ReLU}\Big(\mathbf{W}_{\mathrm{enc}}^{(\mathrm{dead})}\,(\mathbf{x} - \hat{\mathbf{x}}) + \mathbf{b}_{\mathrm{enc}}^{(\mathrm{dead})}\Big) - (\mathbf{x} - \hat{\mathbf{x}}) \right\|_2^2, \tag{12}$$

where $\alpha = 0.03$ and the superscript (dead) restricts the encoder and decoder to latent dimensions with zero activation count, encouraging reuse of underutilized capacity. Optimization used Adam with a trapezoidal learning rate schedule (5% warmup to 1×10^{-4}, 90% constant, 5% decay), gradient norm clipped at 1.0. To avoid materializing the full activation dataset, per-token activations were streamed into a fixed-capacity reservoir buffer of 5×10^6 tokens, with training batches of 4,096 tokens randomly sampled. The final SAE achieved reconstruction loss ~ 0.76, average $L_0 = 32.0$ active features per token, and 100% feature utilization across all 8,192 latent dimensions.

4.5.2 Structure-Specific Feature Annotation of Therapeutic RNA Constructs

To assess whether SAE features capture functionally meaningful structural elements in engineered therapeutic constructs, we applied the BatchTopK SAE to annotated circRNA and mRNA vaccine sequences. For each construct, we passed the full sequence through EVA 1.4B, extracted layer 13 activations, and projected them through the SAE encoder ($k = 32$), yielding a per-nucleotide activation matrix $\mathbf{A} \in \mathbb{R}^{L \times D}$.

Region-specific feature selection. For each annotated functional region r (start s_r, end e_r), we identified the SAE feature best discriminating the target region from the remainder of the sequence via a penalty-weighted score:

$$S_j = \sum_{i=1}^{L} A_{ij} \cdot (4\, m_i - 3), \tag{13}$$

where $m_i = 1$ for $s_r \leq i < e_r$ and 0 otherwise. This assigns weight +4 to in-region activations and −3 to off-target activations. Features were selected by maximizing coverage (fraction of in-region positions where the feature is active) subject to a leakage threshold of $< 50\%$ (ratio of out-of-region to in-region activation); the highest-scoring feature was used when no feature met this criterion. Structural elements shorter than 50 nt were excluded.

circRNA constructs. We analyzed two circRNA vaccine candidates: a rabies virus glycoprotein construct (RABV-G; 2,348 nt) comprising splicing exons (E1+E2), a spacer, a CVB3 IRES (738 nt), and the RABV-G ORF (1,574 nt); and a SARS-CoV-2 RBD construct (1,695 nt) comprising a splicing exon (Exon2), a CVB3 IRES (741 nt), a SP+RBD coding sequence (762 nt), a GS linker, a T4 fibritin foldon domain, and a splicing exon (Exon1)[27]. Each construct was visualized as a polar plot reflecting its covalently closed topology, with per-nucleotide activation magnitudes shown as radial bars.

mRNA constructs. We analyzed four codon-optimized mRNA constructs sharing identical UTRs (5′UTR: 64 nt; 3′UTR: 94 nt) encoding distinct viral antigens: HIV-1 gp160 (2,580 nt CDS), influenza A/PR8 hemagglutinin (1,701 nt CDS), RABV-G (1,575 nt CDS), and varicella-zoster virus glycoprotein E (1,875 nt CDS). Applying the penalty-weighted scoring independently to each construct, the same three features were selected across all four mRNAs: feature f/1998 for the 5′UTR, feature f/4512 for the CDS, and feature f/6236 for the 3′UTR. This consistency across constructs with divergent coding sequences provides evidence that these features encode generalizable structural concepts rather than sequence-specific motifs.

4.5.3 UMAP Visualization of SAE Feature Space

To characterize the global organization of learned features and assess domain-level segregation, we performed UMAP visualization of the SAE decoder geometry, following the approach of Evo 2[19]. Each SAE feature j is associated with a decoder direction $\mathbf{d}_j \in \mathbb{R}^d$ (the j-th column of $\mathbf{W}_{\text{dec}}$). All $D = 3{,}584$ decoder directions from the 167M ReLU SAE were normalized to unit ℓ_2-norm, and UMAP was applied to the resulting $D \times d$ matrix (cosine distance, $n_{\text{neighbors}} = 15$, min_dist = 0.1).

