Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Structure-Guided RNA Design via Multinomial Diffusion

Georg Back

georg.back@hswt.de

Weihenstephan-Triesdorf University of Applied Sciences

Dirk Walther

Max Planck Institute for Molecular Plant Physiology

Florian Haselbeck

Weihenstephan-Triesdorf University of Applied Sciences

Research Article

Keywords:

Posted Date: November 9th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11401521/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Structure-Guided RNA Design via Multinomial Diffusion

Georg Back[1*], Dirk Walther[2] and Florian Haselbeck[1]

[1]Department of Sustainable Agriculture and Energy Systems, Weihenstephan-Triesdorf University of Applied Sciences, Am Staudengarten 1, Freising, 85354, Germany.
[2]Department of Bioinformatics, Max Planck Institute for Molecular Plant Physiology, Am Mühlenberg 1, Potsdam, 14476, Germany.

*Corresponding author(s). E-mail(s): georg.back@hswt.de;

Abstract

The structure of RNA plays an essential role in its biological function, with the secondary structure being a key determinant. While significant advances have been made in RNA folding models, the inverse problem, i.e., generating sequences that fold into a target structure has received less attention. Existing approaches commonly rely on iterative optimization, often guided by a folding oracle, which can limit scalability and risk converging to local optima. Diffusion models have shown a superior performance in many application areas, but their noising process is not directly applicable to discrete nucleotide sequences. In this paper, we adapt the multinomial diffusion framework, which introduces discrete noise via categorical transitions, to the RNA sequence space. Furthermore, we incorporate secondary structure information to guide the generative process toward sequences that fold into a desired target structure. For this purpose, we evaluate two architectural variants: (i) modeling the problem either by projecting the sequence embedding into a 2D pairwise representation that reflects the structure of a contact map, or (ii) by restructuring the architecture as a purely 1D sequence model. Our results show that both approaches generate sequences that fold with high accuracy while maintaining substantial variability among generated sequence samples. We further evaluated constrained generation via sequence inpainting, resulting in performance comparable to unconstrained generation. Even though the performance on the human-designed Eterna100 benchmark was lower than that of the state-of-the-art methods LEARNA and SAMFEO, our results indicate that diffusion-based models provide a promising and scalable framework for conditional nucleotide sequence generation.

Keywords: RNA secondary structure, RNA, Deep learning, Structural bioinformatics, Discrete diffusion models

1 Background

Functional RNA molecules have long been of high interest in biomedical and biotechnological research, for example, in catalysis via ribozymes [1], in RNA interference–based gene-silencing therapies [2], and in vaccine research, including the development of COVID-19 vaccines [3]. Since RNA function is closely linked to its structure, designing RNAs that fold into a specific structure is a central topic in computational biology. Existing approaches typically rely on heuristic search, reinforcement learning (RL), or iterative optimization with a strong reliance on a respective folding algorithm [4–6]. Despite their success, these methods are often computationally expensive and may struggle to capture the one-to-many nature of the design problem. Generative modeling offers an alternative promising approach by directly learning the distribution of valid sequences. In this work, we investigate whether discrete diffusion models can be adapted to the design of secondary-structure–conditioned RNA sequences.

Similarly to proteins, the RNA structure is organized hierarchically, with the secondary structure defined by base-pairing patterns representing the dominant constraint in higher-order folding [7]. In contrast to proteins, where tertiary interactions contribute stronger to the final fold, RNA molecules are more dependent on their secondary structure as a folding scaffold: helices and loop motifs impose strong constraints that restrict the set of possible tertiary conformations. Structural elements such as stems, loops, and junctions facilitate interactions with proteins, other RNAs, and small molecules, and play a central role in processes including translation, regulation, and catalysis [8]. While prediction of RNA tertiary structure has gained increasing attention in recent years, the scarcity of high-quality 3D data and the strong thermodynamic constraints secondary structure imposes on folding have kept secondary structure prediction an active topic in computational biology. Classical methods for predicting RNA structure often rely on thermodynamic models, describing the free energy of an RNA molecule as the sum of contributions of substructures and searching for the structure with the minimum free energy through dynamic programming [9–11]. These methods are interpretable and generalize well, but often struggle in capturing complex structural patterns. Machine learning (ML)-based approaches have achieved higher predictive performance by learning directly from annotated datasets, with underlying model architectures ranging from classical techniques such as support vector machines and boosting to modern deep learning (DL)-based architectures [12–14]. However, they often generalize poorly to unseen RNA families, often due to biased datasets and data leakage [15, 16]. In addition, ML-based approaches often show limited structural diversity in contrast to physics-based models. To account for this drawback, methods that integrate thermodynamic priors have been developed, e.g., MXfold2, which uses free energy calculations for regularization within a neural network [17]. The inverse problem, i.e., designing sequences that fold into a given RNA secondary structure, inherits the difficulties of RNA structure

prediction and introduces further challenges. While computational RNA structure prediction is commonly formulated as mapping a sequence to a single representative structure, inverse folding is a one-to-many problem: many distinct sequences can fold into the same target structure. The search space for sequences grows exponentially with sequence length, and structural constraints introduce long-range dependencies that make local optimization unreliable [18, 19]. Furthermore, not all potential structures correspond to physically realizable RNA folds, and inverse-design pipelines depend on the accuracy of a folding algorithm to evaluate candidate sequences. A wide range of approaches to RNA inverse folding have been developed since the introduction of RNAinverse, spanning heuristic search, evolutionary optimization, constraint-based methods, and more recent machine-learning approaches. Several methods also support additional sequence-level design constraints, including predefined sequence motifs and GC-content control [20–23]. In eM2dRNA, the problem is formulated as a multi-objective search over the sequence space, where an evolutionary algorithm optimizes the structural agreement similarity with the target alongside the thermodynamic stability [6]. SAMFEO similarly uses a non-ML optimization procedure, maintaining multiple candidate sequences and iteratively refining them according to ensemble-based objectives such as target-structure probability or normalized ensemble defect [24]. Several methods start with a random initial sequence and iteratively modify it until the predicted structure matches the given target. For instance, LEARNA [4] uses a reinforcement-learning approach that constructs an RNA sequence nucleotide by nucleotide. After a complete sequence has been generated, RNAfold predicts its structure and the structural difference to the target is used as the reward; for near-solutions, an additional local improvement step modifies mismatching positions. SentRNA and EternaBrain leverage human-designed solutions from the EteRNA platform to train supervised models [5, 25]. Despite their success, ML-based approaches rely on iterative optimization loops, making them sensitive to initialization and prone to premature convergence [26].

