# AutoRNA: RNA tertiary structure prediction using variational autoencoder.

M.A. Kazanskii[1], L. Uroshlev[1*], F. Zatylkin[1], I. Pospelova[1], O.Kantidze[1], Y.Gankin[1]

[1] Quantori LLC, Cambridge, MA, USA

*Corresponding author: leoniduroshlev@gmail.com, leonid.uroshlev@quanori.com

**Abstract**

Understanding the tertiary structure of RNA is essential for advancing therapeutic development and vaccine design. Traditional methods, such as dynamic simulations, have been employed to study RNA structure, but often struggle to capture the complex, non-linear relationships within RNA sequences. Many previous approaches have relied on simpler models, which limited their performance.The scarcity of data is a key challenge in predicting the tertiary structure of RNA. Therefore, herein we introduce a variational autoencoder (VAE), AutoRNA, that achieves a root mean square error (RMSE) of approximately 4.5 Å in predicting nucleotide positions.

## Introduction

Understanding the fundamental properties of RNA molecules[1], including their three-dimensional (3D) structure, is necessary to develop new RNA-based medications. Leveraging machine learning (ML) techniques to decipher protein folding can solve challenges in the RNA domain. AlphaFold2 [2] (and the more recent AlphaFold3 [3]) have demonstrated the power of ML methods for solving protein-folding problems. Nevertheless, adapting neural networks similar to AlphaFold for RNA could present challenges.

Although RNA folding problem may seem less complex compared to protein folding because of the smaller to a smaller variety of nucleotides compared to the amino acids in proteins, which reduces the number of viable options, the datasets available for training are much smaller for RNA structures than for proteins structures [4]. This limitation may hinder the effectiveness of alignment-based ML approaches [2] when applied to RNA. Indeed, the Protein Data Bank currently contains about 4500 RNA structures and over 200,000 protein structures. Therefore, the applicability of ML methods based on the coevolution of monomers (e.g. AlphaFold3[3]), for RNA cases may be limited.

Studies have also indicated that the energy landscape of 3D RNA structures could be more intricate than that of proteins [5], with RNA potentially exhibiting a higher number of local energy minima. Our understanding of nucleotide coevolution in RNA is also limited, mainly because RNA encompasses distinct classes, such as tRNA, miRNA, and siRNA, each with its own evolutionary path. Moreover, the way in which RNA evolved for prokaryotes, eukaryotes, and viruses differed because of the diverse mechanisms of regulation and translation.

Hence, for predicting 3D RNA structures, it is necessary to employ ML techniques that do not solely rely on alignment or empirical knowledge of nucleotide coevolution. The precise determination of a

3D structure requires ML methods that either do not use alignment and coevolutionary data or utilize such information carefully.

Here, we present a novel, relatively simple architecture based on variational autoencoder (VAE) for predicting the 3D structure of RNA sequences. A simple generative model is justified because the dataset is small. Our approach achieved an average root mean squared error (RMSE) of approximately 4.5Å.

## Related Work

Numerous algorithms have been developed for predicting the 3D structure of RNA; however, the accuracy of these predictions remains moderate. Several attempts have been made to generalize methods and algorithms for predicting the 3D structure of RNA, starting from the prediction of the secondary structure. In [6], the authors applied dynamic programming methods for predicting the 3D of RNA. Another example is iFoldRNA [7], which employs a simplified coarse-grained representation and Monte Carlo sampling to simulate RNA folding and predict tertiary structures. This approach demonstrated that simplified energy landscapes could reproduce near-native conformations for short RNA sequences.