To assign domain-level biological interpretation, we computed differential activation frequencies between eukaryotic and prokaryotic sequences. We sampled 100,000 sequences from each domain from the training corpus, passed them through EVA and the SAE at layer 13, and recorded the fraction of tokens at which each feature was active (activation > 0), yielding per-feature densities $\rho_j^{(\text{euk})}$ and $\rho_j^{(\text{prok})}$. The differential score is:

$$\Delta_j = \rho_j^{(\text{euk})} - \rho_j^{(\text{prok})}, \tag{14}$$

where $\Delta_j > 0$ indicates a eukaryote-biased feature and $\Delta_j < 0$ indicates a prokaryote-biased feature. Scores were clipped at the 98th percentile of $|\Delta_j|$ and linearly mapped to $[-1, 1]$ for visualization on a diverging orange-to-blue color scale.

4.5.4 Mixture-of-Experts Routing Analysis across RNA Biotypes

Mixture-of-Experts (MoE) architectures route each input token to a subset of specialized feed-forward sub-networks (experts) via a learned gating mechanism[45,52]. To assess whether different RNA biotypes induce systematically different expert routing patterns, we analyzed the routing behavior of EVA (167M parameters; 16 transformer layers, 8 experts per MoE layer, top-2 gating) using up to 50,000 sequences per RNA biotype sampled from a curated multi-species corpus spanning eight classes: mRNA, rRNA, tRNA, lncRNA, circRNA, miRNA, snoRNA, and snRNA. Sequences were truncated to 2,048 tokens and processed in mini-batches of 16.

Router probability extraction. For each MoE layer, we registered a forward hook on the gating module to capture its output logits before expert dispatch. Given an input token representation $\mathbf{h} \in \mathbb{R}^d$, the router computes:

$$p_e = \frac{\exp(\mathbf{w}_e^\top \mathbf{h})}{\sum_{e'=1}^{E} \exp(\mathbf{w}_{e'}^\top \mathbf{h})}, \tag{15}$$

where $\mathbf{w}_e$ is the learned weight vector for expert e and $E = 8$. Softmax-normalized routing probabilities were averaged across all tokens of a given RNA biotype to obtain a per-biotype, per-layer routing profile $\bar{\mathbf{p}} \in \mathbb{R}^E$.

Routing specialization quantification. To quantify RNA-type-specific expert specialization at each layer, we computed the mean standard deviation of routing probabilities across RNA biotypes:

$$V_\ell = \frac{1}{E} \sum_{e=1}^{E} \mathrm{Std}_{c \in \mathcal{C}} \left[\bar{p}_e^{(\ell,c)} \right], \tag{16}$$

where $\bar{p}_e^{(\ell,c)}$ is the mean routing probability for expert e at layer ℓ over all tokens of RNA biotype c, and $\mathcal{C}$ denotes the set of eight RNA classes. Higher V_ℓ values indicate greater divergence in expert utilization across RNA types. The analysis was performed both without lineage prefix (raw sequence only) and with species-specific taxonomic prefix prepended, to assess how lineage conditioning influences expert routing.

Acknowledgements

We thank Shuya Li, Xiaoliang Shi, Bibi Zhang and Xiaoyu Chen for helpful discussions. We thank the Lingang Laboratory for providing computational resources. We thank Chaoxiang Nan for valuable advice on model training. We thank Lan Li for her aesthetic guidance and suggestions on the schematic figures. We thank Wenhao Yang, Runze Ma, and Zhongmin Li for their support on benchmarking. We thank Aiyuan Zhangxv, Hanlin Yu, Xiangwen Gao, Xinyi Zhang and Songan Zhang for their moral support.

Conflict of interest

The authors declare no competing interests.