Diffusion models provide a generative framework that is well aligned with the inverse-folding problem. In diffusion models, samples are gradually corrupted by noise and reconstructed through a learned reverse process. Once optimized, the model can generate new samples starting from random noise and iteratively denoising it, i.e., sampling from the learned distribution [27]. These sampling dynamics naturally support diverse outputs conditioned on the same input, addressing the one-to-many nature of inverse folding. Furthermore, diffusion models have demonstrated strong performance in continuous domains such as image and molecular generation [28, 29], and have recently been applied to the modeling of biological sequence and structure [30–32]. However, nucleotide sequences are discrete, which limits the applicability of Gaussian diffusion. Multinomial diffusion addresses this limitation by replacing Gaussian noise with categorical transitions [33]. Recent works explore this formulation for RNA modeling. RNADiffFold applies multinomial diffusion to RNA secondary structure prediction by denoising contact maps conditioned on sequence information [34]. Other approaches focus on tertiary structure–based RNA design, generating sequences compatible with fixed 3D backbones [35, 36], or on unconditional RNA sequence generation where secondary structure is evaluated post-hoc [37]. These methods rely on limited

3D-structural data [15] or do not directly address secondary-structure–conditioned inverse folding.

Most inverse-folding approaches rely on iterative optimization loops coupled to external folding oracles for structure-guided sequence generation, which can limit scalability and reduce sequence diversity. A generative framework that directly models the conditional distribution of nucleotide sequences given a target secondary structure, without requiring external guidance during generation, remains underexplored. Adopting such an approach would decouple sequence generation from costly structure evaluation steps and enable more diverse and scalable design. Although recent approaches employ diffusion models on nucleotide sequence data, they typically operate on three-dimensional representations [35], apply Gaussian noise to one-hot–encoded sequences [32], or rely on latent-space transformations [38] rather than defining a diffusion process over discrete categorical tokens.

In this work, we propose a novel approach to RNA inverse folding at the secondary-structure level, formulated as a fully discrete multinomial diffusion process defined directly in nucleotide space. In contrast to existing approaches, we preserve the categorical nature of the sequence space. This aligns the generative process with the discrete design objective and enables structure-conditioned generation without external folding evaluation, potentially providing advantages for scalability and sequence diversity. To this end, we introduce two complementary denoising model architectures: a 2D variant that operates on sequence-derived pairwise representations aligned with the contact map, and a computationally efficient 1D variant that incorporates structural guidance through partner-aware features and attention biasing. We evaluate both architectures with respect to structural consistency and sequence diversity on held-out RNA data of the datasets also used for model development and on the expert-curated mostly synthetic Eterna100 benchmark. We further assess potential benefits by including additional feedback from a structure prediction model for the 2D variant and evaluate all models' capability for constrained generation in the form of partial sequence completion. Finally, we benchmark the models both in terms of performance and computational scaling behavior against the state-of-the-art RL–based method LEARNA.

2 Material and Methods

In this section, we detail the data processing, discrete diffusion framework, model architectures, training strategy, and evaluation protocol of our multinomial diffusion approach for structure-guided RNA sequence design. All source code is publicly available at https://github.com/smAIL-WS/RNA-struct-diff.

2.1 Data

We used the bpRNA and RNAStrAlign datasets provided with RNADiffFold [34]. CD-HIT-EST [39] confirmed that the bpRNA dataset contained no sequences sharing more than 80% sequence identity, while RNAStrAlign 54% of sequences exceeded this threshold. This difference in redundancy was taken into account when constructing the homology-controlled test set described later. Each sample contained an RNA sequence

with its respective structure annotated as pairwise indices. The pairwise indices were converted into binary contact maps. To enable efficient batching, sequences were grouped into length bins of 64, 128, 256, and 640 nucleotides and padded to the corresponding bin size. Since the maximum sequence length in the training set was 640 nucleotides, the final bin size was increased to 640, as only a small number of samples exceeded 512 nucleotides (n = 528). To mitigate the bias toward short sequences, we applied data balancing following RNADiffFold by two-fold oversampling of the two bins containing the longest sequences, which comprise comparatively few samples. The final data split into training (80%), validation (10%), and test sets (10%) resulted in 26,978 training sequences and 3,060 test sequences, comprising 1,304 bpRNA and 1,756 RNAStrAlign sequences. However, to ensure that the sequences used for evaluation are sufficiently distinct from all data used for model development, the primary in-distribution evaluation was restricted to the bpRNA test subset. These sequences were screened against the complete bpRNA and RNAStrAlign training and validation data using CD-HIT-EST, and sequences sharing more than 80% sequence identity with any training or validation sequence were excluded. This yielded a sequence-identity-controlled test set of 1,148 sequences from the original 1,304 bpRNA test sequences. Sequences were tokenized using a standard encoding of the four nucleotides (A, C, G, U). In addition, we extended the alphabet to distinguish between paired and unpaired nucleotide states, yielding four additional tokens and resulting in eight nucleotide tokens plus a padding token. With this representation, nucleotides were mapped to their paired form according to the target contact map. This effectively embedded structural information in the sequence, enabled the model to learn pairing-dependent patterns directly from the data.

2.2 Discrete Diffusion Formulation

Gaussian diffusion models are not directly applicable to discrete data. We therefore follow the multinomial diffusion framework of Hoogeboom et al. 2021, which extends denoising diffusion to categorical variables. Given a sequence, the forward process gradually replaces categorical entries with uniform noise according to

$$q(x_t \mid x_{t-1}) = \mathrm{Cat}\left(x_t \;\middle|\; (1-\beta_t)\, x_{t-1} + \beta_t \frac{1}{\mathcal{K}} \mathbf{1} \right), \tag{1}$$

where $\mathcal{K}$ is the alphabet size and β_t follows a cosine noise schedule. In the reverse process, a neural network predicts the categorical distribution over x_0, from which the exact posterior $q(x_{t-1} \mid x_t, x_0)$ is computed. Training minimizes the negative evidence lower bound (ELBO), which upper-bounds the negative log-likelihood and contains Kullback–Leibler (KL) divergence terms:

$$\mathcal{L} = \mathbb{E}_q \Big[-\log p_\theta(x_0 \mid x_1) + \sum_{t=2}^{T} \mathrm{KL}\big(q(x_{t-1} \mid x_t, x_0) \parallel p_\theta(x_{t-1} \mid x_t) \big) \Big]. \tag{2}$$

