Stanford RNA 3D Folding Kaggle Challenge

Advanced Machine Learning Semester Project

Niklas Schmidt
niklas.schmidt@uzh.ch
19-610-583

Elias Müller
eliaswalterjosef.mueller@uzh.ch
18-615-658

Diyar Taskiran
diyar.taskiran@uzh.ch
18-619-775

April 23, 2025

1 Motivation

Understanding ribonucleic acid (RNA) and its structure is a critical task in modern life sciences. While *AlphaFold* [5] has revolutionized protein structure prediction, similar progress in RNA remains limited due to scarce and imbalanced data. Recently, novel RNA 3D structure prediction methods that applies Graph Neural Networks (GNN) and Denoising Diffusion Probabilistic Models (DDPM) have shown superior performance [9]. Their approach focuses on efficiently predicting the structure of RNA substructures (local descriptors) and combining those into the structure of a full molecule. Although this method demonstrates superior performance compared to AlphaFold on RNA targets, its reliance on the DDPM architecture incurs substantial computational demands.

While achieving SOTA results in many generative ML tasks, standard Denoising Diffusion Probabilistic Models (DDPMs) require a backwards pass through a full Markov chain for each sample. This is computationally expensive and often slow. Various techniques have been developed to address this bottleneck. For instance, Denoising Diffusion Implicit Models (DDIMs)[3] generalize the forward process to allow for deterministic, non-Markovian sampling paths, significantly reducing the number of steps needed. More recently, methods like Inductive Moment Matching (IMM) [8] aim to achieve efficient one or few-step sampling through alternative training procedures designed for stability and speed. Both approaches represent efforts to make powerful diffusion-based generative modeling more computationally tractable.

This project aims to integrate more efficient sampling techniques, such as DDIM and IMM, into the RNAgrail framework. By replacing the computationally demanding DDPM sampler with faster alternatives, we seek to substantially reduce the time required for predicting RNA substructures. This enhancement would make advanced RNA 3D structure prediction more accessible and practical, enabling faster exploration of RNA structures while striving to maintain the high accuracy established by the original RNAgrail model.

2 Overview

This project investigates the potential of integrating efficient diffusion sampling techniques into the *RNAgrail* framework for predicting the 3D structures of RNA. Specifically, we aim to adapt the *RNAgrail* architecture by leveraging faster sampling methods, such as DDIM and potentially IMM, to reduce the substantial computational demands associated with standard DDPM sampling.

The project will be structured as follows: First, we will apply the pretrained *RNAgrail* model to the Stanford RNA 3D Folding Kaggle challenge [7] dataset as a baseline benchmark. Next, we will explore and implement variations of the original model architecture by replacing the DDPM sampler with more efficient non-Markovian alternatives like DDIM and potentially investigating one or few-step sampling methods like IMM. Finally, we will compare the performance of these variants, evaluating the trade-offs between computational speed and prediction accuracy in the context of RNA 3D structure prediction.

3 Problem Formulation and Evaluation

3.1 Problem Formulation

The goal of **RNA structure prediction** is to develop a model that takes as input a sequence of RNA and outputs the 3D position of each nucleotide. Formally, given an input RNA sequence $S = \langle n_1, n_2, \ldots, n_N \rangle$, where n_i represents the i^{th} nucleotide in the sequence of length N, the objective is to predict a corresponding sequence of 3D coordinate vectors $P = \langle p_1, p_2, \ldots, p_N \rangle$, where each $p_i = (x_i, y_i, z_i) \in \mathbb{R}^3$ represents the spatial coordinates of the i^{th} nucleotide n_i. The baseline model achieves this by applying a DDPM to a point-cloud of the individual nucleotides, conditioned on the output of a GNN (see Section 4). The goal of this work is to modify the DDPM section of the model by applying DDIM or IMM (see section 4.1).

Therefore, while the overall problem remains predicting the full RNA structure P from the sequence S, this project specifically reformulates the *generative sampling sub-problem* for local descriptors within RNAgrail. We aim to replace the computationally intensive iterative DDPM sampling process with faster alternatives based on DDIM and potentially IMM, evaluating the trade-off between sampling speed and the accuracy of the final predicted RNA structure.