Author contribution

S.Z. conceived the project. S.Z. and Y.H. supervised the project. G.L. and Y.H. designed the model architecture. Y.H. developed the first version of the training code; G.L. contributed to the training code. Y.H. developed the first version of the inference code; G.L., W.X., X.M., and Yij.H. contributed to the inference code. G.L., A.C., and Y.H. curated and processed the pretraining dataset. Y.H. conducted model pretraining, mid-training, needle-in-a-haystack evaluation, and related evaluations. Y.H., X.M., Yij.H., A.C., G.L. and Y.T. performed zero-shot mutational effect prediction. W.X. and Y.H. evaluated the model's generative capabilities and performed the associated visualizations. Y.H. wrote the fine-tuning code and performed model fine-tuning. Y.H. and Y.X. conducted tRNA and RNA aptamer design, analysis, and visualization. Y.H., L.Z., and Y.X. performed CRISPR (ISCB omegaRNA) data collection, design, analysis, and visualization. Y.H. organized the mRNA and circRNA vaccine datasets. W.X. wrote the codon optimization code for mRNA and circRNA vaccines, conducted codon optimization experiments, and performed the associated visualizations. Y.H. wrote the circRNA IRES generation code and conducted circRNA IRES design, analysis, and visualization. A.C. and Y.H. implemented and trained the sparse autoencoders (SAEs); A.C. performed SAE feature discovery and visualization. Y.C., Z.W., and Y.H. developed the model introduction and usage website; G.L. provided the backend model interface and code. Q.S. performed evolutionary analysis and visualization of the training data. M.C. and Y.H. conceived and created all conceptual figures. Y.H. wrote the first draft of the manuscript. All authors contributed to writing the final manuscript.

Data and Materials Availability

To foster reproducibility and accelerate advancement in the RNA modeling community, we make all resources associated with this work publicly available.

- The OpenRNA dataset used to train EVA is available at:
  https://huggingface.co/datasets/GENTEL-Lab/OpenRNA-v1-114M
- All EVA model checkpoints spanning four scales (21M, 145M, 437M, and 1.4B parameters) are openly accessible at:
  https://huggingface.co/GENTEL-Lab/EVA
- Comprehensive training pipelines for both mixture-of-experts (MoE) and dense architectures, including pretraining, midtraining, and evaluation protocols, can be found at:
  https://github.com/GENTEL-lab/EVA/tree/main/training
- Fine-tuning workflows for downstream RNA design tasks are available at:
  https://github.com/GENTEL-lab/EVA/tree/main/finetune
- For interpretability analysis, we release Sparse Autoencoder (SAE) features and associated notebooks enabling mechanistic insights into EVA's internal representations:
  https://github.com/GENTEL-lab/EVA/tree/main/notebooks/interpretability_analysis
- Practical tools for RNA engineering applications, including fitness prediction, conditional language model (CLM)/guided language model (GLM) design, and directed evolution workflows, are provided at:
  https://github.com/GENTEL-lab/EVA/tree/main/tools

- Complete benchmark datasets and experimental results for RNA generation and codon optimization are archived at:
https://zenodo.org/records/19027142

Supplementary Materials

- Supplementary Figures S1 to S10.
- Supplementary Tables S1 to S14.
- Supplementary Data.

References

[1] Mihir Metkar, Christopher S. Pepin, and Melissa J. Moore. Tailor made: the art of therapeutic mRNA design. *Nature Reviews Drug Discovery*, 23:67–83, 2024. doi: 10.1038/s41573-023-00827-x.

[2] Thomas R. Cech and Joan A. Steitz. The Noncoding RNA Revolution—Trashing Old Rules to Forge New Ones. *Cell*, 157:77–94, 2014. doi: 10.1016/j.cell.2014.03.008.

[3] David P. Bartel. microRNAs in action: biogenesis, function and regulation. *Nature Reviews Genetics*, 24:816–833, 2023. doi: 10.1038/s41576-023-00611-y.

[4] Luisa Statello, Chun-Jie Guo, Ling-Ling Chen, and Maite Huarte. Long non-coding RNAs: definitions, functions, challenges and recommendations. *Nature Reviews Molecular Cell Biology*, 24:430–447, 2023. doi: 10.1038/s41580-022-00566-8.

[5] Charlotte J. Wright, Christopher W. J. Smith, and Chris D. Jiggins. Alternative splicing as a source of phenotypic diversity. *Nature Reviews Genetics*, 23:697–710, 2022. doi: 10.1038/s41576-022-00514-4.