The most common techniques are based on the empirical potential method and its variants, which calculate the interaction forces between RNA atoms by examining interatomic contacts [8,9]. This approach is also used for protein-protein or protein-ligand interaction modeling. Notable empirical potentials include AMBER [10] and CHARMM [11], which are integrated into molecular dynamics software. However, employing molecular dynamics directly for modeling is a time-consuming process. The complete folding of an RNA molecule requires $10^{-6}$ seconds for forming the 3D backbone structure and $10^{-4}$ seconds for folding each nucleotide in the molecule [12]. Therefore, the folding time could be 0.01-0.1 seconds for a molecule containing several tens of nucleotides, making the process computationally expensive. However, coarse-grained potentials [13] may reduce complexity and speed up computations. Hybrid methods have become increasingly popular in recent years. These methods combine empirical potentials with various neural network structures. For instance, one study applied the transformer architecture in conjunction with Rosetta's empirical potentials [14]. An alternative approach derived from protein design involves the assembly of structures using elementary templates. Another technique involves breaking down structures into a collection of 3D templates and then optimizing these through minimization in the final phase of the algorithm. In [15] the authors describe a web server that decomposes structures into a set of 3D templates, followed by minimization in the final stage of the algorithm. In addition, various ML approaches can be used to predict the structure. For example in [16], the authors present a modified U-Net neural network that predicts a distance map for the 3D RNA structure.

Notwithstanding the diversity of RNA folding algorithms and the success of hybrid and deep learning -based methods, most approaches still rely heavily on extensive empirical potentials, handcrafted features, or large-scale coevolutionary datasets, which are often unavailable for RNA. Moreover, many models are computationally expensive and difficult to generalize. Therefore, a lightweight, data-efficient model capable of capturing the essential 3D geometry of RNA molecules without relying on sequence alignment or predefined templates is still required. Our proposed method

addresses this gap by leveraging a VAE trained directly on RNA distance matrices to reconstruct tertiary structures efficiently.

## Methodology

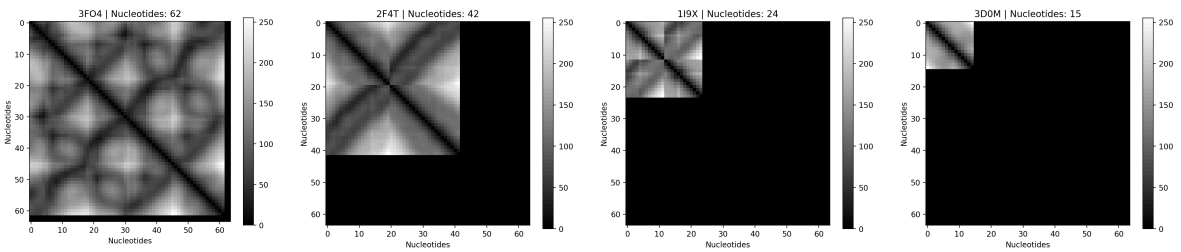
### Data preprocessing and experiment

We used a PDB dataset, containing the 3D structural information of RNA molecules [17]. We selected PDB files that contained only the RNA structure without complexes with protein or DNA. We processed the data in several steps: we removed the duplicates and the empty sequences, dropped molecules from the chain, and calculated the spatial positions of the backbone of each nucleotide. We used the backbone coordinates as a tertiary structure for the RNA sequence.

For the main experiment, we selected only RNA sequences that were 64 nucleotide bases or shorter. We generated a distance matrix to represent the distances between each nucleotide backbone. For nucleotide sequences with fewer than 64 nucleotides, we applied padding with a value of 0. We applied distance clipping at 100 Å to handle a very small number of extreme outliers. Quantitatively, the majority of pairwise distances fell within the range 0–80 Å, and only 0.06% of all distances exceeded 100 Å in the dataset with sequences shorter than 64 nucleotides. Therefore, clipping affected only a negligible fraction of samples and is unlikely to introduce systematic bias, while helping stabilize the model by preventing rare extreme values from disproportionately influencing training. Therefore, fixing distances exceeding 100 to 100 did not substantially affect the algorithm's performance. Figure 1 presents examples of the distance matrices which were normalized in the range of [0, 1] for input into the VAE. The values of the normalized distance matrices were determined as:

$$d_{mnorm} = \frac{d_m}{100}$$

Figure 1. Examples of the distance matrices. Each image is equivalent to the distance matrix corresponding to the RNA sequence with zero padding. The images were scaled by a factor of 255 for better visualization: the original images have pixel values in the range of (0, 1). The nucleotides are represented on the *x* and *y* axes. The sequences are not included. The black area corresponds to the padding (to reach 64 nucleotides).