3.2 Evaluation Metric

The evaluation of our results will follow the established procedure used in the Kaggle competition [7]. The quality assessed by computing a Template Modeling Score (TM-score) that was introduced by Zhang and Skolnick [1], that ranges from 0.0 to 1.0, with higher values indicating a better structural match. The score is defined as follows:

$$\text{TM-score} = \max \left(\frac{1}{L_{\text{ref}}} \sum_{i=1}^{L_{\text{align}}} \frac{1}{1 + \left(\frac{d_i}{d_0} \right)^2} \right),$$

where:

- L_{ref} is the number of residues resolved in the experimental (ground truth) structure.
- L_{align} is the number of aligned residues between the prediction and the reference.
- d_i represents the distance (in Angstroms) between the i^{th} pair of aligned residues.
- d_0 is a normalization factor (in Angstroms), which is calculated by

$$d_0 = \begin{cases} 0.3, & \text{if } L_{\text{ref}} < 12; \\ 0.4, & \text{if } 12 \leq L_{\text{ref}} \leq 15; \\ 0.5, & \text{if } 16 \leq L_{\text{ref}} \leq 19; \\ 0.6, & \text{if } 20 \leq L_{\text{ref}} \leq 23; \\ 0.7, & \text{if } 24 \leq L_{\text{ref}} \leq 29; \\ 0.6\sqrt{L_{\text{ref}} - 0.5} - 2.5, & \text{if } L_{\text{ref}} \geq 30. \end{cases}$$

The TM-score is a suitable metric for evaluating RNA 3D structure prediction for several reasons that are presented in the paper by Zhang and Skolnick [1]: First, it improves upon earlier scoring functions like GDT and MaxSub by incorporating a protein size-dependent scale. This adjustment corrects the inherent bias towards smaller proteins, enabling more equitable comparisons across proteins of different lengths. Second, in contrast to methods relying on specific distance thresholds, the TM-score considers every residue pair in the alignment or model. This leads to a more comprehensive evaluation of structural similarity, capturing both localized and overall deviations. Both protein and RNA molecules adopt complex 3D folds that dictate their function. The TM-score's capacity to assess global structural similarity, while showing less sensitivity to minor local variations, makes it a valuable metric for determining the accuracy of predicted RNA structures. Furthermore, using the same metric as the Kaggle competition ensures that our results are directly comparable to others in the field.

Second, we will evaluate the efficiency of our method by comparing its training time against the training time of the original method presented in the referenced papers. To facilitate a direct comparison, we will train both our method and the original method on a representative subset of the training data to then measure and compare the training time per epoch for both approaches over a series of epochs. This analysis will provide insights into the potential improvements in training efficiency offered by our approach.

4 Related Work

The first subsection gives an overview on DDPM and DDIM's. The subsequent explains the baseline implementation by Justyna et al. [9].

4.1 Background Information on DDPM and DDIM

Denoising Diffusion Probabilistic Models (DDPM) [3] learn to reverse a fixed process that gradually adds noise to data (x_0). A network $\epsilon_\theta(x_t, t)$ is trained to predict the noise at each step t of this noising process. Generating samples requires simulating the learned reverse process step-by-step ($x_T \to \cdots \to x_0$) for many steps (T), which is computationally slow.

Denoising Diffusion Implicit Models (DDIM) [3] enable faster sampling using the *same* trained network $\epsilon_\theta(x_t, t)$. It employs a non-Markovian process allowing deterministic updates (for variance $\sigma = 0$) that skip steps via the relation:

$$x_{\tau_{i-1}} = \sqrt{\bar{\alpha}_{\tau_{i-1}}}(\text{predicted } x_0) + \sqrt{1 - \bar{\alpha}_{\tau_{i-1}}}\epsilon_\theta(x_{\tau_i}, \tau_i)$$

This uses a short subsequence of timesteps $\{\tau_1, ..., \tau_S\}$ ($S \ll T$), where the predicted x_0 is derived from x_{τ_i} and $\epsilon_\theta(x_{\tau_i}, \tau_i)$.

Inductive Moment Matching (IMM) [8] aims for highly efficient one-step or few-step generation, often using a *distinct training objective* (e.g., matching moments of distributions across noise levels) rather than requiring a pre-trained DDPM/DDIM model.