[6] Jiayang Chen, Zhihang Hu, Siqi Sun, Qingxiong Tan, Yixuan Wang, Qinze Yu, Licheng Zong, Liang Hong, Jin Xiao, Tao Shen, Irwin King, and Yu Li. Interpretable RNA Foundation Model from Unannotated Data for Highly Accurate RNA Structure and Function Predictions, August 2022. URL `http://arxiv.org/abs/2204.00300`. arXiv:2204.00300 [q-bio].

[7] Rafael Josip Penić, Tin Vlašić, Roland G. Huber, Yue Wan, and Mile Šikić. RiNALMo: General-Purpose RNA Language Models Can Generalize Well on Structure Prediction Tasks. *Nature Communications*, 16(1):5671, July 2025. ISSN 2041-1723. doi: 10.1038/s41467-025-60872-5. URL `http://arxiv.org/abs/2403.00043`. arXiv:2403.00043 [q-bio].

[8] Xi Wang, Ruichu Gu, Zhiyuan Chen, Yongge Li, Xiaohong Ji, Guolin Ke, and Han Wen. UNI-RNA: UNIVERSAL PRE-TRAINED MODELS REVOLUTIONIZE RNA RESEARCH, July 2023. URL `http://biorxiv.org/lookup/doi/10.1101/2023.07.11.548588`.

[9] Yekaterina Shulgina, Marena I. Trinidad, Conner J. Langeberg, Hunter Nisonoff, Seyone Chithrananda, Petr Skopintsev, Amos J. Nissley, Jaymin Patel, Ron S. Boger, Honglue Shi, Peter H. Yoon, Erin E. Doherty, Tara Pande, Aditya M. Iyer, Jennifer A. Doudna, and Jamie H. D. Cate. RNA language models predict mutations that improve RNA function. *Nature Communications*, 15(1):10627, December 2024. ISSN 2041-1723. doi: 10.1038/s41467-024-54812-y. URL `https://www.nature.com/articles/s41467-024-54812-y`.

[10] Shuxian Zou, Tianhua Tao, Sazan Mahbub, Caleb N. Ellington, Robin Algayres, Dian Li, Yonghao Zhuang, Hongyi Wang, Le Song, and Eric P. Xing. A Large-Scale Foundation Model for RNA Function and Structure Prediction, November 2024. URL http://biorxiv.org/lookup/doi/10.1101/2024.11.28.625345.

[11] Yanyi Chu, Dan Yu, Yupeng Li, Kaixuan Huang, Yue Shen, Le Cong, Jason Zhang, and Mengdi Wang. A 5′ UTR language model for decoding untranslated regions of mRNA and function predictions. *Nature Machine Intelligence*, 6(4):449–460, April 2024. ISSN 2522-5839. doi: 10.1038/s42256-024-00823-9. URL https://www.nature.com/articles/s42256-024-00823-9.

[12] Guipeng Li, Feifei Jiang, Junhao Zhu, Huanhuan Cui, Zefeng Wang, and Wei Chen. HydraRNA: a hybrid architecture based full-length RNA language model. *Genome Biology*, 26(1):383, November 2025. ISSN 1474-760X. doi: 10.1186/s13059-025-03853-7. URL https://genomebiology.biomedcentral.com/articles/10.1186/s13059-025-03853-7.

[13] Aidan T. Riley, James M. Robson, and Alexander A. Green. Generative and predictive neural networks for the design of functional RNA molecules, July 2023. URL http://biorxiv.org/lookup/doi/10.1101/2023.07.14.549043.

[14] Furkan Ozden, Sina Barazandeh, Dogus Akboga, Sobhan Shokoueian Tabrizi, Urartu Ozgur Safak Seker, and A. Ercument Cicek. RNAGEN: A generative adversarial network-based model to generate synthetic RNA sequences to target proteins, July 2023. URL http://biorxiv.org/lookup/doi/10.1101/2023.07.11.548246.

[15] Yichong Zhao, Kenta Oono, Hiroki Takizawa, and Masaaki Kotera. GenerRNA: A generative pre-trained language model for de novo RNA design. *PLOS ONE*, 19(10):e0310814, October 2024. ISSN 1932-6203. doi: 10.1371/journal.pone.0310814. URL https://dx.plos.org/10.1371/journal.pone.0310814.

