gRNAdeX: eXpressive, Biologically-eXtensible gRNAde

Martina Lapera Sancho*
University of Cambridge
ml2169@cam.ac.uk

José Antonio Franca Ibáñez*
University of Cambridge
jaf98@cam.ac.uk

Abstract

Inverse RNA folding—the challenge of designing RNA sequences that reliably adopt a prescribed secondary or tertiary structure—is pivotal for applications ranging from mRNA vaccines and riboswitches to RNA-based nanostructures. Despite recent advances, existing methods such as gRNAde encounter limitations in expressivity, representation, decoding robustness, and the incorporation of biological constraints. In this work, we introduce gRNAdeX, an enhanced RNA design framework that addresses these challenges through four key innovations. Experimental evaluations on a reduced dataset demonstrate that gRNAdeX outperforms the baseline in both sequence recovery and structural alignment, marking a significant step toward more robust and biologically plausible RNA design.

Statement of contribution

We, Martina Lapera Sancho and José Antonio Franca Ibáñez of the University of Cambridge, jointly declare that our work towards this project has been executed as follows:

- **Martina Lapera Sancho** contributed to the implementation of sampling techniques and biologically-motivated constraints, conducted experiments, created visualizations, and was involved in writing and reviewing the report.
- **José Antonio Franca Ibáñez** contributed to the design and implementation of architectural modifications to the model (encoder, decoder, and pooling components), mathematical formulation and was involved in writing and reviewing the report.

We both independently submit identical copies of this paper, certifying this statement to be correct.

GitHub repository with commit log

The companion source code for our project may be found at: https://github.com/antoniofrancaib/gRNAdeX.

1 Introduction

Designing RNA sequences that fold into desired secondary or tertiary structures—known as the *inverse RNA folding problem*—is a fundamental challenge at the intersection of computational biology and generative modeling. This problem underlies a growing number of applications, from programmable therapeutics (e.g., mRNA vaccines) and gene regulation (e.g., riboswitches, CRISPR guide RNAs) to RNA-based nanostructures in synthetic biology. Yet, inverse folding remains computationally formidable due to the vast combinatorial search space of nucleotide sequences and the complex, many-to-many relationship between sequences and their folded structures.

Recent advances in machine learning have enabled the development of generative models that map structured inputs (e.g., RNA backbones) to nucleotide sequences. Among these, `gRNAde` [1]

*Equal contribution.

Martina Lapera Sancho & José Antonio Franca Ibáñez, gRNAdeX: eXpressive, Biologically-eXtensible gRNAde. *Geometric Deep Learning (L65), University of Cambridge*.

introduced a promising geometric graph-based approach, modeling RNA conformations as 3D graphs with rich vector-valued features, and using GVP-based neural networks to generate sequences that respect both local geometry and global structure. However, several important challenges remain unresolved.

First, while GVP layers enforce local equivariance to geometric transformations, their fixed inductive biases may limit expressivity, especially in capturing long-range dependencies or subtle structural variations. Second, standard pooling methods—such as averaging across conformations—can be non-injective, thereby collapsing distinct inputs into the same representation and breaking universality. Third, decoding strategies often rely on greedy autoregressive sampling, which can propagate early errors and lead to suboptimal outputs. Finally, most models are agnostic to biological priors, generating sequences that are statistically plausible but biologically implausible, violating known constraints such as forbidden motifs (e.g., nullomers) or type-specific biases.

In this work, we present `gRNAdeX`, an enhanced RNA design model that builds on the geometric backbone of `gRNAde` while addressing its key limitations through a principled and biologically informed lens. Our proposed contributions fall under four complementary pillars:

- **Geometric Expressivity:** We analyze the Lipschitz properties of the GVP encoder to assess sensitivity and contractiveness, and introduce a parallel attention mechanism to capture global dependencies without sacrificing equivariance.
- **Universal Representation:** We replace simple averaging with a multi-moment tensor pooling scheme, enabling injective set encoding and ensuring theoretical universality under group symmetries relevant to RNA structure.
- **Robust Sampling:** We redesign the decoding process with beam search and adaptive sampling strategies (top-k, nucleus, min-p), reducing exposure bias and improving sequence quality across diverse structures.
- **Biological Constraints:** We incorporate a pipeline that identifies RNA-specific Minimally Absent Words (MAWs) to block biologically implausible motifs during sequence generation, giving researchers greater precision and control over the output. By offering explicit, user-defined control over forbidden motifs, our tunable system bridges the gap between structural intent and sequence design—empowering researchers to generate RNA sequences that are not just plausible, but purpose-built.

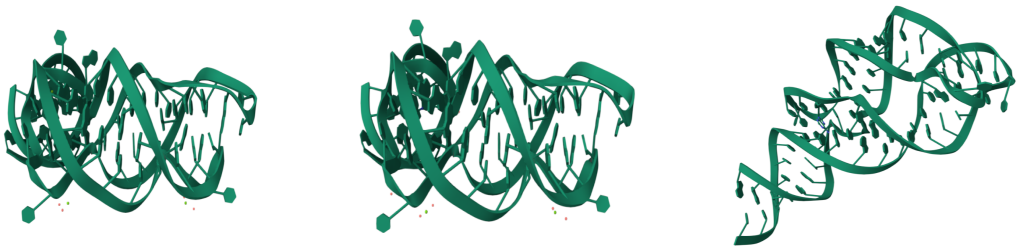
Together, these modifications yield a modular and biologically grounded framework for RNA sequence generation that is expressive, theoretically principled, and practically robust. Despite limited compute resources, our enhanced model demonstrates improved sequence recovery and structural alignment compared to the original `gRNAde` baseline. The contributions introduced here not only improve performance, but also broaden the adaptability and biological plausibility of structure-conditioned RNA generation.

2 Background

The RNA molecule is constituted by a set of ribose sugars attached to nitrogenous bases and phosphate groups [2]. Amongst other factors, the chemical composition of RNA differs from that of DNA in its nitrogenous bases (which contain the pyrimidine molecule of Uracile, also denoted as U, instead of DNA's Thymine, also denoted as T) and its ribose sugar (thereby the names, Rybonucleic Acid, or RNA, and Deoxyribonucleic Acid, or DNA). This ribose compound in fact contains an additional hydroxyl (-OH) group in carbon 2 (C2) [2, 3] which provides the RNA molecule with distinct properties – most importantly, RNA's distinct flexibility or lability [3].

In the context of the inverse folding problem [4], this added "flexibility" in the RNA molecule increases the search space for primary sequences (or nucleotide-based sequences) [5, 6] that are compatible with a given structure. Although this "search" problem has been extensively studied in the case of secondary sequences [7–9], these methods lack notions of the molecule's specific conformational structure and are in fact deeemed impractical for larger RNA sequences (li.e., some large ribosomal RNAs can be of lengths of 3,500 nucleotides [10]).

Similar to proteins [11], RNA function is highly dictated by its 3D conformation. By leveraging only on secondary structure inputs, existing methods [12, 13] cannot benefit from the enriched 3D



(a) PDB ID: 5E54. **(b)** PDB ID: 5SWD. **(c)** PDB ID: 5SWE.

