RIBOGEN: RNA SEQUENCE AND STRUCTURE CO-GENERATION WITH EQUIVARIANT MULTIFLOW

Dana Rubin
MIT CSAIL
MIT Media Lab, Molecular Machines
danaru@mit.edu

Allan dos Santos Costa
Center for Bits and Atoms
MIT Media Lab, Molecular Machines
allanc@mit.edu

Manvitha Ponnapati
Center for Bits and Atoms
MIT Media Lab, Molecular Machines

Joseph Jacobson
Center for Bits and Atoms
MIT Media Lab, Molecular Machines

ABSTRACT

Ribonucleic acid (RNA) plays fundamental roles in biological systems, from carrying genetic information to performing enzymatic function. Understanding and designing RNA can enable novel therapeutic application and biotechnological innovation. To enhance RNA design, in this paper we introduce RiboGen, the first deep learning model to simultaneously generate RNA sequence and all-atom 3D structure. RiboGen leverages the standard Flow Matching with Discrete Flow Matching in a multimodal data representation. RiboGen is based on Euclidean Equivariant neural networks for efficiently processing and learning three-dimensional geometry. Our experiments show that RiboGen can efficiently generate chemically plausible and self-consistent RNA samples, suggesting that co-generation of sequence and structure is a competitive approach for modeling RNA.

1 INTRODUCTION

Ribonucleic acid (RNA) is a fundamental biomolecule that stands at the intersection of modern biology and the origins of life. RNA has proven to be a versatile molecule, playing key roles in messaging (Crick, 1970), catalytic functions (Altman & Guerrier-Takada, 1983), regulation, and diverse biological processes through its complex 3D structures (Fire et al., 1998). While traditional computational methods faced limitations in decoding RNA structure and facilitating RNA design, deep learning emerged as powerful approach to accurately predict RNA structures, enhance RNA engineering, and unlock new insights into its functional roles. Existing deep learning models often predict RNA structure from sequence or design sequences for target structures independently. However, the ability to simultaneously generate both sequence and structure remains a largely unexplored space in deep learning-based modeling of RNA. This co-generation capability can enable exploration of the sequence-structure landscape in novel ways. To address this gap, in this paper we introduce RiboGen for joint generation of all-atom structure and sequence of RNA. RiboGen is based on a Multiflow (Campbell et al., 2024) model that draws on Flow Matching (Lipman et al., 2022; Liu et al., 2022) and Discrete Flow (Gat et al., 2024; Campbell et al., 2024) for its generation. We train a large model and evaluate its chemical validity and self-consistency. Our results showcase the capabilities of RiboGen for generation and highlight that co-generation models are a promising avenue for modeling of RNA.

1.1 RELATED WORK

Previous work in deep learning for RNA design has made significant progress in computational approaches to predict RNA structures (Shen et al., 2024; Abramson et al., 2024), RNA-RNA interactions, and the design of novel RNA sequences. Recent progress in generative modeling for RNA focuses on sequence and structure generation through Denoising Diffusion Probabilistic Mod-

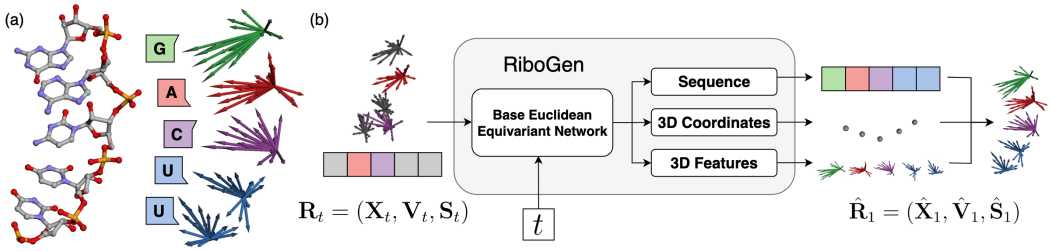

Figure 1: **RNA Sequence and Structure Co-Generation**: (a) Traditional molecular structure showing the nucleotides with atoms and bonds. Right side demonstrates how each nucleotide (G, A, C, U) is represented as both a discrete sequence element (colored boxes) and associated 3D point cloud representation (colored directional features) centered around the $C3'$ atom. (b) The RiboGen model architechture: the model takes noised input of sequence and geometric features $\mathbf{R}_t$, and a time parameter t, process them through the base network and simultaneously predicts three components: the RNA sequence, central coordinates, and 3D features. These components are combined to produce the final RNA structure prediction $\hat{\mathbf{R}}_1$.