Together with the distance matrices of the RNA sequences, we used nucleotide letters A, G, C, and U) as one-hot encoder vectors with the following rules: A = (1, 0, 0, 0), G = (0, 1, 0, 0), C = (0, 0, 1, 0), U = (0, 0, 0, 1). We added padding with values (0, 0, 0, 0) to the sequences of length less than 64.

To avoid overfitting the sequences with similar homology, we performed homology filtering and clustering based on homology, as described in [18]. However, we used the homology between the

sequences as a differentiator between different clusters. According to [19], homology is one of the options to group sequences into structural and functional clusters. We calculated the homology matrix for each pair of RNA sequences. Then, we obtained the similarity matrix by subtracting the homology matrix from the identity matrix. If the element of the similarity matrix is close to zero, the RNA sequences corresponding to that element are homologically similar. We used the hierarchical clustering algorithm to cluster the RNA sequences because it allowed fitting to different shapes[20], which is crucial for datasets with various homologies. A threshold of 60 was chosen as the clustering cutoff described in [21]. This procedure was in accordance with [22] which mentioned an interval of 60 - 70. Additionally, we analyzed the number of resulting clusters. There were 100 clusters chosen with a threshold of 60. However, there were only 16 clusters with threshold 70, an insufficient number of further splitting the data. The largest three clusters contained 111, 77, and 74 sequences. Figure 2 illustrates the clustering of the RNA sequences. We used the PCA algorithm to calculate the first two components and visualize the plot. Different colours correspond to different clusters found with the hierarchical clustering algorithm.

We divided the dataset into training, validation, and testing subsets with approximate proportions of 80%, 15%, and 5%, respectively. A total of 842 sequences were obtained. After partition, there were 719, 96, and 27 RNA sequences in the training, validation, and testing subsets, respectively. Figure 3 presents the distribution of the lengths of the RNA sequences in the training, validation, and testing subsets and the total dataset. The lengths of the distributions were similar across subsets.

Figure 2. A graph of the first two principal components of the similarity matrix. Different colors represent distinct clusters obtained using hierarchical clustering.