Figure 1: Adedine riboswitches with equivalent secondary structure information and slightly distinct 3D conformations obtained from the Protein Database Bank [15] (in apo, intermediate conformations and holo conformations).

atom-positional information of distinct RNA conformations [14]. Some examples of molecules with different 3D conformations (or states) are the apo structures of the adenine riboswitch aptamer domain (see **Figure** 1a), the adenine riboswitch aptamer in an intermediate-bound state (see **Figure** 1b) and the ligand-bound structure of adenine riboswitch aptamer domain (see **Figure** 1c).

gRNAde [1] constitutes a state-of-the-art solution to the RNA inverse-folding problem. Through the use of SE(3)-Equivariant Graph Neural Network Layers [16] and autoregressive decoding, gRNAde maintains the physical properties of RNA molecules, achieving top performance in both, single-state and multi-state RNA design, which was previously not possible with methods such as Rosetta [17] or ViennaRNA [12] (the predecessor of Rosetta). Other generative modeling methods conditioned on RNA backbone structures are RiboDiffusion [5], or RNAFlow [18].

Despite their advancements, these generative models still face a significant search-space challenge, necessitating the generation of multiple candidate sequences (typically 16). In contrast, approaches like RNA-DCGen [19] leverage on pretrained RNA language models (e.g., BiRNA-BERT [20] or RiNALMo [21]) that have been fine-tuned on structural or functional data. These models capture extensive "learned biological priors" from large RNA corpora, enabling RNA-DCGen to more efficiently generate sequences consistent with the target constraints [19]. However, relying on these pretrained distributions can lead to incomplete coverage of the design space sequence, since any biases in the training set—such as over-representing certain RNA families and under-representing unusual motifs—may lead the model to overlook viable designs outside its learned distribution. In this landscape, gRNAdeX introduces biologically informed priors or constraints. By focusing on a narrower, biologically relevant search space, incorporating 3D-enriched information, and ensuring a robust exploration of the possible sequence space, it more effectively guides the final RNA 3D-to-sequence mapping.

3 Methods

In this work, we propose a series of methodological advancements that enhance the original gRNAde architecture. Our contributions are motivated by these central aims: (i) to increase the model's expressivity; (ii) to guarantee universality; (iii) to integrate robust sampling strategies that mitigate error propagation; and (iv) to embed biologically informed constraints (e.g., minimal absent words, typed conditioning) for more realistic RNA designs. For a detailed overview of the blocks/mappings used in gRNAde, see **Table** 5. **Figure** 2 depicts the new gRNAdeX architecture, with the highlighted blocks indicating the novel contributions introduced in this work. The following sections provide a more detailed overview of the motivation and implementation of these contributions.

3.1 Encoder Expressivity

To improve the representational capacity of the model, we performed a detailed analysis of the Lipschitz continuity of the encoder component. This allowed us to assess the potential contractiveness of the overall model mapping, aiming to mitigate underperformance in edge-case scenarios. Note

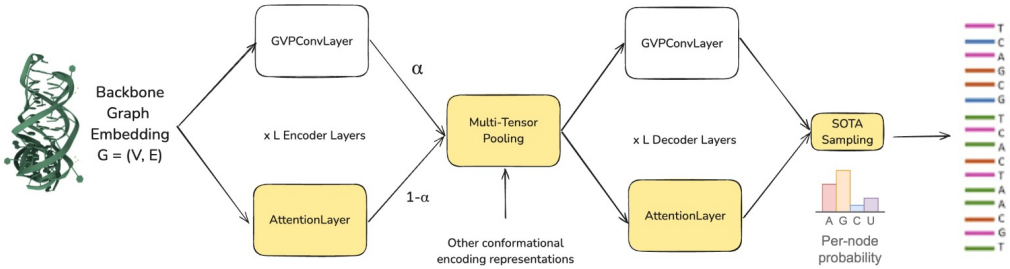


Figure 2: Overview of the gRNAdeX architecture.

that, although we focus on studying the encoder, the results from this section extend naturally to the decoder layers, which can be viewed as encoder layers with $C = 1$.

Denote the overall encoder mapping by $f = f_L \circ \cdots \circ f_1 \circ f_{\text{emb}}$, where the embedding mapping is given by

$$f_{\text{emb}} = \text{GVP}_{\text{emb}} \circ \text{LN} : \{\mathbf{G}_1, \ldots, \mathbf{G}_C\} \mapsto \{\mathbf{H}_1, \ldots, \mathbf{H}_C\},$$

with each $\mathbf{G}_i \in (\mathbb{R}^{64} \times \mathbb{R}^{4\times 3})^N$ corresponding to a conformation, and $\mathbf{H}_i \in (\mathbb{R}^{128} \times \mathbb{R}^{16\times 3})^N$ its high-dimensional representation. Assuming the layer normalization (LN) is 1–Lipschitz, we can upper-bound the Lipschitz constant of the embedding as $L(f_{\text{emb}}) \leq L_{\text{GVP}_{\text{emb}}}$.

Each encoder layer f_l ($l = 1, \ldots, L$) is composed of two submodules: a GVP-based message-passing component (with Lipschitz constant $L_{\text{GVP}_{\text{msg}}}$) and a feedforward update (with Lipschitz constant $L_{\text{GVP}_{\text{ff}}}$), integrated via a residual connection and agreggated by a linear operator $\mathbf{A}$ with spectral norm $\|\mathbf{A}\|$. Thus, the Lipschitz constant for the lth layer is bounded by

$$L(f_l) \leq 1 + \left(\|\mathbf{A}\| \, L^{(l)}_{\text{GVP}_{\text{msg}}} \cdot L^{(l)}_{\text{GVP}_{\text{ff}}}\right).$$

By applying the composition rule for Lipschitz functions, the overall encoder satisfies

$$L(f) \leq L_{\text{GVP}_{\text{emb}}} \cdot \prod_{l=1}^{L} \left(1 + \left(\|\mathbf{A}\| \, L^{(l)}_{\text{GVP}_{\text{msg}}} \cdot L^{(l)}_{\text{GVP}_{\text{ff}}}\right)\right).$$

In this context, we recall that a large Lipschitz constant can make the model overly sensitive to small input perturbations, leading to instability, while a small constant may result in excessive contractiveness, limiting the model's ability to distinguish between different inputs. Our goal is to enhance the model's expressivity while carefully navigating this trade-off. We observe that each additional layer increases the overall Lipschitz constant monotonically, highlighting the growing risk of instability with network depth. Conversely, the only component that could potentially induce contractiveness is the embedding map. For simplicity, we assume this is not the case in our model. This assumption is intuitively reasonable, as the embedding maps inputs to a higher-dimensional space through a learnable matrix, and we rely on the effectiveness of gradient descent to prevent the learned transformation from being excessively contractive.