[16] Zaixi Zhang, Ruofan Jin, Linlin Chao, Guangxue Xu, Yikun Zhang, Guowei Zhou, Di Yin, Yingqing Guo, Yaqi Fu, Yukang Yang, Kaixuan Huang, Xiaotong Wang, Junze Zhang, Yujie Yang, Qirong Yang, Ziyao Xu, E Weinan, Ruhong Zhou, Xiaoming Zhang, Mengdi Wang, and Le Cong. RNAGenesis: A Generalist Foundation Model for Functional RNA Therapeutics, December 2024. URL http://biorxiv.org/lookup/doi/10.1101/2024.12.30.630826.

[17] Ke Chen, Thomas Litfin, Jaswinder Singh, Jian Zhan, and Yaoqi Zhou. MARS and RNAcmap3: The Master Database of All Possible RNA Sequences Integrated with RNAcmap for RNA Homology Search. *Genomics, Proteomics & Bioinformatics*, 22(1):qzae018, May 2024. ISSN 1672-0229, 2210-3244. doi: 10.1093/gpbjnl/qzae018. URL https://academic.oup.com/gpb/article/doi/10.1093/gpbjnl/qzae018/7617691.

[18] Eric Nguyen, Michael Poli, Matthew G. Durrant, Brian Kang, Dhruva Katrekar, David B. Li, Liam J. Bartie, Armin W. Thomas, Samuel H. King, Garyk Brixi, Jeremy Sullivan, Madelena Y. Ng, Ashley Lewis, Aaron Lou, Stefano Ermon, Stephen A. Baccus, Tina Hernandez-Boussard, Christopher Re, Patrick D. Hsu, and Brian L. Hie. Sequence modeling and design from molecular to genome scale with Evo. *Science*, 386(6723):eado9336, November 2024. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.ado9336. URL https://www.science.org/doi/10.1126/science.ado9336.

[19] Garyk Brixi, Matthew G. Durrant, Jerome Ku, Michael Poli, Greg Brockman, Daniel Chang, Gabriel A. Gonzalez, Samuel H. King, David B. Li, Aditi T. Merchant, Mohsen Naghipourfar,

Eric Nguyen, Chiara Ricci-Tam, David W. Romero, Gwanggyu Sun, Ali Taghibakshi, Anton Vorontsov, Brandon Yang, Myra Deng, Liv Gorton, Nam Nguyen, Nicholas K. Wang, Etowah Adams, Stephen A. Baccus, Steven Dillmann, Stefano Ermon, Daniel Guo, Rajesh Ilango, Ken Janik, Amy X. Lu, Reshma Mehta, Mohammad R.K. Mofrad, Madelena Y. Ng, Jaspreet Pannu, Christopher Re, Jonathan C. Schmok, John St. John, Jeremy Sullivan, Kevin Zhu, Greg Zynda, Daniel Balsam, Patrick Collison, Anthony B. Costa, Tina Hernandez-Boussard, Eric Ho, Ming-Yu Liu, Thomas McGrath, Kimberly Powell, Dave P. Burke, Hani Goodarzi, Patrick D. Hsu, and Brian L. Hie. Genome modeling and design across all domains of life with Evo 2, February 2025. URL http://biorxiv.org/lookup/doi/10.1101/2025.02.18.638918.

[20] Ning Sun, Shuxian Zou, Tianhua Tao, Sazan Mahbub, Dian Li, Yonghao Zhuang, Hongyi Wang, Xingyi Cheng, Le Song, and Eric P. Xing. Mixture of Experts Enable Efficient and Effective Protein Understanding and Design, December 2024. URL http://biorxiv.org/lookup/doi/10.1101/2024.11.29.625425.

[21] Aadyot Bhatnagar, Sarthak Jain, Joel Beazer, Samuel C Curran, Alexander M Hoffnagle, Kyle Ching, Michael Martyn, Stephen Nayfach, Jeffrey A Ruffolo, and Ali Madani. Scaling unlocks broader generation and deeper functional understanding of proteins. *arXiv*, 2024.

[22] Brian L. Hie, Varun R. Shanker, Duo Xu, Theodora U. J. Bruun, Payton A. Weidenbacher, Shaogeng Tang, Wesley Wu, John E. Pak, and Peter S. Kim. Efficient evolution of human antibodies from general protein language models. *Nature Biotechnology*, 42(2):275–283, February 2024. ISSN 1087-0156, 1546-1696. doi: 10.1038/s41587-023-01763-2. URL https://www.nature.com/articles/s41587-023-01763-2.