els (DDPM) (Ho et al., 2020) or Flow Matching (Lipman et al., 2022). MMDiff (Morehead et al., 2023a) uses discrete DDPM to co-generate sequence and structure of RNA, DNA and proteins. Our approach instead employs Flow Matching and its discrete variant (Campbell et al., 2024). RNA-FrameFlow (Anand et al., 2024) represents RNA through rigid body frames and uses flow matching to generate 3D backbones, employing inverse folding model gRNAde (Joshi et al., 2023) to obtain sequences. In contrast, while similarly using Flow Matching for 3D generation, our approach additionally models the discrete sequence components of RNA generation. RNAFlow Nori & Jin (2024) uses a GNN conditioned on protein structure and sequence to generate RNA sequences, which are then processed by RoseTTAFold2NA Baek (2024) to predict backbone structure; this method additionally conditions on protein structure as input. Our approach instead focuses on isolated RNA, learning unconditional direct sequence-structure generation. Recent application of the Multiflow (Campbell et al., 2024) framework to protein sequence-structure design has demonstrated the power of joint generation. Our approach builds upon on these insights from protein design, and adapts them to the RNA design domain by enabling the joint generation of RNA sequences and all-atom structures.

2 Methods

2.1 RNA Representation

We represent an RNA molecule as a sequence and a 3D gas of geometric features $\mathbf{R} = (\mathbf{S}, \mathbf{X}, \mathbf{V})$ (Figure 1.a), where:

- $\mathbf{S} \in \mathcal{S}^N$ is sequence of length N, formed out of standard nucleotides $\mathcal{S} = \{A, C, G, U\}$.
- $\mathbf{X} \in \mathbb{R}^{N\times 3}$ contains the 3D coordinates of the C3' atom for each nucleotide, chosen as the **reference center**.
- $\mathbf{V} \in \mathbb{R}^{N\times 24\times 3}$ are geometric features encoding the relative position of up to 24 heavy atoms per nucleotide to its center at C3', in canonical ordering. This representation encompasses both the sugar-phosphate backbone atoms and the base atoms, allowing for complete reconstruction of the RNA structure. Nucleotides that have fewer than 24 heavy atoms have their corresponding channels of $\mathbf{V}$ padded with zeros.

After generating the three components, the predicted vectors $\mathbf{V}$ are added to the predicted centers $\mathbf{X}$ and the sequence $\mathbf{S}$ is utilized for labeling nucleotides and atomic types, for full reconstruction of 3D atomic coordinates. This representation encodes both the chemical identity and geometry of each nucleotide while preserving rotational and translational equivariance, which is essential for downstream learning via Euclidean Equivariant Neural Networks.

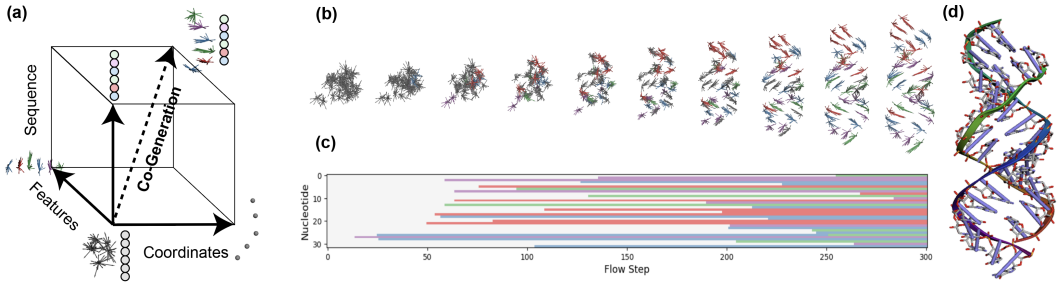

Figure 2: **Multiflow for RNA Sequence, Backbone and Atomistic Structure**: (a) Schematic representation of our Multiflow approach, demonstrating the three dimensions- sequence, coordinates, and features. (b) Visualization of the RNA structure generation across multiple time steps. (c) Visualization of the Discrete flow matching used for sequence prediction in the model, where each color represents a different nucleotide. (d) Final product, a complete generated RNA molecule.