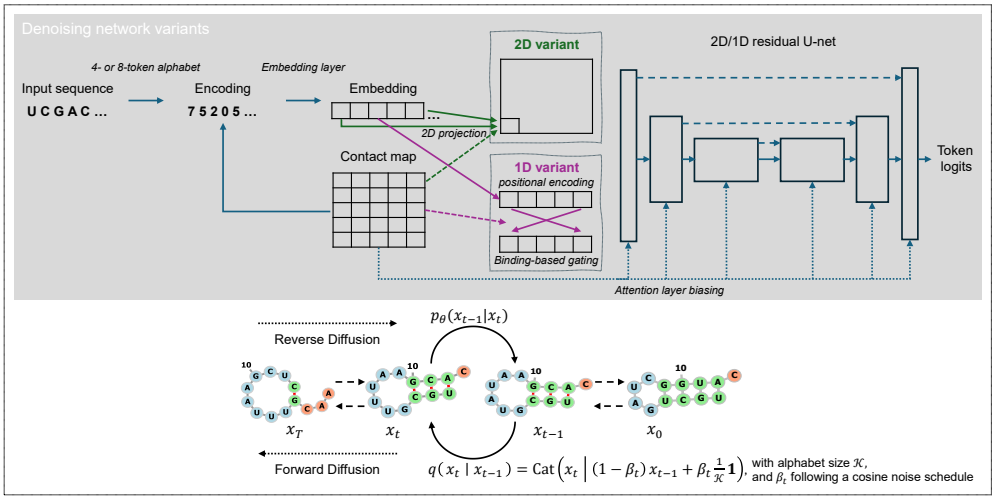


Fig. 1 Visualization of our structure-guided multinomial diffusion RNA design approaches. The forward diffusion process iteratively noises the original sequence, which is reversed by a U-net. Both for the 2D and the 1D variant, the sequence is first encoded and embedded. For the 2D variant, this embedding is projected into a 2D-space aligned with the contact map. For the 1D variant, we conduct a sinusoidal positional encoding and incorporate structural context by mapping each position to its base-pairing partner. Both then leverage a residual U-net, with additional structural information by attention layer biasing.

2.3 Denoising Models

The denoising network that transforms noise into valid samples plays a central role. This component directly determines the model's ability to capture complex data distributions and generate high-quality output, making it a crucial design choice.

2.3.1 Architectural Approaches

We consider two complementary architectures for the denoising network, subsequently called the 2D and 1D variants, see Figure 1. Both build on a U-Net with skip connections, where each resolution level consists of two residual blocks followed by a linear attention layer, and the diffusion timestep is embedded and added to the features in every residual block. A final 1×1 convolution head predicts per-nucleotide class logits. The variants differ in how structural dependencies are represented: the 2D variant operates on pairwise sequence representations aligned with the contact map, whereas the 1D variant incorporates structure through position-aware features and attention biasing within a sequence-based representation.

2D variant: The 2D variant considers sequence generation as a segmentation-like problem over pairwise representations. A sequence of length L is embedded and projected into a $L \times L$ representation aligned with the contact map, enabling direct concatenation of sequence-derived embeddings and the target contact map. Structural guidance is integrated at multiple resolutions: resized versions of the contact map are added to feature maps at each U-Net scale and modulate linear attention

blocks via learned scaling, providing structural bias throughout the denoising process. In an extension of the 2D variant, feedback from a structure prediction model is additionally considered. For this purpose, the structure at time step t is predicted using RNAfold [10], and the predicted and target contact maps are combined into an agreement–difference bias, resulting in our 2D model with RNAfold guidance.

1D variant: The 1D variant operates directly on sequence-level representations using sinusoidal positional encodings to preserve nucleotide order. The structural context is incorporated by mapping each position to its base-pairing partner, as defined by the contact map, and combining partner and local feature representations through a gated residual connection. Additionally, the contact map guides the attention layers. In the full attention layers at the bottleneck and before the output head, it enters as an additive bias on the attention scores, so each nucleotide attends directly to its pairing partner. In the linear attention layers, which never form an explicit attention matrix, a pairwise bias is not possible. Instead, the keys of paired nucleotides are scaled by a learned factor, which raises the attention paid to paired positions in general.

2.3.2 Partial sequence completion

Since both structure and sequence motifs contribute to RNA function, partial sequence design, in which a predefined motif or sequence fragment is fixed and the remaining positions are generated to conform to the input secondary structure, represents a natural extension of the structure-constrained sequence design task. We implemented partial sequence completion by randomly fixing a subset of positions during training and generating the remaining positions leveraging our models. A ratio of up to 30% of positions was fixed to their ground-truth nucleotide, using a 1:2:7 ratio for no fixation, fixation of randomly distributed positions, and fixation of a single continuous block, respectively. Fixed nucleotides were enforced externally during sampling, since explicit conditioning led to overfitting in preliminary experiments. For testing, the same distribution of fixed positions was used. For partial sequence generation, we considered two approaches: (i) fixation to the ground-truth nucleotides, as during training, and (ii) fixation to randomly sampled nucleotides as a control without true motif information. Due to the higher computational cost of the 2D architecture and only a minor improvement in performance shown in preliminary experiments, partial sequence completion experiments were conducted and evaluated only for the 1D variant.

2.3.3 Model optimization

For the optimization of the denoising models, we followed the multinomial diffusion objective described by [33]. For each batch, a diffusion timestep t was sampled and the corresponding noisy sequence x_t was generated based on the clean sequence x_0. The denoising network predicted the categorical distribution over x_0, which we used to compute the exact reverse posterior and the negative ELBO loss. Padding tokens were excluded from the loss calculation as they do not represent valid nucleotide positions. In our extended alphabet that accounts for paired and unpaired states, nucleotides were mapped to their paired representation prior to diffusion and mapped

back to the unpaired alphabet for evaluation. We used Adam with gradient clipping as optimizer, and hyperparameters were tuned using Bayesian optimization [40] to minimize the validation negative ELBO. The search space included learning rate (10^{-5}–10^{-2}, log-uniform), diffusion steps ($\{500, 1000, 2000, \ldots, 10000\}$), model width ($\{32, 64, 128\}$), dropout (0–0.2), weight decay (10^{-6}–10^{-2}, log-uniform), gradient clipping ($\{\text{None}, 1.0, 5.0\}$), batch size ($\{16, 32, 64, 96, 128, 256, 350\}$), U-Net depth configuration ($\{(1, 4, 8), (1, 2, 4, 8)\}$), scheduler patience (2–8) and scheduler factor (0.1–0.8), as well as alphabet size (4 vs. 8). Validation was performed periodically using the token-masked negative ELBO, and early stopping (patience 20, maximum 1000 epochs) was applied for regularization. Distributed training is supported with synchronized loss aggregation.

2.4 Post-processing and candidate selection

For each target structure, the diffusion models generated 100 candidate sequences. The secondary structure of each candidate was predicted using MXfold2 [17] and compared with the target structure using normalized RNAforester similarity [10, 41]. Candidates were ranked according to this score, and the 10 highest-ranking sequences were retained for evaluation. Our comparison partners LEARNA and SAMFEO, see Evaluation, incorporate folding-based evaluation and sequence optimization within their native inference procedures. For both, therefore, 10 final sequences were sampled.