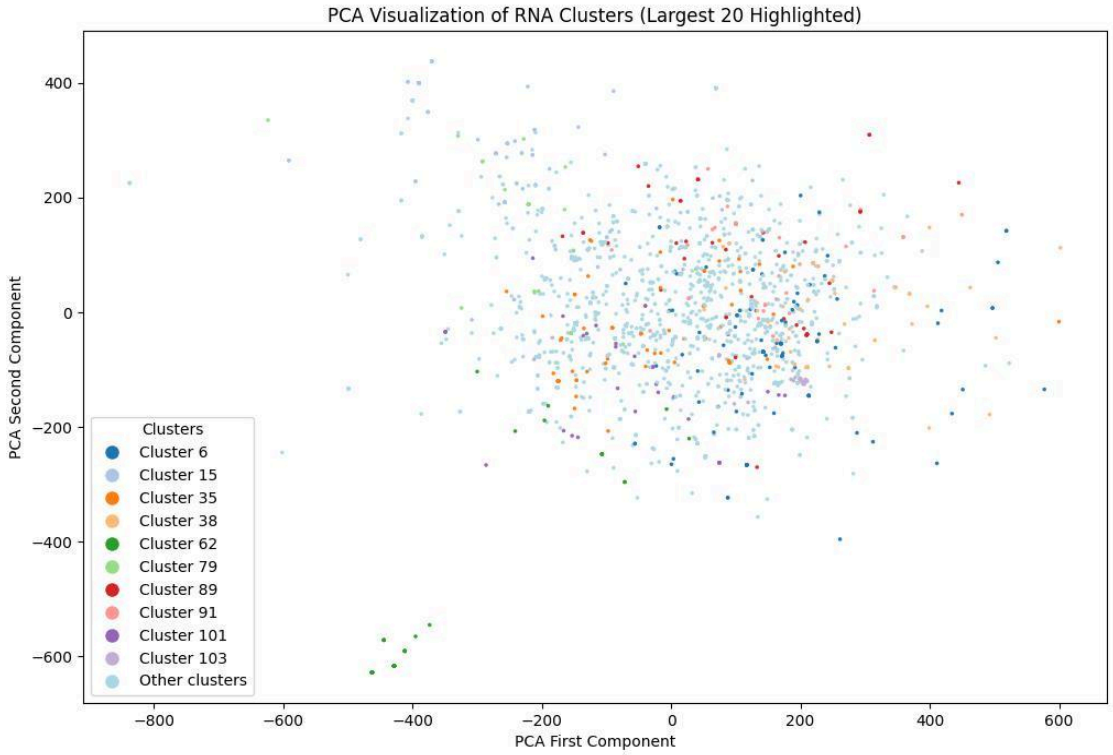
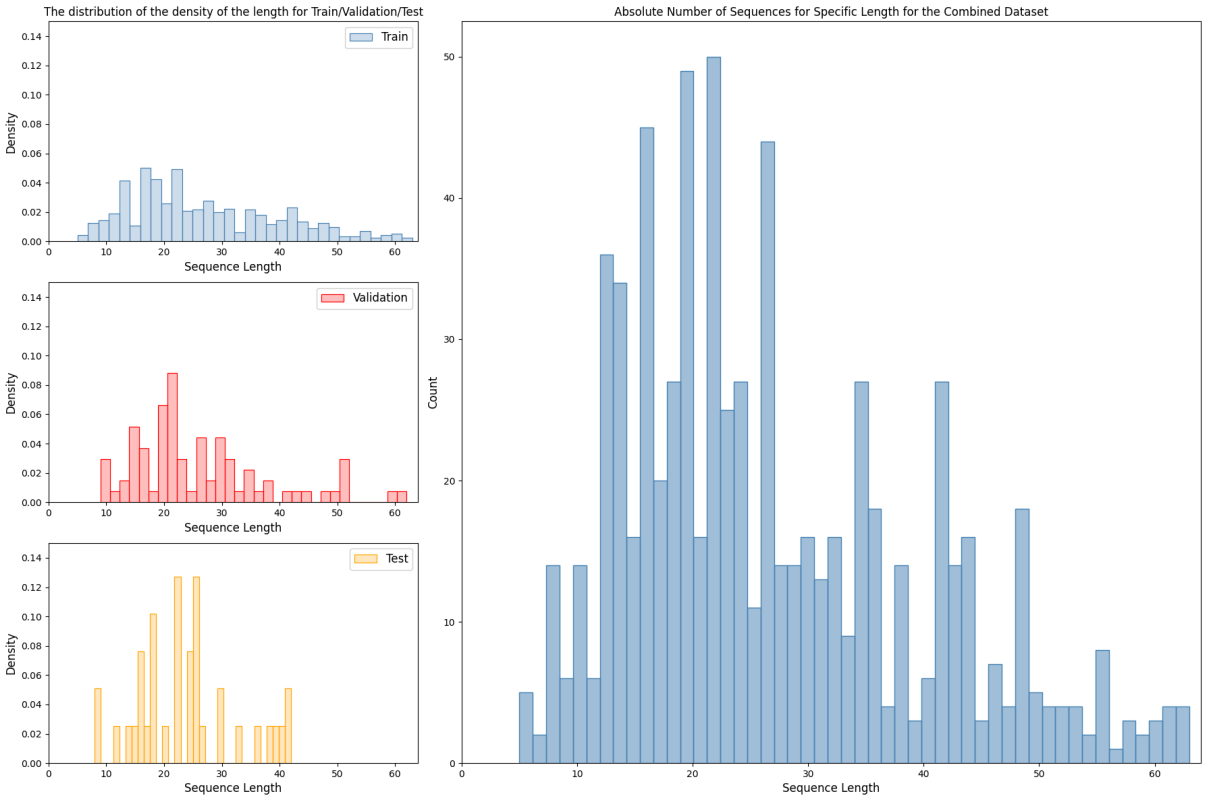


Figure 3. The probability density function of sequence lengths for the training, validation, and testing subsets (left) and the distribution of the sequence lengths in the combined dataset (right).