Motivated by these considerations, this Lipschitz analysis informed us to consider introducing an attention-based layer in parallel with the original GVP-based message passing. Rather than multiplying the existing Lipschitz factors, which could potentially cause a notorious blow-up in sensitivity, a parallel aggregator is simply added by fusing the GVP and attention outputs via $\alpha \times \text{GVP} + (1 - \alpha) \times \text{Attention}$. For simplicity, we set $\alpha = 0.5$, leaving the learning of an optimal α to future work. Because the Lipschitz constant of a sum $f_1 + f_2$ is bounded by $L(f_1) + L(f_2)$ (scaled by the mixing coefficients), this design boosts expressivity—through attention's adaptive neighbor weighting—without recklessly inflating the overall Lipschitz bound. Note the new Lipschitz constant encoder becomes

$$L(f) \leq L_{\text{GVP}_{\text{emb}}} \cdot \prod_{l=1}^{L} \left(1 + \left(\alpha \|\mathbf{A}\| L^{(l)}_{\text{GVP}_{\text{msg}}} \cdot L^{(l)}_{\text{GVP}_{\text{ff}}} + (1 - \alpha) L^{(l)}_{\text{attn}}\right)\right).$$

Unlike a fixed linear operator $\mathbf{A}$, attention learns a data-dependent weighting matrix (constrained to be row-stochastic via a softmax). While the GVP branch enforces strong geometric equivariance and local feature aggregation, the attention branch captures long-range dependencies and global interactions. Our proposed hybrid mechanism can "down-weight" uninformative neighbors and "up-weight" salient ones, capturing more nuanced node relationships and thus increasing the model's capacity to discriminate among different inputs. Their combined output is more expressive and capable of representing a broader class of functions. We achieve this while maintaining a theoretical guarantee that the encoder's Lipschitz constant does not increase significantly.

3.2 Universality via Multi-Tensor Pooling

One anomaly we presumed to be working suboptimally was the way the pooling was working in the original architecture. We note that in this architecture, after the embedding layers, each RNA structure graph is represented as an element of

$$\mathbf{H} = (\mathbb{R}^{128} \times \mathbb{R}^{16\times 3})^N,$$

and the input to the encoding layer is a set

$$\mathbf{X} = \{\mathbf{H}_1, \ldots, \mathbf{H}_C\} \subset \mathbf{H}^C.$$

The encoder layers enriched these represantions and their output is then passed to the pooling layer. The function computed by the pooling must respect the symmetry inherent to this input: it must be invariant under any permutation of the C graphs (i.e., invariant with respect to S_C) and equivariant with respect to the symmetry group $\mathcal{H}$ (e.g. rigid-motion group) that governs the internal geometric structure of each graph.

A critical component in achieving universality for set functions (in our case is set of geometric graphs) is the design of the pooling operator. Standard pooling strategies (such as simple averaging) are invariant under S_C but are not injective; that is, they can collapse different sets into the same representation. To overcome this limitation, we propose a pooling operator based on higher–order statistics.

Let $\psi : \mathbf{H} \to \mathbf{F}$ be the per–graph feature extractor implemented by our AttentiveGVP–GNN layers, where $\mathbf{F} \subset \left(\mathbb{R}^{128} \times \mathbb{R}^{16\times 3}\right)^{N\times C}$. We define the pooling operator ϕ as

$$\phi(\mathbf{X}) = \bigoplus_{k=1}^{K} \frac{1}{C} \sum_{i=1}^{C} \psi(\mathbf{H}_i)^{\odot k},$$

where $\bigoplus$ denotes concatenation along the feature dimension.

Here, $\psi(\mathbf{H}_i)^{\odot k}$ represents the element-wise kth power of the feature vector, and K is chosen sufficiently large so that the mapping

$$\phi : \mathbf{H}^C \to (\mathbb{R}^{K\cdot 128} \times \mathbb{R}^{K\cdot 16\times 3})^N$$

is injective over the compact domain of interest. This formulation retains richer structural information across different moment orders while still enforcing permutation invariance over conformations.

It is worth noting that ϕ remains invariant with respect to S_C because averaging preserves permutation invariance, while the concatenation operation allows for a more expressive feature representation. Additionally, ϕ is equivariant with respect to $\mathcal{H}$, as ψ is built from $\mathcal{H}$–equivariant GVP layers. This pooling strategy strengthens the representational capacity of our model by capturing higher-order geometric interactions among conformations.

We now state the following theorem, which is inspired by and extends results such as those in [22].

Theorem 1 (Universality of the Model). Let $\mathcal{K} \subset \mathbf{H}^C$ be a compact domain that is invariant under the action of $G = S_C \times \mathcal{H}$, where S_C permutes the C graphs and $\mathcal{H}$ acts on the geometric features. Suppose that the per–graph mapping $\psi : \mathbf{H} \to \mathbf{F}$ is implemented by universal $\mathcal{H}$–equivariant layers and that for a sufficiently large integer K, the pooling operator $\phi(\mathbf{X})$ is injective on $\mathcal{K}$. Then, for any continuous G–invariant function

$$f : \mathcal{K} \to (\mathbb{R}^{128} \times \mathbb{R}^{16\times 3})^N,$$

and for every $\epsilon > 0$, there exists a multilayer perceptron M such that

$$\sup_{\mathbf{X} \in \mathcal{K}} \|M(\phi(\mathbf{X})) - f(\mathbf{X})\| < \epsilon.$$

The proof of this theorem is presented in **Appendix** B. This theoretical result motivates our architectural modification: by replacing simple averaging with higher–order tensor moment pooling, we ensure that the aggregated representation retains all necessary information to uniquely characterize the input set. Consequently, this guarantees the universality of the encoder component of our model, and consequently for our whole model .

3.3 State-of-the-Art Sampling Strategies

RNA design consists of the search of given nucleotide sequences that are compatible with certain predefined structures [1, 19]. The inverse RNA folding problem is therefore a mapping from structures to nucleotide-based sequences.

To enhance the quality of the generated sequences, we modified the greedy autoregressive decoding strategy used in gRNAde [1]. Our approach incorporates beam search with adjustable branching and width parameters, and supports multiple sampling techniques—including top-k and priority sampling [23], top-p (nucleus) sampling [24], and min-p sampling [25]. By enabling these diverse sampling strategies, our method addresses the limitations of early commitment in standard single-path decoding, promoting a more effective exploration of the RNA sequence space.

Let $x = (x_1, x_2, \ldots, x_T)$ denote a candidate RNA sequence of length T. In the previous strategy, the log-probability of a complete sequence is given by the autoregressive decomposition

$$\log P(x) = \sum_{t=1}^{T} \log p(x_t \mid x_1, \ldots, x_{t-1}).$$

In that framework, at each decoding step a single token is sampled, which is equivalent to a *greedy* [26] or stochastic selection from the categorical distribution $p(x_t \mid x_1, \ldots, x_{t-1})$. Although computationally efficient, this approach is prone to error propagation: an erroneous token sampled early in the sequence may irreversibly bias subsequent predictions.

To address this limitation, our new strategy introduces a beam search mechanism parameterized by a *beam width* B and a *branching factor* b. At each token position t, we maintain a beam B_t consisting of the B most promising partial sequences $x_{1:t}$ based on their cumulative log-probabilities. For each sequence in B_t, the model proposes b candidate continuations using a chosen sampling strategy σ (e.g., top-k, top-p, or min-p sampling), yielding an expanded set of $B \times b$ candidates.