[23] Varun R. Shanker, Theodora U. J. Bruun, Brian L. Hie, and Peter S. Kim. Unsupervised evolution of protein and antibody complexes with a structure-informed language model. *Science*, 385(6704):46–53, July 2024. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.adk8946. URL https://www.science.org/doi/10.1126/science.adk8946.

[24] Ali Madani, Ben Krause, Eric R. Greene, Subu Subramanian, Benjamin P. Mohr, James M. Holton, Jose Luis Olmos, Caiming Xiong, Zachary Z. Sun, Richard Socher, James S. Fraser, and Nikhil Naik. Large language models generate functional protein sequences across diverse families. *Nature Biotechnology*, 41(8):1099–1106, August 2023. ISSN 1087-0156, 1546-1696. doi: 10.1038/s41587-022-01618-2. URL https://www.nature.com/articles/s41587-022-01618-2.

[25] Qingquan Xiao, Guoling Li, Dingyi Han, Haoqiang Wang, Mingyu Yao, Tingting Ma, Jingxing Zhou, Yu Zhang, Xiumei Zhang, Bingbing He, Yuan Yuan, Linyu Shi, Tong Li, Hui Yang, Jinhai Huang, and Hainan Zhang. Engineered IscB–omegaRNA system with expanded target range for base editing. *Nature Chemical Biology*, 21(1):100–108, January 2025. ISSN 1552-4450, 1552-4469. doi: 10.1038/s41589-024-01706-1. URL https://www.nature.com/articles/s41589-024-01706-1.

[26] Kazuki Kato, Sae Okazaki, Soumya Kannan, Han Altae-Tran, F. Esra Demircioglu, Yukari Isayama, Junichiro Ishikawa, Masahiro Fukuda, Rhiannon K. Macrae, Tomohiro Nishizawa, Kira S. Makarova, Eugene V. Koonin, Feng Zhang, and Hiroshi Nishimasu. Structure of the IscB-ωRNA ribonucleoprotein complex, the likely ancestor of CRISPR-Cas9. *Nature Communications*, 13(1):6719, November 2022. ISSN 2041-1723. doi: 10.1038/s41467-022-34378-3. URL https://www.nature.com/articles/s41467-022-34378-3.

[27] Liang Qu, Zongyi Yi, Yong Shen, Liangru Lin, Feng Chen, Yiyuan Xu, Zeguang Wu, Huixian Tang, Xiaoxue Zhang, Feng Tian, Chunhui Wang, Xia Xiao, Xiaojing Dong, Li Guo, Shuaiyao Lu, Chengyun Yang, Cong Tang, Yun Yang, Wenhai Yu, Junbin Wang, Yanan Zhou, Qing Huang, Ayijiang Yisimayi, Shuo Liu, Weijin Huang, Yunlong Cao, Youchun Wang, Zhuo Zhou, Xiaozhong Peng, Jianwei Wang, Xiaoliang Sunney Xie, and Wensheng Wei. Circular RNA vaccines against SARS-CoV-2 and emerging variants. *Cell*, 185(10):1728–1744.e16, May 2022. ISSN 00928674. doi: 10.1016/j.cell.2022.03.044. URL https://linkinghub.elsevier.com/retrieve/pii/S0092867422003944.

[28] Robert Chen, Sean K. Wang, Julia A. Belk, Laura Amaya, Zhijian Li, Angel Cardenas, Brian T. Abe, Chun-Kan Chen, Paul A. Wender, and Howard Y. Chang. Engineering circular RNA for enhanced protein production. *Nature Biotechnology*, 41(2):262–272, February 2023. ISSN 1087-0156, 1546-1696. doi: 10.1038/s41587-022-01393-0. URL https://www.nature.com/articles/s41587-022-01393-0.

[29] He Zhang, Liang Zhang, Ang Lin, Congcong Xu, Ziyu Li, Kaibo Liu, Boxiang Liu, Xiaopin Ma, Fanfan Zhao, Huiling Jiang, Chunxiu Chen, Haifa Shen, Hangwen Li, David H. Mathews, Yujian Zhang, and Liang Huang. Algorithm for optimized mRNA design improves stability and immunogenicity. *Nature*, 621(7978):396–403, September 2023. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-023-06127-z. URL https://www.nature.com/articles/s41586-023-06127-z.