## Pipeline

We used the VAE [23] to predict the distance matrix. In particular, we concatenated the conditions vector with the random vector as input. The condition vector is the flattened one-hot encoded nucleotide sequence (size = 256 nodes). The random greyscale image represents a noise vector of the same size as the distance matrix (64 × 64 = 4096 nodes). We used a traditional VAE that consisted of an encoder, bottleneck, and decoder. The encoder consisted of two fully connected layers with 4096 and 2048 nodes. The bottleneck layer was trivial and consisted of only one layer with 1024 nodes. The decoder comprised three fully connected layers with 2048, 4096, and 4096 nodes. The last layer represented the values of the distance matrix. After each layer was fully connected (excluding the last one), we used batch normalization to enhance performance and stability. We also applied dropout at a rate of 0.5 for the output signals from the batch-normalization layers. The total number of trainable parameters in the model equaled 64,532,480.

Our choice of a fully connected VAE was motivated by the characteristics of the representation and the study objective. First, the input to the model is a global pairwise distance matrix, where every element depends on every other nucleotide; therefore, using dense layers ensures that no connectivity assumptions are imposed a priori, allowing the model to learn non-local tertiary dependencies directly from the data instead of constraining it to local receptive fields as in CNNs.

The model's loss followed the standard formulation for variational autoencoders and consisted of two terms: the reconstruction loss and the Kullback–Leibler (KL) divergence [24]. The reconstruction term was computed using the mean absolute error (MAE), chosen for its robustness to outliers

(compared to MSE - mean squared error). A weighting coefficient β was used to balance the reconstruction and regularization contributions. The total loss is

$$\mathcal{L}_{\mathrm{VAE}} = \mathcal{L}_{\mathrm{RECON}} + \beta \mathcal{L}_{\mathrm{KL}}$$

where the reconstruction loss is defined as

$$\mathcal{L}_{\mathrm{MAE}} = \frac{1}{N} \sum_{i=1}^{N} |\mathrm{dm}_{\mathrm{pred},i} - \mathrm{dm}_{\mathrm{true},i}|$$

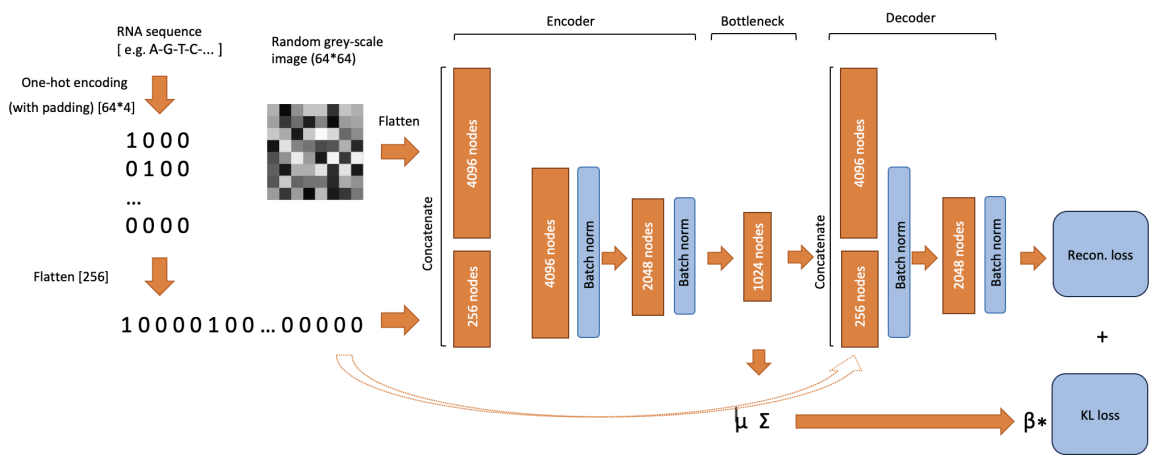
and the KL divergence term is

$$\mathcal{L}_{\mathrm{KL}} = -\frac{1}{2} \sum_{j=1}^{d} \left(1 + \log \sigma_j^2 - \mu_j^2 - \sigma_j^2\right)$$

In these expressions,  denotes the iii-th input sample and  its reconstructed counterpart produced by the decoder, while $N$ represents the number of samples and $d$ is the dimensionality of the latent space. Here, $\mu_j$ determines the expected location of the latent variable, while $\sigma_j$ characterizes the uncertainty of this representation. The parameter β is a scaling (regularization) coefficient that controls the relative influence of the KL divergence term: when $\beta = 1$ the loss corresponds to the classical VAE formulation, whereas $\beta > 1$ enforces stronger regularization and encourages a more structured and disentangled latent space.