2.2 Flow Matching

To model the distribution of 3D RNA coordinates $\mathbf{X}$ and features $\mathbf{V}$ we use Flow Matching (Lipman et al., 2022; Liu et al., 2022; Albergo et al., 2023). Flow Matching parameterizes a conditional probability path $\rho_t(\mathbf{X}_t|\mathbf{X}_1)$ on time t by learning a conditional velocity field $\hat{v}_t^\theta(\mathbf{X}_t) \approx v_t(\mathbf{X}_t|\mathbf{X}_1)$ that transforms samples from a prior distribution $\mathbf{X}_0 \sim \rho_0 = \mathcal{N}$ to a target data distribution $\mathbf{X}_1 \sim \rho_1 = \rho_D$. To learn this transport, we use the standard form of Flow Matching to obtain a noised version of $\mathbf{X}_1$ via the linear interpolant and its associated velocity:

$$\mathbf{X}_t = (1-t)\mathbf{X}_0 + t\mathbf{X}_1 \tag{1}$$

$$v_t(\mathbf{X}_t|\mathbf{X}_1) = \mathbf{X}_1 - \mathbf{X}_0 \tag{2}$$

We build our model to reconstruct the target $\hat{\mathbf{X}}^\theta_{1|\mathbf{X}_t} \approx \mathbf{X}_1$ from its noised counterpart $\mathbf{X}_t$. We follow the reparameterization of (Jing et al., 2024; Pooladian et al., 2023) and obtain the learned conditional velocity through:

$$v_t^\theta(\mathbf{X}_t) = \frac{1}{(1-t)}\left(\hat{\mathbf{X}}^\theta_{1|\mathbf{X}_t} - \mathbf{X}_t\right) \approx v_t(\mathbf{X}_t|\mathbf{X}_1) \tag{3}$$

We then sample our learned model via integration $\mathbf{X}_1 = \mathbf{X}_0 + \int_0^1 v_t^\theta(\mathbf{X}_t)dt$ where $\mathbf{X}_0 \sim \mathcal{N}$.

2.3 Discrete Flow Matching

While the standard form of Flow Matching is effective for continuous data, it is not appropriate for categorical domains. Hence, for modeling the RNA sequence we employ the extended framework of Discrete Flow Matching (Gat et al., 2024; Campbell et al., 2024). In this setting, the sequence data $\mathbf{S} \in \mathcal{S}^N$ is described over a vocabulary $\mathcal{S}$. We parameterize a discrete flow by describing the velocity field over a probability vector on this categorical space:

$$\mathbf{S}_t \sim \text{Cat}((1-t)\delta_{\mathbf{S}_0} + t\delta_{\mathbf{S}_1}) \tag{4}$$

$$v_t(\mathbf{S}_t|\mathbf{S}_1) = \delta_{\mathbf{S}_1} - \delta_{\mathbf{S}_0} \tag{5}$$

where $\delta_{\mathbf{S}} \in \mathbb{R}^{N\times|\mathcal{X}|}$ is the Dirac delta representation of $\mathbf{S}$ and $\text{Cat}(\cdot)$ denotes the categorical distribution. We learn a model to predict the probability vector $p^\theta_{1|\mathbf{S}_t} \approx \delta_{\mathbf{S}_1}$. In similar reparameterization to Equation 3, we obtain the approximate conditional velocity through:

$$v_t^\theta(\mathbf{S}_t) = \frac{1}{(1-t)}\left(\hat{p}^\theta_{1|\mathbf{S}_t} - \delta_{\mathbf{S}_t}\right) \approx v_t(\mathbf{S}_t|\mathbf{S}_1) \tag{6}$$

2.4 MULTIFLOW

To generate full RNA representations, we use Multiflow (Campbell et al., 2024) and train a neural network to learn the multimodal velocity field $d\mathbf{R}_t = (d\mathbf{S}_t, d\mathbf{X}_t, d\mathbf{V}_t)$ given jointly noised data $\mathbf{R}_t$ and time t (Figure 1.b). We employ the standard Flow Matching for $\mathbf{X}$ and $\mathbf{V}$ and its discrete counterpart for $\mathbf{S}$. This decomposition enables the model to separately capture the distributions of sequence, backbone, and atomic positions, allowing for unconditional generation or conditional generation based on specific structural or sequence constraints (Figure 2.a), such as structure prediction or inverse folding.