4.2 High Level Architecture

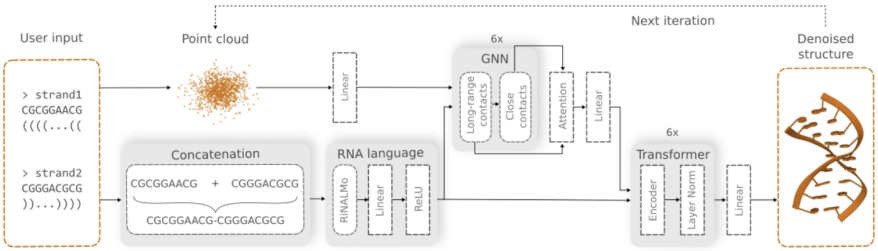

Figure 1: RNA Architecutre

RNAgrail predicts RNA 3D structures from sequence and secondary structure using three main components: an RNA language model, a graph neural network (GNN), and a transformer. It leverages RiNALMo (vgl. [6]), a pre-trained RNA language model for enhanced performance. A modified PAMNet GNN considers both short- and long-range atomic interactions, outputting atom coordinates, types, and residue types and thus provides structural embeddings for the transformer. Finally, a transformer, inspired by AlphaFold's IPA ([2]), refines the 3D structure prediction.

GNN Architecture

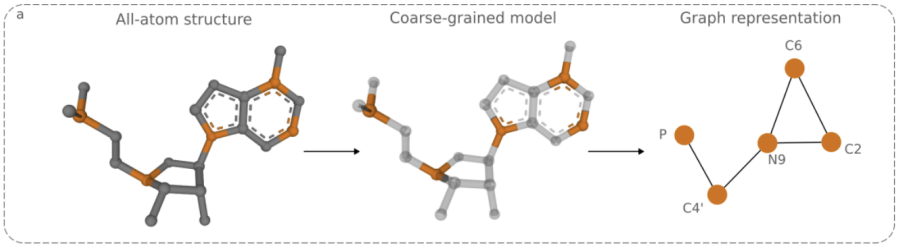

Figure 2: GNN Architecture

The second component for RNA 3D structure prediction utilizes a modified PAMNet (see [4]), a graph neural network, to consider both short (0-5 Å) and long-range (up to 16 Å) atomic interactions by employing two distinct layers and outputting atomic coordinates, types, and residue

types. The GNN has the following characteristics: Each atom is represented as a node. Nodes are connected by covalent and other user-specified interactions according to 2D structural information. This two-way edge representation allows direct connectivity as well as user-specified constraints to be incorporated. The model takes into account close (top) and distant (bottom) interactions between next-nearest atoms. For feature computation, close interactions are of the following kinds: calculate distance and two types of angles: θ (between a node and its one-hop neighbors) and θ (between a node and its two-hop neighbors). For long-range interactions, only the the distances are calculated.

5 Self-Contribution

This project's primary contribution lies in **integrating and evaluating computationally efficient diffusion sampling techniques within the state-of-the-art RNAgrail framework** for RNA 3D structure prediction. Building upon the official RNAgrail implementation [9] (available on GitHub) and pre-trained weights (available on Zenodo), our work will involve the following key technical steps:

1. **Adaptation for Target Dataset:** Modify the existing RNAgrail codebase and input/output pipelines to be directly compatible with the data format and requirements of the Stanford RNA 3D Folding Kaggle challenge [7].
2. **Implementation of Efficient Samplers:** Replace the standard Denoising Diffusion Probabilistic Model (DDPM) sampling mechanism within the generative component of RNAgrail. Specifically, we will implement and integrate:
   - **Denoising Diffusion Implicit Models (DDIM):** Leveraging its deterministic, non-Markovian sampling path to enable faster generation with fewer steps [3].
   - **(Optional/Exploratory) Few-Step Samplers:** Investigate the feasibility of integrating methods like Inductive Moment Matching (IMM) [8] or similar techniques designed for one or few-step generation, potentially requiring modifications to the training procedure if necessary.
3. **Re-training and Fine-tuning:** Train the modified RNAgrail models (using DDIM/IMM samplers) on the Kaggle dataset. This may involve fine-tuning from the original pre-trained weights or training from scratch, depending on preliminary experiments and computational feasibility.
4. **Comparative Analysis:** Systematically evaluate the trade-offs between prediction accuracy (using TM-score, see Section 3.2) and computational efficiency (sampling speed, training time) for the different sampler implementations compared to the original RNAgrail model.

The expected outcome is a modified RNAgrail variant that achieves significantly faster structure prediction, particularly during the sampling phase, making advanced RNA structure prediction more practical, while quantifying any potential impact on prediction accuracy.