2.5 Evaluation

For each generated sequence, secondary structure was predicted using MXfold2 [17]. Structural agreement with the target was quantified using normalized RNAforester similarity [], which compares RNA secondary structures through alignment of their tree representations. Additionally, we calculated the base-pair F1 score for each predicted structure, defined as the harmonic mean of precision and recall. Precision is the fraction of predicted base pairs present in the target, while recall is the fraction of target base pairs recovered in the prediction. Thermodynamic compatibility with the target was assessed for each sequence using normalized ensemble defect (NED), defined as the expected fraction of nucleotides with incorrect pairing states under the thermodynamic folding ensemble, calculated using the ViennaRNA package [10].

Sequence recovery was calculated as the fraction of nucleotide positions matching the native reference sequence, to assess the similarity of generated sequences to their native counterparts.

To assess sequence diversity, variability among generated sequences for the same target was quantified using Levenshtein distance normalized by sequence length, defined as the minimum number of single-character edits (insertions, deletions, or substitutions) required to transform one sequence into another. Pairwise sequence identity was additionally calculated as the fraction of matching nucleotide positions between generated sequences.

For comparison with established inverse-folding approaches, we evaluated LEARNA, a state-of-the-art RL-based method [4], and SAMFEO, an optimization-based method [24], see Introduction. Both methods were applied to the in-distribution

test set and to the Eterna100 benchmark. SAMFEO was run using the probability-defect objective and default settings, with 10 targets and 32 workers. In cases where a baseline method failed to return any sequence due to timeouts or non-solvability, comparisons involving that method were restricted to the targets for which sequences were successfully generated. Runtime scalability was evaluated separately on target structures with a length of 50 to 165 nucleotides, randomly sampled from our in-distribution test set, and sequence lengths not represented in the test set were omitted. Since GPU and CPU runtimes are not comparable, computational efficiency is reported as absolute time per generated sequence. In addition to runtimes, the generated sequences were evaluated using the structural similarity and diversity metrics described above. We conducted our experiments on a server equipped with two AMD EPYC 9654 96-core CPUs and four NVIDIA H100 NVL GPUs with 96 GB of memory each. Evaluation runs used a single GPU.

3 Results and Discussion

Subsequently, we first evaluate our structure-guided multinomial diffusion models with respect to structural accuracy and sequence variability, including a comparison between models trained on the combined dataset and on bpRNA alone. We then analyze performance for specific RNA families and partial sequence completion. Finally, we benchmark our models against LEARNA and SAMFEO in terms of performance and scalability.

3.1 Performance on the In-Distribution Test Set

For the in-disturbtion test set, all models achieved high structural accuracy in terms of the relative RNAforester similarity score. Median best performance across all targets reached 0.82 and 0.79 for the 2D and 1D model, respectively, see Figure 2a and Table 1. However, the mean scores of 0.69 and 0.7 indicated outliers where the models failed to generate accurate samples. Notably, MXfold2 [17] was not able recreate the reference structures for all reference sequences in the train and test sets, which could partly explain poor performance for certain input structures. The additional RNAfold guidance we considered for the 2D variant only provided slight improvements with median and mean scores of 0.82 and 0.70, respectively. Training the 1D model on bpRNA data only provided a slight improvement, with a median normalized RNAforester similarity of 0.82 and a mean similarity of 0.72. In addition to structural accuracy, the models differed in the diversity of sequences they generate for a given target, see Figure 2a and Table 1. The median normalized Levenshtein distances were 0.54 for the 2D model, 0.55 for the 2D model with RNAfold guidance, 0.55 for the 1D model, and 0.54 for the bpRNA-only 1D model. For comparison, expected values for randomized sequences were estimated by Monte Carlo simulation, yielding median values ranging from 0.57 to 0.53 for sequence lengths between 64 and 640 nucleotides. Thus, the overall sequence diversity was relatively high, even comparable to random sequences.

Table 1 Performance on the in-distribution test set. Values represent medians across targets. RNAforester similarity and F1 score (both based on MXfold2 predictions) measure structural accuracy, while NED measures the expected fraction of nucleotides with incorrect pairing. Levenshtein distance and pairwise identity measure sequence diversity among generated samples. Sequence recovery denotes identity to the native sequence.

Model	RNAforester ↑	Levenshtein ↑	Pairwise id. ↓	Seq. rec.	NED ↓	F1 ↑
1D bound	0.787	0.549	0.278	0.270	0.335	0.759
1D bpRNA	0.816	0.538	0.302	0.308	0.320	0.779
1D MXfold	0.859	0.544	0.289	0.270	0.282	0.818
2D bound	0.803	0.551	0.286	0.275	0.331	0.747
2D RNAfold	0.822	0.538	0.330	0.281	0.319	0.757
LEARNA	1.000	0.455	0.427	0.261	0.070	0.884
SAMFEO	1.000	0.242	0.740	0.293	0.003	1.000

Note: Arrows indicate the direction of preferable values (↑ higher, ↓ lower). LEARNA metrics were calculated only for targets for which LEARNA successfully generated solutions. RNAforester similarity and F1 scores are based on MXfold2-predicted structures.

3.2 Performance on the Eterna100 Benchmark

The performance on the expert-curated and mainly synthetic Eterna100 benchmark showed a different pattern, see Figure 2b. Although several targets, particularly those corresponding to naturally occurring motifs or less constrained structural elements, were achieved with high accuracy, the majority of Eterna structures yielded lower normalized RNAforester similarities, with median scores of 0.49 (2D), 0.66 (2D with RNAfold guidance), 0.59 (1D), and 0.56 (bpRNA-only 1D), see Table 2. The RNAfold-guided 2D variant performed best among the models, suggesting that additional thermodynamic information can improve performance on out-of-distribution targets. Sequence diversity remained close to the expected values for random sequences, with median normalized Levenshtein distances of 0.54 (2D), 0.51 (2D with RNAfold guidance), 0.54 (1D), and 0.54 (bpRNA-only 1D), as shown in Figure 2b.