2.5 ARCHITECTURE AND TRAINING

We use Euclidean-Equivariant Neural Networks (Geiger & Smidt, 2022) for processing our RNA representation. To handle the different modalities of $\mathbf{R}$, our model consists of a base network that feeds into 3 headers for each data component: sequence, coordinates and 3D features. The sequence header predicts a probability vector $\hat{p}_{\mathbf{R}_1} \in \mathbb{R}^{N \times |\mathcal{S}|}$, while coordinate and feature headers predict equivariant variables $\hat{\mathbf{X}}_1$ and $\hat{\mathbf{V}}_1$. We train the model to reconstruct the original structure $(\mathbf{X}, \mathbf{V})$ and sequence $\mathbf{S}$:

$$\mathcal{L} = \mathcal{L}_{\text{struct}} + \mathcal{L}_{\text{seq}} = \text{Mean}(\|V(\mathbf{X}_1, \mathbf{V}_1) - V(\hat{\mathbf{X}}_1, \hat{\mathbf{V}}_1)\|^2) + \text{CrossEntropy}(\hat{p}_{\mathbf{S}_1}, \delta_{\mathbf{S}_1}) \quad (7)$$

where $V(\mathbf{X}, \mathbf{V}) \in \mathbb{R}^{N_A \times N_A \times 3}$ is the 3D vector map between every atom in the system (size N_A).

3 RESULTS

We leverage the RNASolo dataset (Adamczyk et al., 2022) which consists of extracted individual RNA structures from the Protein Databank (PDB) (Berman et al., 2000) to train our model. Following (Anand et al., 2024), we filter the full dataset to resolution < 4Å and sequence lengths between 40 and 150 to a total dataset size of 6090 data points. The dataset exhibits significant length imbalances which led to biased model performance initially, to address this, we implemented a length-balanced sampling that ensures uniform representation of RNA sequences for training as described in Appendix A.1. We train our model with batch size of 64 for 120k steps on 4 GPUs. Our model's flow process is trained on 300 timesteps. To evaluate our model, we follow (Anand et al., 2024) and sample 50 RNA structures at each sequence length from 40-150 with step size of 10.

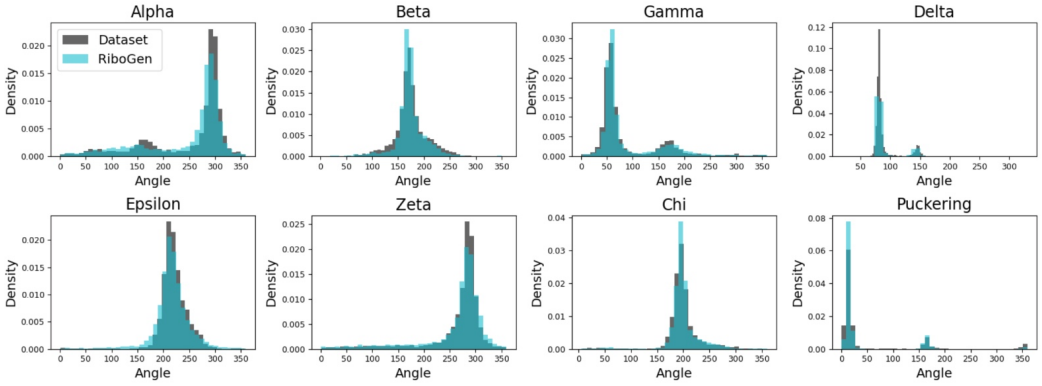

Figure 3: **RiboGen Chemical Analysis**: Distribution comparison of key RNA geometric parameters between the training dataset and 50 random samples of RiboGen generations across all lengths. (5 from each length) The analyzed parameters include alpha, beta, gamma, chi dihedral angles, and ribose puckering phase, which are strong indicators of RNA backbone and chemical validity.