Formally, if $B_t = x_{1:t}^{(1)}, \ldots, x_{1:t}^{(B)}$ with corresponding log-probabilities $\ell(x_{1:t}^{(i)})$, then for each i we obtain candidate tokens $c_1^{(i)}, \ldots, c_b^{(i)}$ along with their log-probabilities $\delta_1^{(i)}, \ldots, \delta_b^{(i)}$. The log-probability of each extended sequence is updated as

$$\ell(x_{1:t}^{(i)} \circ c_j^{(i)}) = \ell(x_{1:t}^{(i)}) + \delta_j^{(i)},$$

where $\circ$ denotes sequence concatenation. The B candidates with the highest updated scores are then selected to form the new beam B_{t+1}.

This branching and pruning mechanism can be seen as an approximate maximization of the sequence likelihood:

$$x^* = \arg\max_x \log P(x).$$

By tracking multiple candidates, our strategy reduces the risk of local optima that may occur in greedy, single-path sampling, and thus more effectively approximates the true maximum a posteriori (MAP) sequence. Nonetheless, the runtime complexity of the decoding process is $\mathcal{O}(T \cdot B \cdot C)$, where T is the sequence length, B is the beam width, and C is the cost of generating and sampling b candidates per beam element. Thus, improvements in diversity and sequence quality via higher B or k come at a cost of greater computational overhead.

Another well-known limitation of beam search—often referred to as the *beam search curse* [27]—arises when increasing the beam width, which leads to a degradation in output quality. This counterintuitive effect is closely related to the curse of dimensionality: as the search space expands

combinatorially with larger beams and deeper token horizons, the model becomes more likely to overcommit to sequences with superficially high likelihood but poor semantic or structural integrity.

To mitigate these effects, we incorporate a flexible temperature parameter τ_t that increases as a function of sequence length t. This dynamic temperature flattens the probability distribution at each decoding step, thereby counteracting the model's tendency to over-concentrate on top-ranked tokens as the sequence length is increased. By scaling the logits before sampling, τ_t ensures that as beam sequences grow longer, the effective entropy of the distribution remains sufficiently high to encourage diversity in candidate generation, similar to [28].

The enhanced decoding strategy of gRNAdeX combines the strengths of deterministic algorithms such as beam search — which seek to maximize the cumulative logarithmic probability of generated sequences — with adaptive stochastic sampling (e.g., top-p, min-p), further refined by a calibrated stochasticity parameter that dynamically regulates diversity during generation.

3.4 RNA-Informed Sequence Generation

In the context of proteins and DNA, we often come across the term *nullomers*. These represent short DNA or amino acid sequences that are absent from the genome or proteome, respectively [29]. A broader term, *Minimally Absent Words* (MAWs), describes both nullomers and longer absent sequences that become present in a sequence after the removal of either their leftmost or rightmost letter [30]. MAWs have been computed in the context of bioinformatics for sequence comparison in organisms of all domains of life [31]. However, while methods for identifying absent sequences are well-established for DNA and proteins [30], they have been largely overlooked in RNA sequence generation—despite the abundance of RNA sequence data, especially when compared to the availability of 3D backbone structures [5].

Definition 1. *Let Σ be a finite alphabet and $S \in \Sigma^*$ be a finite string. A string $w \in \Sigma^+$ is said to be* absent *[32] from S if w does not appear as a contiguous substring of S. We call w a* minimal absent word *of S if:*

1. *w is absent from S, and*
2. *every proper substring of w is present as a substring in S.*

In this work, we introduce gRNAdeX, a modular and transparent pipeline for identifying RNA-specific MAWs which researchers can easily adapt to different datasets and biological contexts. Unlike approaches that rely on pretrained language models—often trained on general-purpose datasets with unknown biases—our method leverages the rich availability of RNA sequence data [19] directly, without introducing opaque, internally learned constraints during its training process, such as in [19].

The core of the gRNAdeX pipeline combines a linear-time algorithm for MAWs extraction [33] with Markov modeling and multiple-testing correction techniques [30], enabling the identification of substrings that are truly absent from RNA types in a user-defined design task. This pipeline empowers users to tailor absence constraints to their specific biological questions, a crucial feature in RNA design research.

By assigning near-zero probability (or a logit of $-\infty$) to tokens that would complete a MAW, we prevent the generation of biologically implausible subsequences—while only restricting the final token. This is key, as the rest of the substring does appear in the reference dataset, highlighting the precision and benefit of using MAWs to guide sampling in autoregressive decoding.

This targeted filtering narrows the sampling space to empirically valid sequences, enhancing biological relevance in RNA design. As a result, *gRNAdeX* provides a transparent, modular, and trustworthy framework for researchers. By focusing on biologically plausible sequences, it reduces the complexity of the inverse folding search space and improves both the quality and interpretability of the generated RNA.

4 Results

Due to constraints in GPU resources, we did not pursue deeper or hierarchical architectures or substantially increase latent dimensions. Instead, our focus was on implementing the targeted

modifications proposed in **Section** 3: multi-head attention in the encoder/decoder, higher-order tensor moment pooling, and a robust sampling mechanism. All experiments reported here were conducted on a reduced dataset restricted to RNA backbones with fewer than 500 nucleotides, allowing us to obtain preliminary yet indicative results without prohibitive computational overhead.

4.1 Architecture

Table 2 summarize our ablation study, comparing the baseline model with various architectural modifications. The specific configurations are detailed in **Table** 1. In our designs, the baseline uses a standard GVP encoder with mean pooling and categorical sampling, while the modified variants incorporate attention mechanisms and Multi-Tensor pooling; here, *Attention** denotes attention on both scalar and vector features, whereas *Attention* is applied only to scalars.

Table 1: Architectural components used in each model variant

Model Variant	Encoder	Pooling	Decoder	Sampling
Baseline	GVP	Mean	GVP	Categorical
Decoder-Light Attention	GVP + Attention*	Mean	GVP	Categorical
Hybrid Light	GVP + Attention*	Mean	GVP + Attention*	Categorical
Hybrid + Attention Pooling	GVP + Attention*	Attention	GVP + Attention*	Categorical
Hybrid + Multi-Tensor	GVP + Attention*	Multi-Tensor	GVP + Attention*	Categorical
Edge Ablation	GVP	Multi-Tensor	GVP	Categorical
Decoder Ablation	GVP + Attention	Multi-Tensor	GVP	Categorical
Full Hybrid	GVP + Attention	Multi-Tensor	GVP + Attention	Categorical
Full Hybrid + Beam	GVP + Attention	Multi-Tensor	GVP + Attention	Beam

Our results seem to indicate that employing Multi-Tensor pooling on node features generally improves performance, as demonstrated by the Hybrid + Multi-Tensor variant, which achieved the highest scScore. In contrast, excessive attention—especially when applied to node vectors (Hybrid + Attention Pooling)—degrades performance. One plausible explanation for this is that node vectors inherently capture critical geometric and spatial information in a structured manner. Applying an attention mechanism directly to these vectors may disrupt their intrinsic spatial relationships, leading to a loss of essential structural cues. Furthermore, the Full Hybrid + Beam variant—integrating comprehensive attention with beam sampling—attains the best recovery performance, offering a well-balanced trade-off between recovery and structural accuracy. Based on these findings, we select the Full Hybrid + Beam model for future experiments, as it leverages the strengths of our architectural modifications while mitigating the negative effects associated with excessive attention.