3.3 Performance across RNA families

Models may learn family-specific sequence patterns through homology rather than structural features. To examine this, we analyzed the performance across the RNA families represented in the test set. Since we observed similar behavior for all the models presented in this paper, and since the computationally efficient 1D variant shows competitive performance, we focused on the results of the 1D variant for subsequent analyses to improve readability. In general, the normalized RNAforester similarities of our models correlate negatively with sequence length (Spearman $\rho = -0.61$, $p < 1 \times 10^{-118}$), indicating that longer sequences are harder to generate, as expected. However, sequence length alone does not fully explain the observed differences across families. Rather short, highly conserved RNA classes achieve near-perfect structural matches, with tRNAs ($n = 72$) and C4 antisense RNAs ($n = 12$) both reaching median normalized RNAforester similarities of 1.0. Similarly strong performance is

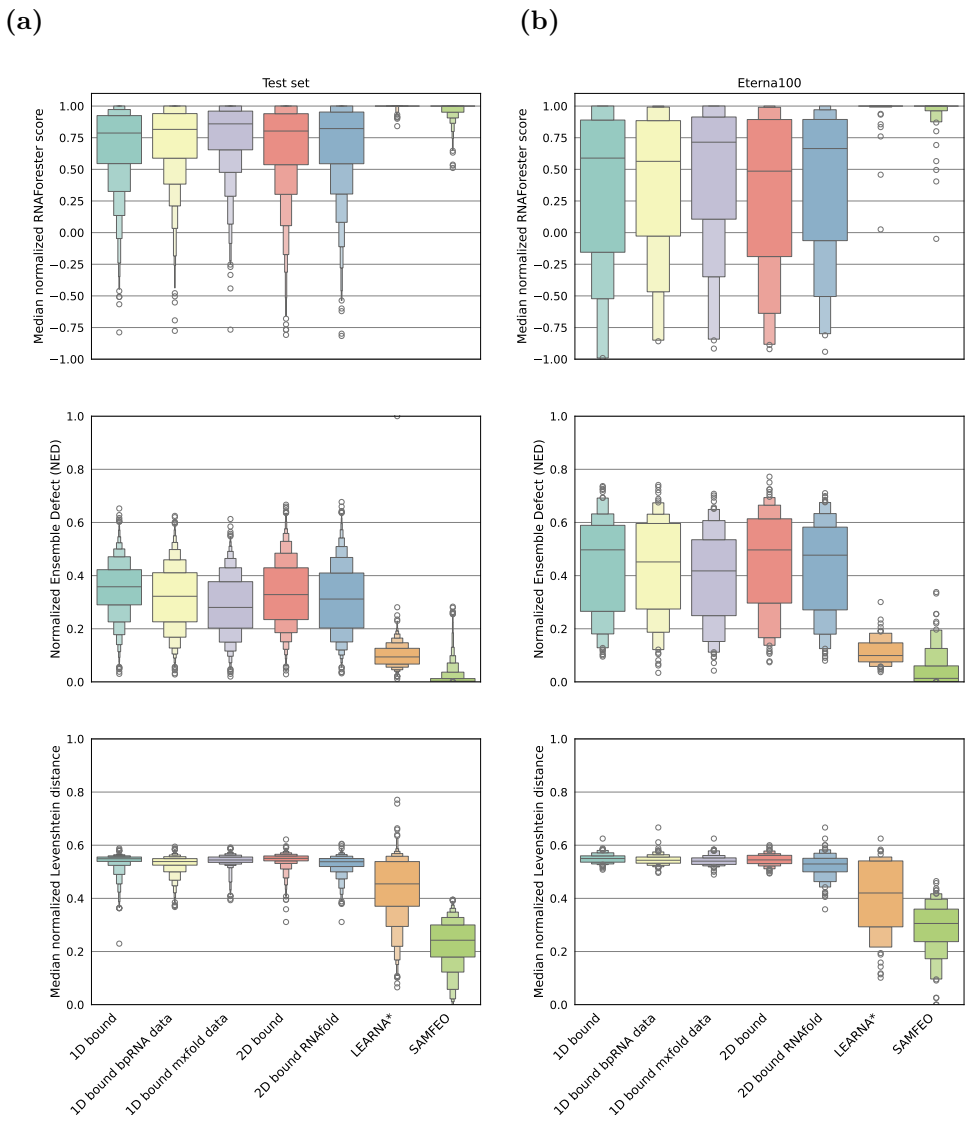


Fig. 2 Comparative performance. (a) Results for the natural RNA test set, and (b) the Eterna100 benchmark. Upper panels: Structural accuracy measured by normalized RNAforester similarity between the predicted fold of generated sequences and the reference structure. Middle panels: Normalized ensemble defect (NED) of the generated sequences with respect to the target structure. Lower panels: Sequence diversity per target, represented as the median normalized pairwise Levenshtein distance between generated sequences. (*Only targets for which LEARNA produced sequences are shown.)

observed for miRNAs ($n = 203$, median=0.91), which are also rather short and structurally constrained. Intermediate performances were observed for families with greater structural heterogeneity, including snoRNAs, snRNAs, riboswitches, tmRNAs, and Group II introns, with median similarities ranging from 0.64 to 0.74. The lowest

Table 2 Performance on the Eterna100 benchmark. Values represent medians across targets. RNAforester similarity and F1 score (both based on MXfold2 predictions) measure structural accuracy, while NED measures the expected fraction of nucleotides with incorrect pairing. Levenshtein distance and pairwise identity measure sequence diversity among generated samples.

Model	RNAforester ↑	Levenshtein ↑	Pairwise id. ↓	NED ↓	F1 ↑
1D bound	0.589	0.550	0.284	0.497	0.545
1D bpRNA	0.563	0.543	0.295	0.451	0.575
1D MXfold	0.715	0.540	0.301	0.418	0.656
2D bound	0.486	0.545	0.306	0.497	0.483
2D RNAfold	0.664	0.529	0.365	0.477	0.575
LEARNA	1.000	0.431	0.525	0.099	0.805
SAMFEO	1.000	0.306	0.673	0.013	0.971

Note: Arrows indicate the direction of preferable values (↑ higher, ↓ lower). LEARNA metrics were calculated only for targets for which LEARNA successfully generated solutions. RNAforester similarity and F1 scores are based on MXfold2-predicted structures.

scores were observed for RNase P RNAs, particularly the bacterial RNase P A subclass (median=0.25), consistent with the greater difficulty of generating sequences for large and structurally complex RNA families. In addition, we observed a negative Spearman correlation between sequence length and normalized Levenshtein distance ($\rho = -0.51$, $p = 5 \times 10^{-88}$). The lowest median normalized Levenshtein distance, 0.42, was observed for tRNAs, the second-largest class in the test set. However, other large groups, such as miRNAs, snoRNAs, and riboswitches, displayed diversity levels close to those of randomized sequences, with normalized distances of 0.55, 0.54, and 0.54, respectively. This indicates that, for highly conserved structures such as tRNAs, the model may rely heavily on learned sequence patterns associated with these folds, thereby reducing the sequence space it explores. Implementing mechanisms that introduce greater sequence variability, such as enforcing a specific GC content in the generated sequences, may help counteract this tendency.

3.4 Exploring partial sequence completion

With respect to constrained sequence generation by fixing a random subset of sequence positions during training, our 1D model achieved comparable normalized RNAforester values to unconditional generation for the test set, with a median similarity score of 0.88. Our second experiment using a fixation to randomly sampled nucleotides as a control without true motif information led to a slightly decreased median similarity of 0.83. However, certain random constraints can introduce incompatibilities with the target secondary structure, leading to sequences that cannot satisfy the structure constraint.