3.1 CHEMICAL VALIDITY

To assess the chemical validity of our generated structures, we analyzed key geometric parameters that define RNA backbone and base conformations. Using MDAnalysis (Gowers et al., 2016)

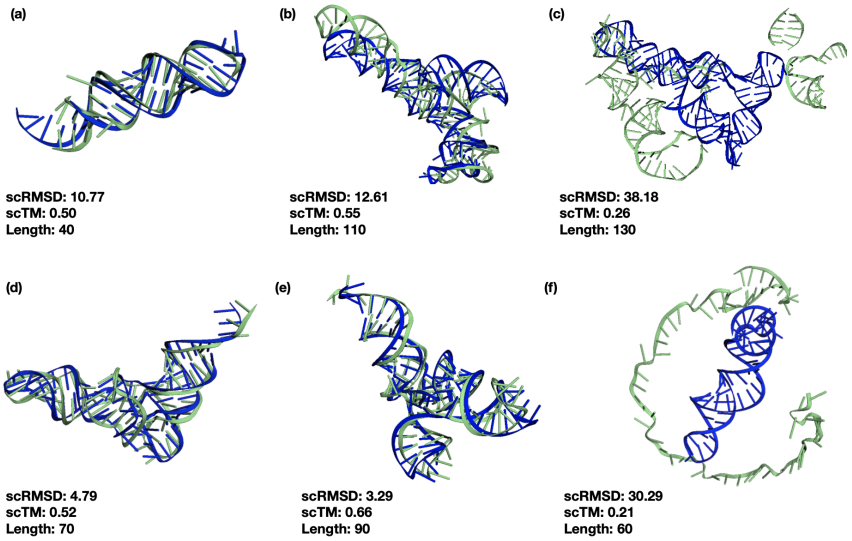

Figure 4: **Self-consistency Visualization of RiboGen's Joint Sequence-Structure Generation Aligned with Boltz Structure:** RiboGen-generated RNA structures (green) aligned with Boltz structure predictions (blue) derived from the corresponding co-generated sequences of RiboGen. Six examples across different sequence lengths demonstrate varying degrees of structural agreement. Notably, in some cases (c, f) RiboGen generates fragmented or unfolded structures, suggesting failure modes in the sampling process for long or structurally complex sequences.

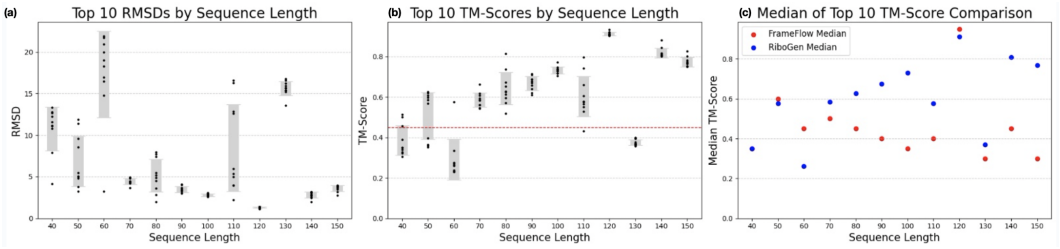

Figure 5: **Self-Consistency Evaluation**: (a) RMSD and (b) TM-score between our generated structures and Boltz-1 predictions across different sequence lengths (40-150 nucleotides), showing the top 10 generated structures for each length. The TM-score ranges from 0 to 1, with higher values indicating better structural agreement, while lower RMSD values indicate better structural similarity. (c) Median of TM-scores of top 10 generated structures: The plot compares RiboGen's and FrameFlow's medians across various RNA sequence lengths and illustrates that RiboGen achieves higher TM-scores for RNA sequences between 70-150 nucleotides, excluding 120 which has similar median. FrameFlow demonstrates comparable performance for shorter sequences but shows decreased structural accuracy as sequence length increases.

(Michaud-Agrawal et al., 2011) we computed all dihedral angles and the pseudo-angle for the ribose pucker. Figure 3 shows the distributions of the dihedral angles across the training data (representing experimentally determined structures) and 50 randomly selected RiboGen structures, 5 from each sequence length. The results demonstrate that for most angles RiboGen generated structures capture the dihedral angles distributions and the general trends of the training set, though with some discrepancies. Our RiboGen structures show broader distribution across the Alpha angle indicating mild divergence from the experimental data in this specific torsion. Overall these results suggest that RiboGen has successfully learned the geometric constraints of RNA molecules.