We used the MAE as a metric to represent the algorithm's performance. Note that the MAE was calculated only for significant values (the padding values were skipped while calculating the MAE and loss function). Figure 4 displays the workflow of the ML part of the experiment. The architecture was chosen after fine-tuning the model carefully.

Figure 4. Schematic representation of the variational autoencoder (VAE). The RNA sequences are one-hot encoded into a numeric representation and input together with the random image into the VAE. The VAE consists of five hidden layers: the encoder consists of two hidden layers (4096 and 2048 nodes), the bottleneck consists of one layer (1024 nodes), and the decoder consists of two layers (4096+256 and 2048 nodes). The loss function is a combination of the reconstruction loss based on the root mean square error (RMSE) and the Kullback–Leibler (KL) loss.



It is worth mentioning that the distance matrices are supposed to be symmetric. We perform the symmetrization of the predicted distance matrices as a part of postprocessing as follows:

$$dm_{\text{sym}} = \frac{dm + dm^{\text{T}}}{2}$$

If symmetrization is performed before calculating the loss function, training becomes unstable because a set of values for the predicted distance matrix could satisfy the same value for the symmetric distance matrix. Hence, the sum would matter for the specific element of the symmetric matrix, making the values more flexible by adding an extra degree of freedom, and resulting in unstable training. Therefore, symmetrization was performed after the training loop as a postprocessing technique for the training subset.

After we recover three-dimensional points coordinates by multidimensional scaling algorithm (MDS). This is a classical algorithm, based on spectral decomposition of distance matrix. Moreover, this algorithm allows working with noisy distance matrices using nonlinear optimization methods. Since our distance matrix is precisely that, we decided to use this algorithm.

The full inference pipeline comprised the following steps:

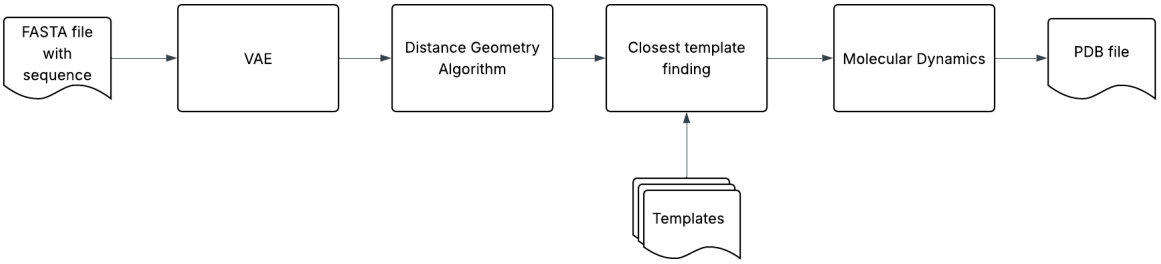
1. VAE center of mass prediction.
2. Structure assembly from templates.
3. Correcting the obtained structure by adding atoms.
4. Molecular dynamics for polishing the structure and removing clashes between the atoms.

After receiving the predictions of the center of mass, the rest of the RNA structure should be restored. For this, we used a base of templates from the PDB. We extract from PDB structures for all possible fourths of nucleotides. After obtaining the distance matrix from our VAE, we used distance geometries to convert the matrix into 3D coordinates of RNA molecules. We used the library's most suitable templates and assembly structure. Note that the library was assembled from the RNA data in the PDB. For each RNA molecule in the PDB, all possible nucleotide quadruplets were selected and put into their respective catalogs, resulting in 256 catalogs comprising 46,199 files. For each nucleotide in each template, we have computed the centres of mass and align obtained centres of mass of template and prediction into our results. The most optimal template is selected based on the RMSD between the predicted points and the template atoms. The template is selected for each quartet of nucleotides with a step size of one nucleotide. At the same time, overlapping regions from neighboring templates are averaged. No additional features, such as torsion angles or covalent bond lengths, were taken into account. This approach helped to reduce the number of parameters in the VAE.