3.5 Benchmarking

Comparing the generated sequences shows that both LEARNA and SAMFEO outperform our models in terms of structural similarity (Figure 2 and Tables 1, 2) .

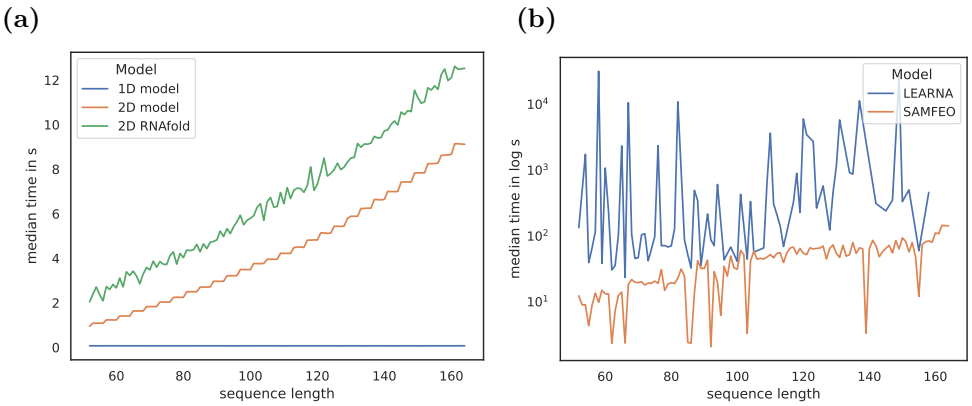


Fig. 3 Computational runtime benchmark. (a) median time for a sequence per sequence length for our diffusion models, (b) median time for a sequence per sequence length for LEARNA and SAMFEO.

However, LEARNA successfully generated sequences for only 701 of 1,148 test targets and 61 of 100 Eterna100 targets, whereas our models and SAMFEO produced results for all test sequences. On the full in-distribution test set, our diffusion models achieved median best RNAforester similarities ranging from 0.79 to 0.86, increasing to 0.88–0.94 when restricting the comparison to the 701 targets successfully solved by LEARNA. On this subset, both LEARNA and SAMFEO achieved a median best similarity of 1. Similarly, both comparison methods achieved lower NED values than our diffusion models. SAMFEO's strong performance is consistent with its use of an ensemble-based probability-defect objective. The variability among the sequences generated by LEARNA and SAMFEO was comparable to or lower than that of our models, particularly for SAMFEO.

This discrepancy was more pronounced for the Eterna100 set. Across all targets, our diffusion models achieved median best RNAforester similarities ranging from 0.49 to 0.72, increasing to 0.72–0.81 when restricting the comparison to the 61 targets successfully solved by LEARNA. On this subset, both LEARNA and SAMFEO achieved a median best similarity of 1 and lower NED values, whereas our diffusion-based models maintained higher sequence variability, with median normalized Levenshtein distances ranging from 0.53 to 0.55, compared with 0.43 for LEARNA and 0.3 for SAMFEO.

Sequence recovery was generally low, with median values ranging from 0.27 to 0.31, indicating that the generated sequences differed substantially from the native reference sequences.

Since LEARNA and SAMFEO are primarily CPU-based optimization methods, whereas our diffusion models were implemented and optimized for GPU inference, absolute runtimes are not directly comparable in an equal-hardware benchmark. We therefore used wall-clock time to characterize general scaling trends. LEARNA and SAMFEO showed generally longer and non-linear increases with respect to sequence length in runtime, likely reflecting dependence on target-structure complexity, as

shown in Figure 3b. In contrast, our diffusion models showed an approximately linear increase in runtime with sequence length, with the 1D architecture exhibiting the highest throughput, as shown in Figure 3a.

3.6 Usage of thermodynamic-informed oracles as benchmarking

Of note is the different use of computational folding models across inverse-folding methods. LEARNA and SAMFEO incorporate thermodynamic folding information provided by the ViennaRNA framework directly into their optimization procedures.. In contrast, the diffusion models we present are trained on annotated reference structures and do not require a folding model during generation. One of the variants we present receives RNAfold-derived information as soft guidance, but this information is not enforced through a hard constraint or explicit folding-based loss. To estimate the extent of disagreement in the underlying data, we compared annotated reference structures with MXfold2 and RNAfold predictions. Exact agreement with MXfold2 was 4.03–4.70% for bpRNA and 15.08–16.57% for RNAStrAlign, while agreement with RNAfold ranged from 1.31–2.15% and 4.20–5.39%, respectively. Consistent with this, a 1D model trained and evaluated on MXfold2-predicted structures for the bpRNA training, validation, and test sets, termed "1D bound MXfold data" in Figure 3, achieved the highest performance among our diffusion models, with median normalized RNAforester similarity of 0.86 for the test set and 0.72 for the Eterna100 set. These results show that oracle-based benchmarks are useful, but they also measure agreement with the folding model used for evaluation, not only agreement with the annotated reference structure.

3.7 Advantages, limitations, and future directions

Multinomial diffusion offers a promising approach for adapting generative diffusion models to the discrete nucleotide space, as illustrated here for RNAs. Given a structural input, the proposed models generate sequences with both high structural similarity to the target and a high degree of variability. While the 2D architecture demonstrates slightly better performance in terms of both structural similarity and variability, the computational efficiency of the 1D model makes it a promising alternative, particularly for longer sequences or applications requiring the generation of large candidate sets. Our models do not match the structural performance of target-specific thermodynamic optimization methods such as LEARNA and SAMFEO. However, their inference time scales more predictably with sequence length, and they generate more diverse sets of candidate sequences. This increased diversity may be advantageous when multiple structurally compatible candidates are required for downstream screening or when additional sequence-level constraints need to be considered. Furthermore, the diffusion framework supports constrained sequence generation through partial sequence completion, allowing known motifs or fixed nucleotide positions to be preserved during sequence design. A limitation of our current approach is that structural information is primarily learned from sequence–structure associations in the training data rather than from an explicit thermodynamic objective,

and purely ML-based RNA design has been suggested to remain limited by biases in the available training data [42]. The improved out-of-distribution performance of the RNAfold-informed 2D model, together with the strong performance of LEARNA and SAMFEO, suggests that incorporating thermodynamic information may complement the homology-based sequence diffusion process. The development and integration of a fully differentiable thermodynamic folding algorithm therefore represents a promising future direction. Further work could also make greater use of diffusion models' inherent capability for constrained sequence generation, such as inpainting.

4 Conclusion

In this study, we demonstrate that multinomial diffusion can be adapted to generate diverse RNA sequences conditioned on target secondary structures. Although the proposed models do not match thermodynamic optimization methods in structural accuracy, they offer predictable runtime scaling, high sequence diversity, and support for constrained sequence generation. These results establish multinomial diffusion as a complementary approach to existing RNA inverse-folding methods, while leaving room for future research.