3.2 SELF-CONSISTENCY

To evaluate the quality and biological plausibility of our generated RNA, we employed a self consistency validation process. For each generated RNA molecule, we extracted its sequence and used Boltz-1 (Wohlwend et al., 2024) to obtain a reference structure. We quantify structural similarity between our generated structure and Boltz's structure using two complementary metrics: Root Mean Square Deviation (RMSD) and Template Modeling score (TM-score). We calculated these metrics across all 50 samples for each sequence length. Following (Anand et al., 2024), in Figure 5(a) and (b) we report results on 10 best-performing samples per length. Our predictions scTM scores in 5(b) demonstrate lower variance than in Anand et al. (2024), suggesting better consistency and generalization across different lengths. In Figure 5(c) we observe that RiboGen achieves higher median TM-scores than RNA-FrameFlow for most sequence lengths above 70 nucleotides. This may highlight the benefits of co-generation, especially for longer RNA.

3.3 STRUCTURAL EVALUATION

To evaluate RiboGen's performance in generating valid RNA structure-sequence pairs, we utilized the metrics from the evaluation suite proposed by (Anand et al., 2024). While RNA-FrameFlow focuses on structure generation alone, RiboGen jointly generates both the sequence and its corresponding structure. Therefore, instead of using gRNAde (Joshi et al., 2025) sequences and corresponding RhoFold (Shen et al., 2024) structures to calculate TM-score, we folded the co-generated sequences using Boltz-1 and computed the TM-score after aligning them to the generated structures. While RNA-FrameFlow employs an additional 8-shot inverse folding process when calling gRNAde eight times for each backbone, RiboGen performs one-shot co-generation of both sequence and structure in a single sampling process. This highlights RiboGen's potential for simpler and more efficient RNA design workflows, avoiding the need for expensive post-hoc sequence inference. Following (Anand et al., 2024), samples with TM-score ≥ 0.45 were considered valid. To assess diversity, we measured the number of unique qTM clusters among the valid samples and normalized this by the total number of valid samples. Although RiboGen is sampling jointly RNA sequences and structures, it achieved performance on par with RNA-FrameFlow in terms of backbone validity and diversity. Our results demonstrate competitive metrics and efficient sampling, validating RiboGen as a promising baseline for joint RNA structure-sequence generation.

Model	**Sampling Steps N_T**	**% Validity ↑**	**Diversity ↑**	**Time (s) ↓**
RiboGen	100	27.17	**0.604**	**1.18**
	200	32.17	0.553	4.50
	300	**34.17**	0.585	9.06
RNA-FrameFlow *	10	16.7	**0.62**	–
	50	**41.0**	0.61	4.74
	100	20.0	0.61	–
MMDiff *	100	0	–	27.30

Table 1: **Performance comparison of unconditional RNA structure generation models**. This table presents RiboGen's performance across varying flow sampling timesteps (N_T), alongside other models. Metrics include structural validity, diversity measured through qTM clusters, and computational cost in seconds per generation. * Results for RNA-FrameFlow and MMDiff (Morehead et al., 2023b) methods are reported from (Anand et al., 2024).

4 CONCLUSION

In this paper, we introduced RiboGen, the first generative model to jointly produce RNA sequences and their corresponding all-atom 3D structures by learning a single multi-modal Flow field. Our approach leverages Flow Matching for continuous structural components and Discrete Flow Matching for sequence generation within the Multiflow framework. We demonstrated that RiboGen can generate RNA structures that are chemically plausible, as evidenced by the distributions of key geometric parameters including dihedral angles and ribose puckering. Our model outperforms previous

approaches in self-consistency evaluation (scTM score) across a wide range of sequence lengths, particularly for longer RNAs. Our early results demonstrate that the generation of RiboGen provides a competitive and efficient RNA design workflow, suggesting sequence-structure co-generation to be a strong approach for RNA modeling. As the field of RNA design continues to grow in importance for therapeutic and biotechnology applications, we believe that generative models like RiboGen will become increasingly valuable tools for exploring and engineering RNA.

ACKNOWLEDGMENTS

This research was made possible through the support of the Eleven Eleven Foundation, the Center for Bits and Atoms, and the MIT Media Lab Consortium. Their support was fundamental in enabling this work.

REFERENCES

J. Abramson, J. Adler, J. Dunger, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 2024. doi: https://doi.org/10.1038/s41586-024-07487-w.

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

Michael S Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *arXiv preprint arXiv:2303.08797*, 2023.

Sidney Altman and Cecile Guerrier-Takada. The rna moiety of ribonuclease p is the catalytic subunit of the enzyme. *Cell*, 35(3):849–857, 1983. doi: 10.1016/0092-8674(83)90117-4.