Table 2: Performance metrics including perplexity and scScore (RMSD)

Model Variant	BEST test recovery	Perplexity	scScore	scScore (RMSD)
Baseline	0.4301	1.7471	0.5144	14.4954
Decoder-Light Attention	0.4255	1.6879	0.5661	14.1097
Hybrid Light	0.4292	1.7393	0.6786	13.8746
Hybrid + Attention Pooling	0.4079	2.6126	0.0305	22.7999
Hybrid + Multi-Tensor	0.4288	1.7656	**0.6889**	13.0251
Edge Ablation	0.4460	1.6988	0.6775	12.2989
Decoder Ablation	0.4518	1.6551	0.5568	**11.7571**
Full Hybrid	0.4575	1.7223	0.5245	13.6263
Full Hybrid + Beam	**0.4609**	**1.6837**	0.5254	13.0505

It is noteworthy that the original gRNAde model achieves a sequence recovery rate of approximately 56%. In contrast, our best model currently attains around 46% on a reduced dataset—limited to backbones with no more than 500 nodes due to restricted GPU resources. Consequently, the observed improvement of roughly 3% over the baseline might underestimate the potential gains on the full dataset.

4.2 Sampling Strategies

Due to a cluster access failure prior to submission, complete analytics were unavailable. The results and figures presented herein are based on the most recent data downloads and offer only a preliminary view of performance trends. Hyperparameters (see **Table** 3) were tuned to balance computational efficiency and sequence recovery accuracy, selecting the configuration that minimizes runtime while maximizing RNA sequence reconstruction from 3D backbone inputs.

Based on empirical evaluations summarized in **Figures** 3 to 5, and although non-clearly distinguishable from the lack of availability of latest hyperparameter runs, we hypothesize, using literature and incomplete data downloads, that min-p sampling is the method deemed most effective for our use case scenario, tuned with a threshold of 0.05. This low threshold ensures that the sampling process prioritizes high-probability continuations by pruning unlikely paths while still maintaining diversity among plausible candidates. Beam parameters were selected to optimize computational efficiency and model recovery, as evidenced by **Figure** 7. The chosen configuration reflects the best trade-off between computational cost and accurate sequence recovery.

Figure 14 further illustrates the limitations associated with increasing beam branching—a phenomenon known as the beam search curse [27]. While a higher beam branching factor improves self-consistency and lowers perplexity by exploring a larger set of candidate sequences and refining outputs to adhere to learned patterns, excessive branching can detrimentally impact native sequence recovery. Overextensive exploration tends to favor fluency and internal consistency over strict adherence to native RNA structures, leading to outputs that deviate from actual biological sequences. This trade-off underscores the necessity of aligning the sampling strategy with the specific recovery objective rather than relying solely on general-purpose language modeling metrics.

Finally, as shown in **Figure** 18, the results obtained after hyperparameter tuning—despite being based on limited hyperparameter-tuning runs—demonstrate that our proposed gRNAdeX model outperforms the original gRNAde across all evaluated metrics. These findings validate our architectural modifications and sampling strategies, providing a solid foundation for future experiments.

Table 3: Hyperparameter search space for generation experiments.

Temperature	Max Temp	Temp Factor	Beam Width	Beam Branch	greedy	top-p	top-k	min-p
`0.0,0.1,0.5`	`0.5`	`1e-6,1e-5`	`1,2,4,6,8`	`1,2,4,6,8`	`-`	`0.8,0.9`	`2,3`	`0.05,0.1`

4.3 RNA-Guided Generation

Using the open-source repository MAWs with a few data pre-processing and post-processing stages to ensure compatibility with RNA formatting (original code implementation is designed for DNA), we created a dataset of RNA-specific MAWs in FASTA file formats from the RNAsolo database [35] of sequence lengths between 11 and 16 nucleotides [29, 36, 37]. The dataset is made available for use at unique_rfam_maw. For reference, this dataset can be either directly used or replaced by compatible file formatting to guide the specific design task in hand.

4.3.1 Use Case

To showcase the modular RNA-typed gRNA generation capabilities of gRNAdeX, we conducted a targeted experiment using the organism *Streptobacillus moniliformis DSM 12112*. A detailed description of the dataset associated with this organism can be found in Appendix D.

We utilized the custom-built dataset, `unique_rfam_maw`, and performed statistical analyses using an RNA-specific Markovian testing pipeline akin to the Nullomers Assessor. Through this pipeline, we identified statistically significant minimally absent words (MAWs) tailored to a specific 5s rRNA design scenario for *S. moniliformis*.

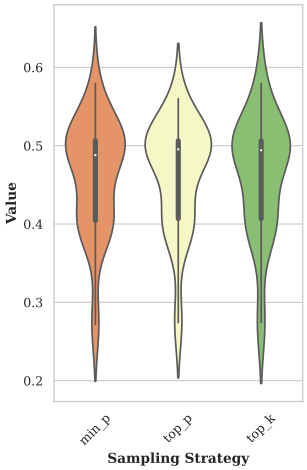


Figure 3: Native Sequence Recovery for values in **Table** 3.

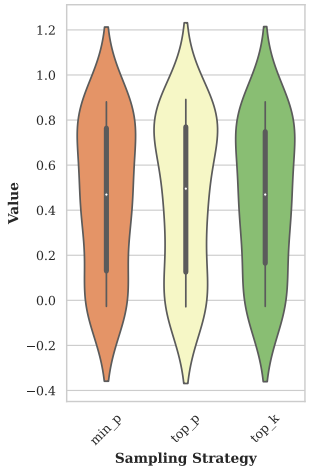


Figure 4: Self-Consistency Scores for values in **Table** 3.

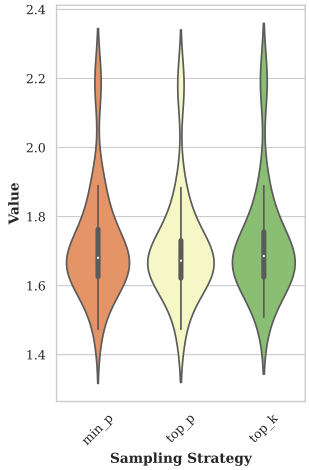


Figure 5: Perplexity values for values in **Table** 3.

Figure 6: Evaluating sampling strategies for different model metrics across [34] benchmark.

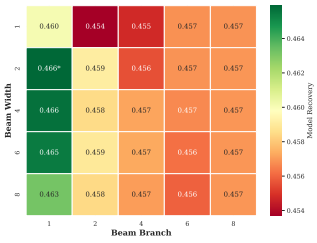


Figure 7: Native Sequence Recovery.