Declarations

Competing interests

The authors declare that they have no conflicting interests.

Funding

We thank the Free State of Bavaria for funding the positions of G.B. and F.H. within the framework of the Hightech Agenda Bavaria.

Data availability

All source code and the models with checkpoints are available at GitHub: https://github.com/smAIL-WS/RNA-struct-diff

Author contributions statement

G.B.: Conceptualization, Data curation, Methodology, Software, Validation, Visualization, Writing – original draft. D.W.: Conceptualization, Writing – review & editing. F.H.: Conceptualization, Supervision, Funding acquisition, Writing – review & editing.

Use of generative AI and AI-assisted technologies

The authors used ChatGPT and Claude to improve the clarity of the English language and to assist with code development. The authors subsequently reviewed, tested and significantly revised the resulting text and code, and assume full responsibility for the final publication.

References

[1] Micura, R., Höbartner, C.: Fundamental studies of functional nucleic acids: Aptamers, riboswitches, ribozymes and DNAzymes. Chemical Society Reviews **49**(20), 7331–7353 (2020) https://doi.org/10.1039/D0CS00617C

[2] Traber, G.M., Yu, A.-M.: The Growing Class of Novel RNAi Therapeutics. Molecular Pharmacology **106**(1), 13–20 (2024) https://doi.org/10.1124/molpharm.124.000895

[3] Polack, F.P., Thomas, S.J., Kitchin, N., Absalon, J., Gurtman, A., Lockhart, S., Perez, J.L., Marc, G.P., Moreira, E.D., Zerbini, C., Bailey, R., Swanson, K.A., Roychoudhury, S., Koury, K., Li, P., Kalina, W.V., Cooper, D., Frenck, R.W., Hammitt, L.L., Türeci, Ö., Nell, H., Schaefer, A., Ünal, S., Tresnan, D.B., Mather, S., Dormitzer, P.R., Şahin, U., Jansen, K.U., Gruber, W.C.: Safety and Efficacy of the BNT162b2 mRNA Covid-19 Vaccine. New England Journal of Medicine **383**(27), 2603–2615 (2020) https://doi.org/10.1056/NEJMoa2034577

[4] Runge, F., Hutter, F.: Machine Learning for RNA Design: LEARNA. In: Churkin, A., Barash, D. (eds.) RNA Design: Methods and Protocols, pp. 63–93. Springer, New York, NY (2025). https://doi.org/10.1007/978-1-0716-4079-1_5

[5] Shi, J., Das, R., Pande, V.S.: SentRNA: Improving computational RNA design by incorporating a prior of human design strategies. arXiv (2019) https://doi.org/10.48550/arXiv.1803.03146 [cs, q-bio, stat]

[6] Rubio-Largo, Á., Lozano-García, N., Granado-Criado, J.M., Vega-Rodríguez, M.A.: Solving the RNA inverse folding problem through target structure decomposition and Multiobjective Evolutionary Computation. Applied Soft Computing **147**, 110779 (2023) https://doi.org/10.1016/j.asoc.2023.110779

[7] Butcher, S.E., Pyle, A.M.: The Molecular Interactions That Stabilize RNA Tertiary Structure: RNA Motifs, Patterns, and Networks. Accounts of Chemical Research **44**(12), 1302–1311 (2011) https://doi.org/10.1021/ar200098t

[8] Chen, Y., Varani, G.: RNA Structure. In: Encyclopedia of Life Sciences. John Wiley & Sons, Ltd, ??? (2010). https://doi.org/10.1002/9780470015902.a0001339.pub2

[9] Zuker, M.: Mfold web server for nucleic acid folding and hybridization prediction. Nucleic acids research **31**(13), 3406–3415 (2003)

[10] Lorenz, R., Bernhart, S.H., Höner zu Siederdissen, C., Tafer, H., Flamm, C., Stadler, P.F., Hofacker, I.L.: ViennaRNA Package 2.0. Algorithms for Molecular Biology **6**(1), 26 (2011) https://doi.org/10.1186/1748-7188-6-26

[11] Mathews, D.H., Andre, T.C., Kim, J., Turner, D.H., Zuker, M.: An Updated

Recursive Algorithm for RNA Secondary Structure Prediction with Improved Thermodynamic Parameters. In: Molecular Modeling of Nucleic Acids. ACS Symposium Series, vol. 682, pp. 246–257. American Chemical Society, ??? (1997). Chap. 15. https://doi.org/10.1021/bk-1998-0682.ch015

[12] Akiyama, M., Sato, K., Sakakibara, Y.: A max-margin training of RNA secondary structure prediction integrated with the thermodynamic model. Journal of Bioinformatics and Computational Biology **16**(06), 1840025 (2018) https://doi.org/10.1142/S0219720018400255

[13] Jain, D.S., Gupte, S.R., Aduri, R.: A Data Driven Model for Predicting RNA-Protein Interactions based on Gradient Boosting Machine. Scientific Reports **8**(1), 9552 (2018) https://doi.org/10.1038/s41598-018-27814-2

[14] Zhang, H., Zhang, C., Li, Z., Li, C., Wei, X., Zhang, B., Liu, Y.: A New Method of RNA Secondary Structure Prediction Based on Convolutional Neural Network and Dynamic Programming. Frontiers in Genetics **10** (2019) https://doi.org/10.3389/fgene.2019.00467

[15] Schneider, B., Sweeney, B.A., Bateman, A., Cerny, J., Zok, T., Szachniuk, M.: When will RNA get its AlphaFold moment? Nucleic Acids Research **51**(18), 9522–9532 (2023) https://doi.org/10.1093/nar/gkad726

[16] Bernett, J., Blumenthal, D.B., Grimm, D.G., Haselbeck, F., Joeres, R., Kalinina, O.V., List, M.: Guiding questions to avoid data leakage in biological machine learning applications. Nature Methods **21**(8), 1444–1453 (2024) https://doi.org/10.1038/s41592-024-02362-y

[17] Sato, K., Akiyama, M., Sakakibara, Y.: RNA secondary structure prediction using deep learning with thermodynamic integration. Nature Communications **12**(1), 941 (2021) https://doi.org/10.1038/s41467-021-21194-4

[18] Aviran, S., Incarnato, D.: Computational Approaches for RNA Structure Ensemble Deconvolution from Structure Probing Data. Journal of Molecular Biology **434**(18), 167635 (2022) https://doi.org/10.1016/j.jmb.2022.167635

[19] Churkin, A., Retwitzer, M.D., Reinharz, V., Ponty, Y., Waldispühl, J., Barash, D.: Design of RNAs: Comparing programs for inverse RNA folding. Briefings in Bioinformatics **19**(2), 350–358 (2018) https://doi.org/10.1093/bib/bbw120