6 Evaluation Protocol

We will evaluate the performance of our modified RNAgrail models on the Stanford RNA 3D Folding Kaggle challenge dataset. Following the competition protocol, for each target RNA sequence in the test set, we will generate five distinct 3D structure predictions. The quality of each prediction will be assessed using the Template Modeling Score (TM-score), as defined in Section 3.2. The final performance score for each target will be the average of the highest TM-scores (best-of-5) achieved across these five predictions.

To ensure a fair comparison, all baseline models will be evaluated on the exact same data splits provided by the Kaggle challenge. This consistent evaluation protocol will allow for a direct and meaningful comparison of the different approaches.

Furthermore, we will analyze the computational efficiency of our methods by measuring:

- **Average inference time:** The time taken to generate a single 3D structure prediction for a given RNA sequence. This will be compared against the inference time of the original RNAgrail model and the "randomly angle" baseline.

- **Training time (per epoch):** The time required to complete one full training epoch on a representative subset of the training data for both our modified models and the original RNAgrail model (if retraining is necessary). This will provide insight into the computational cost of training.

7 Baselines

To evaluate the performance and efficiency of our implementation with existing approaches, we will consider the following baselines:

- **Performance of the original RNAGrail model:** We will compare the performance of our model with the performance of the original RNAGrail [9] on the Stanford RNA 3D Folding Challenge [7] as a primary benchmark for prediction accuracy.
- **Computational resources for comparable performance:** We will assess the total compute resources required to achieve a similar level of training and evaluation performance as the original *RNAgrail* model on equivalent hardware on a subset of the training dataset.
- **Average inference time:** We will measure and compare the average inference time of our adapted model against the original *RNAgrail* model on equivalent hardware.
- **Kaggle Challenge leaderboard ranking:** We will track the rank of our submissions on the Stanford RNA 3D Folding Kaggle Challenge leaderboard. This will provide a real-world, competitive assessment of our model's performance against other submitted approaches.
- **Statistical Baseline (random angle)** We will implement a simple baseline that generates RNA 3D structures by placing nucleotides based on random angles. This approach will provide a lower bound on expected performance and help to highlight the effectiveness of model. The "randomly angle" baseline will involve the following steps:
  1. **Sequential Placement:** Start with the first nucleotide at an arbitrary origin (0,0,0).
  2. **Random Angles and Fixed Bond Lengths:** For each subsequent nucleotide, its position will be determined by:
     - The position of the previous nucleotide.
     - A fixed, biologically plausible bond length to the previous nucleotide.
     - Randomly chosen bond angles and dihedral angles within a physically reasonable range.

This comprehensive set of baselines will allow us to thoroughly evaluate the performance of our proposed approach in terms of both accuracy and computational efficiency.

References

[1] Yang Zhang and Jeffrey Skolnick. "Scoring function for automated assessment of protein structure template quality". In: *Proteins: Structure, Function, and Bioinformatics* 57.4 (2004), pp. 702–710.

[2] John Jumper et al. "Highly accurate protein structure prediction with AlphaFold". In: *nature* 596.7873 (2021), pp. 583–589.

[3] Jiaming Song, Chenlin Meng, and Stefano Ermon. *Denoising Diffusion Implicit Models*. 2022. arXiv: 2010.02502 [cs.LG]. URL: https://arxiv.org/abs/2010.02502.

[4] Shuo Zhang, Yang Liu, and Lei Xie. "A universal framework for accurate and efficient geometric deep learning of molecular systems". In: *Scientific Reports* 13.1 (2023), p. 19171.

[5] Josh Abramson et al. "Accurate structure prediction of biomolecular interactions with AlphaFold 3". In: *Nature* 630 (May 2024), pp. 493–500. DOI: 10.1038/s41586-024-07487-w.

[6] Rafael Josip Penić et al. "Rinalmo: General-purpose rna language models can generalize well on structure prediction tasks". In: *arXiv preprint arXiv:2403.00043* (2024).

[7] Shujun He et al. *Stanford RNA 3D Folding*. https://kaggle.com/competitions/stanford-rna-3d-folding. Kaggle. 2025.

[8] Linqi Zhou, Stefano Ermon, and Jiaming Song. *Inductive Moment Matching*. 2025. arXiv: 2503.07565 [cs.LG]. URL: https://arxiv.org/abs/2503.07565.

[9] Marek Justyna et al. "RNAgrail: graph neural network and diffusion model for RNA 3D structure prediction". In: ().