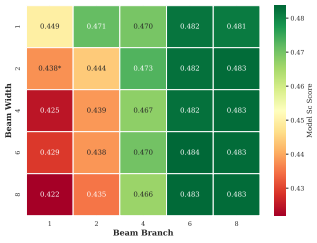


Figure 8: Self-Consistency Scores.

Figure 9: Model Perplexity values.

Figure 10: Effect of beam branch and beam width on different model metrics for min_p 0.05 on [34] benchmark.

The resulting MAW enrichment analysis for this use case is presented in Table 4, with each MAW serving as a prior or forbidden motif during the 5s rRNA generation task. Provided these restrictions, we can feed the 3D backbone coordinates of a single .pdb file (for single-state) or folder (for multi-state design) into our gRNAdeX pipeline. Due to the aforementioned data access issue on the cluster, we are currently unable to provide a comprehensive qualitative analysis of these results.

5 Conclusion & Future Work

By leveraging a robust mathematical architectural framework that enhances expressivity and integrating a novel pipeline for incorporating biologically motivated constraints, gRNAdeX has surpassed all performance metrics of the original gRNAde model. Our experiments demonstrate that the combination of advanced sampling strategies, improved pooling mechanisms, and targeted attention modules yields significant gains in sequence recovery and structural alignment.

For future work, it will be valuable to incorporate additional RNA backbone information alongside protein spatial context. In particular, exploring a one-hot encoding scheme to represent adjacent or interacting amino acid sequences could capture critical structural and functional cues currently overlooked by models trained solely on the RNAsolo dataset. Moreover, integrating physical constraints into the loss function and refining fusion weights within the hybrid encoder may further enhance model performance, ultimately advancing the state-of-the-art in RNA design and expanding its applicability to RNA–protein interaction studies.

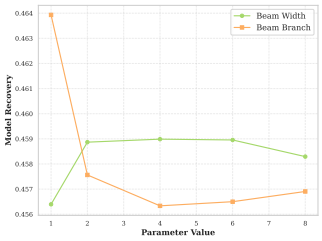


Figure 11: Native Sequence Recovery.

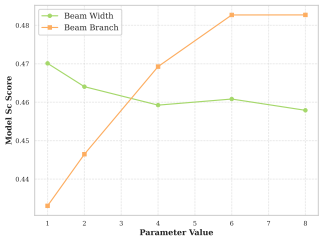


Figure 12: Self-Consistency Scores.

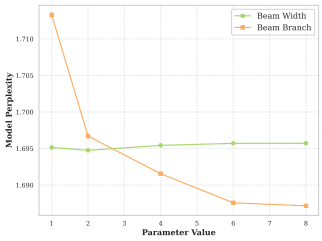


Figure 13: Model Perplexity values.

Figure 14: Marginal effect of beam branch and beam width on different model metrics for min_p 0.05 on [34] benchmark.

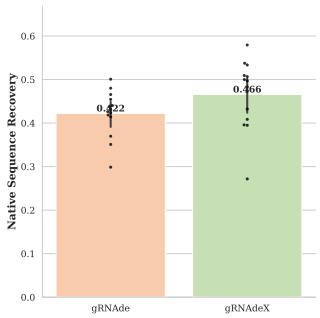


Figure 15: Native Sequence Recovery.

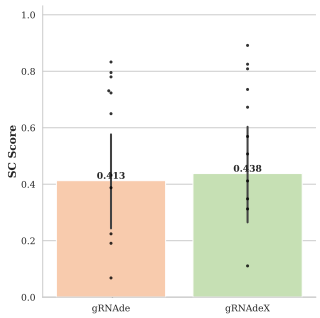


Figure 16: Self-Consistency Scores.

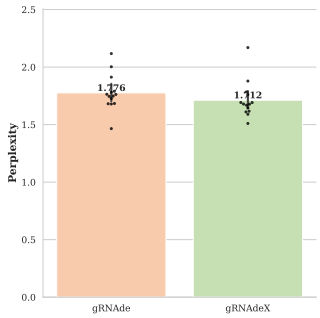


Figure 17: Model Perplexity values.

Figure 18: Comparison of best gRNAdeX architecture versus gRNAde for different model metrics on [34] benchmark.

Table 4: Significantly absent MAWs in *Streptobacillus moniliformis DSM 12112* during 5S rRNA design task using Bonferroni correction.

MAW	Organism	Category	Design Task	Correction Method	Adjusted p-value
UUUUUUUUUUU	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	4.09×10^{-7}
UUAAUAUAUUU	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	6.10×10^{-5}
UUAAUAUAUAU	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	4.43×10^{-3}
UUAAUAAUAUA	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	3.19×10^{-2}
UUAAAAUAAUA	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	6.26×10^{-4}
UAUUAUAAUAU	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	9.07×10^{-3}
UAUAUAAAUAU	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	5.63×10^{-3}
UAAUAUAUUUU	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	2.82×10^{-5}
UAAAAUAAUUA	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	1.18×10^{-2}
UAAAAAAAAAU	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	3.94×10^{-6}
AUUAAUAUAUU	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	1.15×10^{-2}
AUAUAUAUAUU	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	2.29×10^{-2}
AUAAUAUAUUA	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	1.02×10^{-2}
AUAAUAAUAUU	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	1.05×10^{-2}
AUAAAAUAAUU	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	1.03×10^{-3}
AUAAAAAAAAA	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	5.02×10^{-6}
AAUUUAAUUUA	Streptobacillus moniliformis DSM 12112	Bacteria	5S rRNA	Bonferroni	9.35×10^{-3}

Acknowledgements

We thank Haitz Sáez de Ocáriz Borde and Chaitanya K. Joshi for helpful discussions.

References

[1] Chaitanya K Joshi, Arian R Jamasb, Ramon Viñas, Charles Harris, Simon V Mathis, Alex Morehead, and Pietro Liò. gRNAde: Geometric Deep Learning for 3Dd RNA inverse design. *bioRχiv*, 2024.

[2] David Wang and Aisha Farhana. *StatPearls*, chapter Biochemistry, RNA structure. StatPearls Publishing, 2023.

[3] Bruce Alberts, Rebecca Heald, Alexander Johnson, David Morgan, Martin Raff, Keith Roberts, and Peter Walter. *Molecular biology of the cell: seventh international student edition*. WW Norton & Company, 2022.

[4] Ivo L. Hofacker, Walter Fontana, Peter F. Stadler, L. S. Bonhoeffer, Manfred Tacker, and Philipp Schuster. Fast folding and comparison of RNA secondary structures. *Monatshefte für Chemie / Chemical Monthly*, 125:167–188, 1994.

[5] Han Huang, Ziqian Lin, Dongchen He, Liang Hong, and Yu Li. RiboDiffusion: Tertiary Structure-based RNA Inverse Folding with Generative Diffusion Models. *Bioinformatics*, 40 (Supplement 1):i347–i356, 2024.