Rishabh Anand, Chaitanya K. Joshi, Alex Morehead, Arian R. Jamasb, Charles Harris, Simon V. Mathis, Kieran Didi, Bryan Hooi, and Pietro Liò. Rna-frameflow: Flow matching for de novo 3d rna backbone design. *arXiv preprint*, 2024. URL https://doi.org/10.48550/arXiv.2406.13839.

Minkyung Baek. Towards the prediction of general biomolecular interactions with ai. *Nature Methods*, 21:1382–1383, 2024. URL https://www.nature.com/articles/s41592.

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.

Andrew Campbell, Jason Yim, Regina Barzilay, Tom Rainforth, and Tommi Jaakkola. Generative flows on discrete state-spaces: Enabling multimodal flows with applications to protein co-design. *arXiv preprint arXiv:2402.04997*, 2024.

Francis Crick. Central dogma of molecular biology. *Nature*, 227:561–563, 1970. doi: 10.1038/227561a0.

Andrew Fire, SiQun Xu, Mary K. Montgomery, Steven A. Kostas, Samuel E. Driver, and Craig C. Mello. Potent and specific genetic interference by double-stranded rna in caenorhabditis elegans. *Nature*, 391(6669):806–811, 1998. doi: 10.1038/35888.

Itai Gat, Tal Remez, Neta Shaul, Felix Kreuk, Ricky T. Q. Chen, Gabriel Synnaeve, Yossi Adi, and Yaron Lipman. Discrete flow matching, 2024. URL https://arxiv.org/abs/2407.15595.

Mario Geiger and Tess Smidt. e3nn: Euclidean neural networks. *arXiv preprint arXiv:2207.09453*, 2022. doi: 10.48550/arXiv.2207.09453. URL https://doi.org/10.48550/arXiv.2207.09453. draft.

R. J. Gowers, M. Linke, J. Barnoud, T. J. E. Reddy, M. N. Melo, S. L. Seyler, D. L. Dotson, J. Domanski, S. Buchoux, I. M. Kenney, and O. Beckstein. MDAnalysis: A Python package for the rapid analysis of molecular dynamics simulations. In S. Benthall and S. Rostrup (eds.), *Proceedings of the 15th Python in Science Conference*, pp. 98–105, Austin, TX, 2016. SciPy. doi: 10.25080/Majora-629e541a-00e.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv preprint arXiv:2006.11239*, 2020. doi: 10.48550/arXiv.2006.11239. URL https://doi.org/10.48550/arXiv.2006.11239.

Bowen Jing, Bonnie Berger, and Tommi Jaakkola. Alphafold meets flow matching for generating protein ensembles. *arXiv preprint arXiv:2402.04845*, 2024.

Chaitanya K. Joshi, Arian R. Jamasb, Ramon Viñas, Charles Harris, Simon V. Mathis, Alex Morehead, Rishabh Anand, and Pietro Liò. gRNAde: Geometric Deep Learning for 3D RNA Inverse Design. *arXiv preprint arXiv:2305.14749*, 2023. URL https://doi.org/10.48550/arXiv.2305.14749.

Chaitanya K. Joshi, Arian R. Jamasb, Ramon Viñas, Charles Harris, Simon V. Mathis, Alex Morehead, Rishabh Anand, and Pietro Liò. grnade: Geometric deep learning for 3d rna inverse design, 2025. URL https://arxiv.org/abs/2305.14749.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

N. Michaud-Agrawal, E. J. Denning, T. B. Woolf, and O. Beckstein. MDAnalysis: A Toolkit for the Analysis of Molecular Dynamics Simulations. *J. Comput. Chem.*, 32:2319–2327, 2011. doi: 10.1002/jcc.21787.

Alex Morehead, Jeffrey Ruffolo, Aadyot Bhatnagar, and Ali Madani. Towards joint sequence-structure generation of nucleic acid and protein complexes with se(3)-discrete diffusion, 2023a. URL https://arxiv.org/abs/2401.06151.

Alex Morehead, Jeffrey Ruffolo, Aadyot Bhatnagar, and Ali Madani. Towards joint sequence-structure generation of nucleic acid and protein complexes with se(3)-discrete diffusion. *arXiv preprint arXiv:2401.06151*, 2023b. URL https://doi.org/10.48550/arXiv.2401.06151. Presented at NeurIPS 2023 MLSB Workshop.