[30] Adibvafa Fallahpour, Vincent Gureghian, Guillaume J. Filion, Ariel B. Lindner, and Amir Pandi. CodonTransformer: a multispecies codon optimizer using context-aware neural networks. *Nature Communications*, 16(1):3205, April 2025. ISSN 2041-1723. doi: 10.1038/s41467-025-58588-7. URL https://www.nature.com/articles/s41467-025-58588-7.

[31] Zilin Ren, Lili Jiang, Yaxin Di, Dufei Zhang, Jianli Gong, Jianting Gong, Qiwei Jiang, Zhiguo Fu, Pingping Sun, Bo Zhou, and Ming Ni. CodonBERT: a BERT-based architecture tailored for codon optimization using the cross-attention mechanism. *Bioinformatics*, 40(7):btae330, July 2024. ISSN 1367-4803, 1367-4811. doi: 10.1093/bioinformatics/btae330. URL https://academic.oup.com/bioinformatics/article/doi/10.1093/bioinformatics/btae330/7681883.

[32] Sajad Darabi, Fan Cao, Mohsen Naghipourfar, Sara Rabhi, Ankit Sethia, Kyle Gion, Jasleen Grewal, Jonathan Cohen, William J Greenleaf, and Laksshman Sundaram. Learning the Language of Codon Translation with CodonFM.

[33] James (DR BTD) BIP-US-R Heuschkel. Advancing Codon Language Modeling with Synonymous Codon Constrained Masking.

[34] Binita Rajbanshi. codonGPT: Reinforcement learning on a generative language model optimizes RNA sequences under biological constraints.

[35] He Zhang, Hailong Liu, Yushan Xu, Haoran Huang, Yiming Liu, Jia Wang, Yan Qin, Haiyan Wang, Lili Ma, Zhiyuan Xun, Xuzhuang Hou, Timothy K. Lu, and Jicong Cao. Deep generative models design mRNA sequences with enhanced translational capacity and stability. *Science*, page eadr8470, August 2025. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.adr8470. URL https://www.science.org/doi/10.1126/science.adr8470.

[36] Yupeng Li, Fan Wang, Jiaqi Yang, Zirong Han, Linfeng Chen, Wenbing Jiang, Hao Zhou, Tong Li, Zehua Tang, Jianxiang Deng, Xin He, Gaofeng Zha, Zhaoyu Hu, Yong Hu, Linping

Wu, Changyou Zhan, Caijun Sun, Yao He, and Zhi Xie. Deep generative optimization of mRNA codon sequences for enhanced mRNA translation and therapeutic efficacy. *Nature Communications*, 16(1):9957, November 2025. ISSN 2041-1723. doi: 10.1038/s41467-025-64894-x. URL https://www.nature.com/articles/s41467-025-64894-x.

[37] Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. SPARSE AUTOENCODERS FIND HIGHLY INTER- PRETABLE FEATURES IN LANGUAGE MODELS. 2024.

[38] Samuel H King, Claudia L Driscoll, David B Li, Daniel Guo, Aditi T Merchant, Garyk Brixi, Max E Wilkinson, and Brian L Hie. Generative design of novel bacteriophages with genome language models.

[39] Dana Arad, Aaron Mueller, and Yonatan Belinkov. SAEs Are Good for Steering – If You Select the Right Features, May 2025. URL http://arxiv.org/abs/2505.20063. arXiv:2505.20063 [cs].

[40] Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering Language Models With Activation Engineering, October 2024. URL http://arxiv.org/abs/2308.10248. arXiv:2308.10248 [cs].

[41] Jacob Dunefsky, Philippe Chlenski, and Neel Nanda. Transcoders Find Interpretable LLM Feature Circuits, November 2024. URL http://arxiv.org/abs/2406.11944. arXiv:2406.11944 [cs].

[42] Mohammad Bavarian, Heewoo Jun, Nikolas Tezak, John Schulman, Christine McLeavey, Jerry Tworek, and Mark Chen. Efficient Training of Language Models to Fill in the Middle, July 2022. URL http://arxiv.org/abs/2207.14255. arXiv:2207.14255 [cs].