[6] Joseph D. Yesselman, Daniel Eiler, Erik D. Carlson, Michael R. Gotrik, Anne E. D'Aquino, Alexandra N. Ooms, Wipapat Kladwang, Paul D. Carlson, Xuesong Shi, David A. Costantino, Daniel Herschlag, Julius B. Lucks, Michael C. Jewett, Jeffrey S. Kieft, and Rhiju Das. Computational design of three-dimensional RNA structure and function. *Nature Nanotechnology*, 14 (9):866–873, 2019.

[7] Hua-Ting Yao, Cedric Chauve, Mireille Regnier, and Yann Ponty. Exponentially Few RNA Structures are Designable. In *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, pages 289–298, 2019.

[8] Yu Zhou, Yann Ponty, Stéphane Vialette, Jérôme Waldispuhl, Yi Zhang, and Alain Denise. Flexible RNA design under structure and sequence constraints using formal languages. In *Proceedings of the International Conference on Bioinformatics, Computational Biology and Biomedical Informatics*, pages 229–238, 2013.

[9] Tianshuo Zhou, Wei Yu Tang, Apoorv Malik, David H Mathews, and Liang Huang. Scalable and Interpretable Identification of Minimal Undesignable RNA Structure Motifs with Rotational Invariance. *ArXiv*, 2024.

[10] Sunandan Mukherjee, S. Naeim Moafinejad, Nagendar Goud Badepally, Katarzyna Merdas, and Janusz M. Bujnicki. Advances in the field of RNA 3D structure prediction and modeling, with purely theoretical approaches, and with the use of experimental data. *Structure*, 32(11): 1860–1876, 2024.

[11] Almudena Ponce-Salvatierra, Astha, Katarzyna Merdas, Chandran Nithin, Pritha Ghosh, Sunandan Mukherjee, and Janusz M Bujnicki. Computational modeling of RNA 3D structure based on experimental data. *Bioscience reports*, 39(2), 2019.

[12] Ronny Lorenz, Stephan H Bernhart, Christian Höner zu Siederdissen, Hakim Tafer, Christoph Flamm, Peter F Stadler, and Ivo L Hofacker. ViennaRNA Package 2.0. *Algorithms for molecular biology*, 6:1–14, 2011.

[13] Alexander Churkin, Matan Drory Retwitzer, Vladimir Reinharz, Yann Ponty, Jérôme Waldispühl, and Danny Barash. Design of RNAs: comparing programs for inverse RNA folding. *Briefings in Bioinformatics*, 19(2):350–358, 2018.

[14] Quentin Vicens and Jeffrey S Kieft. Thoughts on how to think (and talk) about RNA structure. *Proceedings of the National Academy of Sciences*, 119(17), 2022.

[15] Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The Protein Data Bank. *Nucleic acids research*, 28(1):235–242, 2000.

[16] Weitao Du, He Zhang, Yuanqi Du, Qi Meng, Wei Chen, Bin Shao, and Tie-Yan Liu. SE(3) Equivariant Graph Neural Networks with Complete Local Frames. *ArXiv*, 2022.

[17] Julia Koehler Leman, Brian D Weitzner, Steven M Lewis, Jared Adolf-Bryfogle, Nawsad Alam, Rebecca F Alford, Melanie Aprahamian, David Baker, Kyle A Barlow, Patrick Barth, et al. Macromolecular modeling and design in Rosetta: recent methods and frameworks. *Nature methods*, 17(7):665–680, 2020.

[18] Divya Nori and Wengong Jin. RNAFlow: RNA Structure & Sequence Design via Inverse Folding-Based Flow Matching. *ArXiv*, 2024.

[19] Haz Sameen Shahgir, Md Rownok Zahan Ratul, Md Toki Tahmid, Khondker Salman Sayeed, and Atif Rahman. RNA-DCGen: Dual Constrained RNA Sequence Generation with LLM-Attack. *bioRXiv*, 2024.

[20] Md Toki Tahmid, Haz Sameen Shahgir, Sazan Mahbub, Yue Dong, and Md Shamsuzzoha Bayzid. BiRNA-BERT allows efficient RNA language modeling with adaptive tokenization. *bioRxiv*, 2024.

[21] Rafael Josip Penić, Tin Vlašić, Roland G Huber, Yue Wan, and Mile Šikić. RiNALMo: General-Purpose RNA Language Models Can Generalize Well on Structure Prediction Tasks. *ArXiv*, 2024.

[22] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Ruslan Salakhutdinov, and Alexander J. Smola. Deep Sets. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 3391–3401, 2017.

[23] Dejan Grubisic, Volker Seeker, Gabriel Synnaeve, Hugh Leather, John Mellor-Crummey, and Chris Cummins. Priority Sampling of Large Language Models for Compilers. In *Proceedings of the 4th Workshop on Machine Learning and Systems*, pages 91–97, 2024.

[24] Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The Curious Case of Neural Text Degeneration. *ArXiv*, 2020.

[25] Minh Nguyen, Andrew Baker, Clement Neo, Allen Roush, Andreas Kirsch, and Ravid Shwartz-Ziv. Turning Up the Heat: Min-p Sampling for Creative and Coherent LLM Outputs. *ArXiv*, 2025.

[26] Yifan Song, Guoyin Wang, Sujian Li, and Bill Yuchen Lin. The Good, The Bad, and The Greedy: Evaluation of LLMs Should Not Ignore Non-Determinism. *ArXiv*, 2024.

[27] Yilin Yang, Liang Huang, and Mingbo Ma. Breaking the Beam Search Curse: A Study of (Re-)Scoring Methods and Stopping Criteria for Neural Machine Translation. *ArXiv*, 2018.

[28] Dongkyu Lee, Gyeonghun Kim, Janghoon Han, Taesuk Hong, Yi-Reun Kim, Stanley Jungkyu Choi, and Nevin L Zhang. Local Temperature Beam Search: Avoid Neural Text DeGeneration via Enhanced Calibration. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9903–9915, 2023.

[29] Ilias Georgakopoulos-Soares, Ofer Yizhar-Barnea, Ioannis Mouratidis, Martin Hemberg, and Nadav Ahituv. Absent from DNA and protein: genomic characterization of nullomers and nullpeptides across functional categories and evolution. *Genome Biology*, 22:1–24, 2021.

[30] Grigorios Koulouras and Martin C Frith. Significant non-existence of sequences in genomes and proteomes. *Nucleic Acids Research*, 49(6):3139–3155, 2021.

[31] Carl Barton, Alice Heliou, Laurent Mouchard, and Solon P Pissis. Linear-time Computation of Minimal Absent Words Using Suffix Array. *BMC Bioinformatics*, 15:1–10, 2014.

[32] Tooru Akagi, Yuki Kuhara, Takuya Mieno, Yuto Nakashima, Shunsuke Inenaga, Hideo Bannai, and Masayuki Takeda. Combinatorics of minimal absent words for a sliding window. *Theoretical Computer Science*, 927:109–119, 2022.

[33] Alice Héliou, Solon P Pissis, and Simon J Puglisi. emmaw: computing minimal absent words in external memory. *Bioinformatics*, 33(17):2746–2749, 04 2017. ISSN 1367-4803. doi: 10.1093/bioinformatics/btx209. URL https://doi.org/10.1093/bioinformatics/btx209.