Divya Nori and Wengong Jin. Rnaflow: Rna structure & sequence design via inverse folding-based flow matching. *arXiv preprint arXiv:2405.18768*, 2024.

Aram-Alexandre Pooladian, Heli Ben-Hamu, Carles Domingo-Enrich, Brandon Amos, Yaron Lipman, and Ricky T. Q. Chen. Multisample flow matching: Straightening flows with minibatch couplings, 2023. URL https://arxiv.org/abs/2304.14772.

Tian Shen, Zhen Hu, Shuxin Sun, et al. Accurate rna 3d structure prediction using a language model-based deep learning approach. *Nature Methods*, 21:2287–2298, 2024. doi: 10.1038/s41592-024-02487-0. URL https://doi.org/10.1038/s41592-024-02487-0.

Jeremy Wohlwend, Gabriele Corso, Saro Passaro, Mateo Reveiz, Ken Leidal, Wojtek Swiderski, Tally Portnoi, Itamar Chinn, Jacob Silterra, Tommi Jaakkola, and Regina Barzilay. Boltz-1: Democratizing biomolecular interaction modeling. *bioRxiv*, 2024. doi: 10.1101/2024.11.19.624167. URL https://doi.org/10.1101/2024.11.19.624167. Preprint.

A Appendix

A.1 Data Distribution and Data Balancing

In RNA sequence analysis, certain RNA types (such as tRNA, rRNA) are over-represented in the different datasets. As shown in Figure 6, our original dataset exhibited significant length imbalances, with pronounced peaks at specific length ranges (70-79 and 120-129 nucleotides). This imbalance led to biased model performance, where prediction accuracy was significantly higher for over-represented length ranges but suffered for underrepresented ones. To address this issue, we implemented a length-balanced sampling in the dataset class that ensures uniform representation of RNA sequences across the entire length spectrum during training without changing the original dataset. Our algorithm divides RNA data into length buckets, with range of 10 per bucket; 40-49, 50-59, etc. (except of the last bucket which contains 140-150) During training a random length bucket is chosen uniformly, and out of it a random datum from this bucket is sampled. It dynamically balances the dataset during training and allows the model to see all available data while preventing over-represented lengths from dominating the training process. This balancing technique led to significant improvements in the model's performance across all length ranges, particularly for previously underrepresented sequences.

Algorithm 1 Length-Balanced RNA Sequence Sampling

Require: Dataset D containing RNA sequences with lengths $L \in [40, 150]$
Ensure: Uniformly sampled sequence across all length ranges
Preprocessing:
  Group sequences into buckets B by length range (40–49, 50–59, ..., 140–150)
for each sequence $s \in D$ **do**
  Determine length l of s
  Assign s to bucket $B[\lfloor l/10 \rfloor \times 10]$
end for
Sampling:
Select target length range t uniformly at random from available buckets
Return a randomly selected sequence from $B[t]$

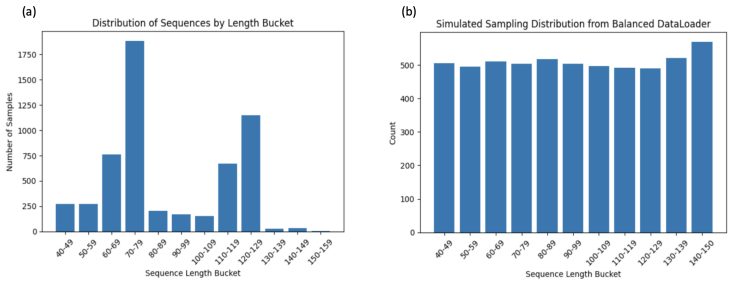

Figure 6: **Distribution of Sequence Lengths of RNAs in Training Dataset, by the buckets, before and after balancing:** (a) the original distribution of RNA sequences in the training dataset, categorized by length buckets of 10 nucleotides each. The distribution exhibits significant imbalance, with pronounced peaks at 70-79 nucleotides and 120-129 nucleotides, likely corresponding to over-represented tRNA and rRNA classes. In contrast, sequences in the 80-109 range and those longer than 130 nucleotides are substantially underrepresented, with fewer than 200 samples in some buckets. (b) our balanced sampling implementation on the training distribution results in all length buckets are uniformly sampled with approximately 500 sequences per bucket during training. This uniform distribution ensures that the model receives equal exposure to RNA sequences across the entire length spectrum from 40 to 150 nucleotides.