[43] Raymond Li, Loubna Ben Allal, Yangtian Zi, Niklas Muennighoff, Denis Kocetkov, Chenghao Mou, Marc Marone, Christopher Akiki, Jia Li, Jenny Chim, Qian Liu, Evgenii Zheltonozhskii, Terry Yue Zhuo, Thomas Wang, Olivier Dehaene, Mishig Davaadorj, Joel Lamy-Poirier, Joao Monteiro, Oleh Shliazhko, Nicolas Gontier, Nicholas Meade, Armel Randy, Ming-Ho Yee, Logesh Kumar Umapathi, Jian Zhu, Benjamin Lipkin, Muhtasham Oblokulov, Zhiruo Wang, Rudra Murthy, Jason Stillerman, Siva Sankalp Patel, Dmitry Abulkhanov, Marco Zocca, Manan Dey, Zhihan Zhang, Nour Fahmy, Urvashi Bhattacharyya, Suriya Gunasekar, Wenhao Yu, Swayam Singh, Sasha Luccioni, Paulo Villegas, Maxim Kunakov, Fedor Zhdanov, Manuel Romero, Tony Lee, Nadav Timor, Jennifer Ding, Claire Schlesinger, Hailey Schoelkopf, Jan Ebert, Tri Dao, Mayank Mishra, Alex Gu, Jennifer Robinson, Carolyn Jane Anderson, and Brendan Dolan-Gavitt. StarCoder: may the source be with you!

[44] Bo Chen, Xingyi Cheng, Pan Li, Yangli-ao Geng, Jing Gong, Shen Li, Zhilei Bei, Xu Tan, Boyan Wang, Xin Zeng, Chiming Liu, Aohan Zeng, Yuxiao Dong, Jie Tang, and Le Song. xTrimoPGLM: unified 100-billion-parameter pretrained transformer for deciphering the language of proteins. *Nature Methods*, 22(5):1028–1039, May 2025. ISSN 1548-7091, 1548-7105. doi: 10.1038/s41592-025-02636-z. URL https://www.nature.com/articles/s41592-025-02636-z.

[45] William Fedus, Barret Zoph, and Noam Shazeer. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity, June 2022. URL http://arxiv.org/abs/2101.03961. arXiv:2101.03961 [cs].

[46] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling Laws for Neural Language Models, January 2020. URL http://arxiv.org/abs/2001.08361. arXiv:2001.08361 [cs].

[47] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training Compute-Optimal Large Language Models, March 2022. URL http://arxiv.org/abs/2203.15556. arXiv:2203.15556 [cs].

[48] Yanjie Huang, Bibi Zhang, Jack X Chen, Runze Ma, Zhongyue Zhang, Xianjun Chen, Yi Yang, and Shuangjia Zheng. Capturing Natural Evolution in Function-guided RNA Design via Genomic Foundation Models, April 2025. URL https://www.biorxiv.org/content/10.1101/2025.04.08.647793. bioRxiv:2025.04.08.647793.

[49] R. Zhang. DEG: a database of essential genes. *Nucleic Acids Research*, 32(90001):271D–272, January 2004. ISSN 1362-4962. doi: 10.1093/nar/gkh024. URL https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkh024.

[50] Elana Simon and James Zou. InterPLM: Discovering Interpretable Features in Protein Language Models via Sparse Autoencoders, November 2024. URL http://biorxiv.org/lookup/doi/10.1101/2024.11.14.623630.

[51] Bart Bussmann, Patrick Leask, and Neel Nanda. BatchTopK Sparse Autoencoders, December 2024. URL http://arxiv.org/abs/2412.06410. arXiv:2412.06410 [cs].

[52] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer, January 2017. URL http://arxiv.org/abs/1701.06538. arXiv:1701.06538 [cs].

[53] Robert C. Spitale and Danny Incarnato. Probing the dynamic RNA structurome and its functions. *Nature Reviews Genetics*, 24:178–196, 2023. doi: 10.1038/s41576-022-00546-w.

[54] Deepak L. Bhatt and Deepak L. Bhatt. Regulatory RNA: from molecular insights to therapeutic frontiers. *Experimental & Molecular Medicine*, 56:515–524, 2024. doi: 10.1038/s12276-024-01267-2.