[34] Rhiju Das, John Karanicolas, and David Baker. Atomic accuracy in predicting and designing noncanonical rna structure. *Nature methods*, 7(4):291–294, 2010.

[35] Bartosz Adamczyk, Maciej Antczak, and Marta Szachniuk. Rnasolo: a repository of cleaned pdb-derived rna 3d structures. *Bioinformatics*, 38(14):3668–3670, 2022.

[36] Candace S.Y. Chan, Ioannis Mouratidis, Austin Montgomery, Georgios Christos Tsiatsianis, Nikol Chantzi, Martin Hemberg, Nadav Ahituv, and Ilias Georgakopoulos-Soares. The topography of nullomer-emerging mutations and their relevance to human disease. *Computational and Structural Biotechnology Journal*, 30:1–11, 2025.

[37] Austin Montgomery, Georgios Christos Tsiatsianis, Ioannis Mouratidis, Candace SY Chan, Maria Athanasiou, Anastasios D Papanastasiou, Verena Kantere, Nikos Syrigos, Ioannis Vathiotis, Konstantinos Syrigos, et al. Utilizing nullomers in cell-free RNA for early cancer detection. *Cancer Gene Therapy*, 31(6):861–870, 2024.

[38] George Cybenko. Approximation by Superpositions of a Sigmoidal Function. *Mathematics of Control, Signals and Systems*, 2(4):303–314, 1989.

[39] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer Feedforward Networks are Universal Approximators. *Neural Networks*, 2(5):359–366, 1989.

A Tables

Table 5: Summary of GNN transformations across the architecture pipeline.

Stage/Layer	Mapping
Embedding	LayerNorm + GVP: $\left(\mathbb{R}^{64} \times \mathbb{R}^{4\times 3}\right)^{N\times C} \mapsto \left(\mathbb{R}^{128} \times \mathbb{R}^{16\times 3}\right)^{N\times C}$ $\left(\mathbb{R}^{32} \times \mathbb{R}^{1\times 3}\right)^{E\times C} \mapsto \left(\mathbb{R}^{32} \times \mathbb{R}^{1\times 3}\right)^{E\times C}$
Encoder	MultiGVPConvLayer $\times$ 3 (edges remain fixed): $\left(\mathbb{R}^{128} \times \mathbb{R}^{16\times 3}\right)^{N\times C} \mapsto \left(\mathbb{R}^{128} \times \mathbb{R}^{16\times 3}\right)^{N\times C}$
Pooling	Averaging (across conformational representations): $\left(\mathbb{R}^{128} \times \mathbb{R}^{16\times 3}\right)^{N\times C} \mapsto \left(\mathbb{R}^{128} \times \mathbb{R}^{16\times 3}\right)^{N}$ $\left(\mathbb{R}^{32} \times \mathbb{R}^{1\times 3}\right)^{E\times C} \mapsto \left(\mathbb{R}^{32} \times \mathbb{R}^{1\times 3}\right)^{E}$
Decoder	GVPConvLayer $\times$ 3 (autoregressive): $\left(\mathbb{R}^{128} \times \mathbb{R}^{16\times 3}\right)^{N} \mapsto \left(\mathbb{R}^{128} \times \mathbb{R}^{16\times 3}\right)^{N}$ From pooled edge + sequence embeddings $\mapsto (s_e^{\text{dec}}, \mathbf{v}_e^{\text{dec}})$
Output Layer	GVP: $\left(\mathbb{R}^{128} \times \mathbb{R}^{16\times 3}\right)^{N} \mapsto (\mathbb{R}^4)^N$
Sampling	Categorical Sampling: $(\mathbb{R}^4)^N \mapsto \{A, C, G, U\}^N$

B Proofs

Proof 1. Consider an arbitrary function

$$f : \mathcal{K} \to (\mathbb{R}^{128} \times \mathbb{R}^{16\times 3})^N,$$

with $\mathcal{K} \subset \mathbf{H}^C = \left((\mathbb{R}^{128} \times \mathbb{R}^{16\times 3})^N\right)^C$ and the group action $G = S_C \times \mathcal{H}$ defined as in the Theorem 1 statement.

Note in the encoding of our model, each graph $\mathbf{H}_i$ is mapped to a feature representation $\psi(\mathbf{H}_i) \in \mathbf{F}$ by a stack of $\mathcal{H}$–equivariant GVP–GNN layers. We assume ψ can approximate any continuous $\mathcal{H}$–equivariant function on $\mathbf{H}$. The mapping ϕ aggregates higher–order moments of the set $\{\psi(\mathbf{H}_1), \ldots, \psi(\mathbf{H}_C)\}$. Classical results in symmetric polynomial theory (e.g., Newton's identities) imply that, if K is chosen sufficiently large relative to the dimension of $\psi(\mathbf{H}_i)$, then the collection

$$\left\{ \sum_{i=1}^{C} \psi(\mathbf{H}_i)^{\odot k} \;:\; k = 1, \ldots, K \right\}$$

uniquely determines the multiset $\{\psi(\mathbf{H}_i)\}$. Thus, ϕ is injective. Since f is G–invariant and ϕ is both invariant and injective, there exists a continuous function ρ defined on $\phi(\mathcal{K})$ such that

$$f(\mathbf{X}) = \rho(\phi(\mathbf{X})).$$

By the classical universal approximation theorem (e.g., [38, 39]), there exists an MLP M that can approximate ρ uniformly over the compact set $\phi(\mathcal{K})$ to within any desired error ϵ. Thus, the composite mapping

$$F(\mathbf{X}) = M(\phi(\mathbf{X}))$$

approximates f uniformly on $\mathcal{K}$. This proves that the encoder part of our architecture is a universal approximator of continuous G–invariant functions, provided that the pooling operator uses higher–order statistics. □

C Minimally Absent Words and Nullomers Assessor

The database created with minimally absent words is found in unique_rfam_maw, where we provide the user with a list of forbidden motifs per RNA type from the RNASolo dataset. This database should be used with care, taking into account that these sequences should be tested against desired tasks using the modified pipeline of Nullomers Assessor.

D Use-Case Organism Details

Table 6: Metadata summary for *Streptobacillus moniliformis DSM 12112* genome assemblies.

Field	Value
Organism Scientific Name	*Streptobacillus moniliformis DSM 12112*
Organism Common Name	—
Organism Qualifier	strain: DSM 12112
Taxonomy ID	519441
Assembly Name	ASM2456v1
Assembly Accession (GenBank)	GCA_000024565.1
Source (GenBank)	GenBank
Annotation (GenBank)	Annotation submitted by US DOE Joint Genome Institute (JGI-PGF)
Assembly Accession (RefSeq)	GCF_000024565.1
Source (RefSeq)	RefSeq
Annotation (RefSeq)	GCF_000024565.1-RS_2024_12_09
Level	Complete Genome
Contig N50	1,662,578
Size (bp)	1,673,280
Submission Date	2009-11-16
Gene Count (GenBank)	1,568
Gene Count (RefSeq)	1,570
BioProject	PRJNA29309
BioSample	SAMN00002603