[20] Hofacker, I.L., Fontana, W., Stadler, P.F., Bonhoeffer, L.S., Tacker, M., Schuster, P.: Fast folding and comparison of RNA secondary structures. Monatshefte für Chemie / Chemical Monthly **125**(2), 167–188 (1994) https://doi.org/10.1007/BF00818163

[21] Garcia-Martin, J.A., Clote, P., Dotu, I.: RNAiFold: A web server for RNA inverse folding and molecular design. Nucleic Acids Research **41**(W1), 465–470 (2013)

https://doi.org/10.1093/nar/gkt280

[22] Kleinkauf, R., Mann, M., Backofen, R.: antaRNA: Ant colony-based RNA sequence design. Bioinformatics **31**(19), 3114–3121 (2015) https://doi.org/10.1093/bioinformatics/btv319

[23] Churkin, A., Retwitzer, M.D., Reinharz, V., Ponty, Y., Waldispühl, J., Barash, D.: Design of RNAs: Comparing programs for inverse RNA folding. Briefings in Bioinformatics **19**(2), 350–358 (2018) https://doi.org/10.1093/bib/bbw120

[24] Zhou, T., Dai, N., Li, S., Ward, M., Mathews, D.H., Huang, L.: RNA design via structure-aware multifrontier ensemble optimization. Bioinformatics **39**(Supplement_1), 563–571 (2023) https://doi.org/10.1093/bioinformatics/btad252

[25] Koodli, R.V., Keep, B., Coppess, K.R., Portela, F., Participants, E., Das, R.: EternaBrain: Automated RNA design through move sets and strategies from an Internet-scale RNA videogame. PLOS Computational Biology **15**(6), 1007059 (2019) https://doi.org/10.1371/journal.pcbi.1007059

[26] Goldie, A.D., Lu, C., Jackson, M.T., Whiteson, S., Foerster, J.N.: Can Learned Optimization Make Reinforcement Learning Less Difficult? Advances in Neural Information Processing Systems **37**, 5454–5497 (2024) https://doi.org/10.52202/079017-0177

[27] Ho, J., Jain, A., Abbeel, P.: Denoising Diffusion Probabilistic Models. In: Advances in Neural Information Processing Systems, vol. 33, pp. 6840–6851. Curran Associates, Inc., ??? (2020)

[28] Hoogeboom, E., Satorras, V.G., Vignac, C., Welling, M.: Equivariant Diffusion for Molecule Generation in 3D. In: Proceedings of the 39th International Conference on Machine Learning, pp. 8867–8887. PMLR, ??? (2022)

[29] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-Resolution Image Synthesis With Latent Diffusion Models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10684–10695 (2022)

[30] Lee, J.S., Kim, J., Kim, P.M.: Score-based generative modeling for de novo protein design. Nature Computational Science **3**(5), 382–392 (2023) https://doi.org/10.1038/s43588-023-00440-3

[31] Gao, Z., Tan, C., Li, S.Z.: DiffSDS: A language diffusion model for protein backbone inpainting under geometric conditions and constraints. arXiv (2023) https://doi.org/10.48550/arXiv.2301.09642 [cs, q-bio]

[32] DaSilva, L.F., Senan, S., Patel, Z.M., Janardhan Reddy, A., Gabbita, S.,

Nussbaum, Z., Valdez Córdova, C.M., Wenteler, A., Weber, N., Tunjic, T.M., Ahmad Khan, T., Li, Z., Smith, C., Bejan, M., Karmel Louis, L., Cornejo, P., Connell, W., Wong, E.S., Meuleman, W., Pinello, L.: DNA-Diffusion: Leveraging Generative Models for Controlling Chromatin Accessibility and Gene Expression via Synthetic Regulatory Elements. bioRxiv, 2024–0201578352 (2024) https://doi.org/10.1101/2024.02.01.578352

[33] Hoogeboom, E., Nielsen, D., Jaini, P., Forré, P., Welling, M.: Argmax Flows and Multinomial Diffusion: Learning Categorical Distributions. arXiv (2021) https://doi.org/10.48550/arXiv.2102.05379 [stat]

[34] Wang, Z., Feng, Y., Tian, Q., Liu, Z., Yan, P., Li, X.: RNADiffFold: Generative RNA secondary structure prediction using discrete diffusion models. Briefings in Bioinformatics **26**(1), 618 (2025) https://doi.org/10.1093/bib/bbae618

[35] Huang, H., Lin, Z., He, D., Hong, L., Li, Y.: RiboDiffusion: Tertiary structure-based RNA inverse folding with generative diffusion models. Bioinformatics **40**(Supplement_1), 347–356 (2024) https://doi.org/10.1093/bioinformatics/btae259

[36] Hou, D., Zhang, S., Ma, M., Lin, H., Wan, Z., Zhao, H., Zhou, R., He, X., Wei, X., Ju, D., Zeng, X.: A Hyperbolic Discrete Diffusion 3D RNA Inverse Folding Model for Functional RNA Design. Journal of Chemical Information and Modeling **65**(13), 6568–6584 (2025) https://doi.org/10.1021/acs.jcim.5c00527

[37] Mora-Poblete, F., Maldonado, C., Henrique, L., Uhdre, R., Scapim, C.A., Mangolim, C.A.: Multi-trait and multi-environment genomic prediction for flowering traits in maize: A deep learning approach. Frontiers in Plant Science **14** (2023) https://doi.org/10.3389/fpls.2023.1153040

[38] Li, Z., Ni, Y., Beardall, W.A.V., Xia, G., Das, A., Stan, G.-B., Zhao, Y.: DiscDiff: Latent Diffusion Model for DNA Sequence Generation. arXiv (2024) https://doi.org/10.48550/arXiv.2402.06079 [cs, q-bio]

[39] Li, W., Godzik, A.: Cd-hit: A fast program for clustering and comparing large sets of protein or nucleotide sequences. Bioinformatics **22**(13), 1658–1659 (2006) https://doi.org/10.1093/bioinformatics/btl158

[40] Snoek, J., Larochelle, H., Adams, R.P.: Practical Bayesian Optimization of Machine Learning Algorithms. In: Advances in Neural Information Processing Systems, vol. 25. Curran Associates, Inc., ??? (2012)

[41] Höchsmann, M., Töller, T., Giegerich, R., Kurtz, S.: Local similarity in RNA secondary structures. Proceedings. IEEE Computer Society Bioinformatics Conference **2**, 159–168 (2003)

[42] Mukherjee, S., Zakh, R., Churkin, A., Barash, D.: RNA design: Update on

computational frameworks and programs for inverse RNA folding. Briefings in Bioinformatics **27**(3), 327 (2026) https://doi.org/10.1093/bib/bbag327