In the final stage, we corrected the covalent bond and flat angles that arise in the resulting RNA structure. For this, we performed molecular dynamics computations and optimized the potential energy received by the RNA molecules. We used the OpenMM package as a molecular dynamics engine [26]. The AlphaFold2 pipeline implemented a similar solution for structure refinement as the final stage [2]. Molecular dynamics was run for 1M steps by solving Langevin dynamics equations. We used the AMBER14 force field as an atom force field. All computations were performed under

standard conditions (300 K, pH 7, 1 atm). Modeling was performed in a water cube with a 3-tip water model and 2 nm padding from RNA molecules. RNA molecules were protonated using the pdb4amber software, and missing atoms in some templates were filled using the same software.

Figure 5 illustrates the full AutoRNA reconstruction workflow. Starting from the FASTA sequence, the VAE predicts structural priors that are converted into spatial constraints and transformed into a coarse 3D model via distance geometry, which is then matched to the closest structural template. The final structure is refined using molecular dynamics to obtain a physically realistic RNA model, resulting in the final PDB structure.




## Results

For the experiment, we trained our model for 2000 epochs. We used the validation subset for fine-tuning the parameters, and the testing only for assessing the final prediction. We stored the model with the best validation MAE. For training, the batch size was set to 16, which was chosen to obtain a stable solution while achieving a sufficiently fast learning rate. However, we did not observe a significant correlation between batch size and final performance. We also used the Adam optimizer [27]. The minimum and maximum lengths of the RNA sequences were 4 and 64 nucleotides, respectively. We used the best-performing loss parameter, which balances the KL loss and reconstruction loss, namely that which equals 1.1. The training time for 1000 epochs was approximately 2.1 hours (Apple M1 processor). The inference time for one RNA sequence during batch execution was approximately 0.001 seconds.

In the inference stage, we also assembled the final PDB file containing the final structure from a set of templates. The most suitable template was selected based on its proximity to the predicted points. Next, we optimized the assembled structure using molecular dynamics methods. As our experiments show, the centers of mass of the structure obtained in this way differ from the points obtained from VAE by no more than 1 Å; thus, they do not affect the final results.

In addition to RMSE and MAE, we calculated the global distance test (GDT) and template modeling (TM) score after symmetrization. We calculated the GDT score based on the distances between the corresponding atoms (in our case, centers of masses) in the predicted and reference structures, evaluated at multiple distance thresholds. We used the values of 1, 2, 4, and 8 Å, according to [28]. We calculated the TM score based on [29] using the tmscoring Python library [30]. We included the GDT and TM scores in the final computation due to the limitations of the RMSE (and MAE) metric for different nucleotide lengths.

Table 1 presents the results of the experiment for the training, validation, and testing subsets. The results (RMSE, MAE, RMSE + sym, and MAE + sym) were slightly better for the testing subset than

for the validation subset due to the stochasticity of the values. Interestingly, the GDT score was significantly lower for the testing subset than for the training and validation subsets. For the testing subset, we achieved an RMSE of 4.49Å and an MAE of 3.13 Å. Overfitting was noticeable, resulting from the differences in performance in the validation and training subsets. Therefore, having additional data would enhance the performance of our algorithm. Note that we also included the performance results after symmetrization during postprocessing. However, symmetrization of the predicted distance matrix resulted in negligible performance improvement. Indeed, the RMSE and MAE were only slightly higher with symmetrization than without symmetrization. For the testing subset, the RMSE was 4.49 Å before symmetrization and 4.48 Å after symmetrization; thus, symmetrization had a minimal effect. Any slight reduction in RMSE after symmetrization is a mathematical consequence of averaging rather than an intrinsic improvement of the predictive model. Our goal in reporting the symmetrized scores is only to demonstrate that the errors are largely symmetric noise and that the predicted matrices remain internally consistent. It is worth mentioning that the test set was small (37 sequences), and the result that the performance on the test set is slightly better than on validation is possible due to high variance on the test set.

Table 1. The results of the variational autoencoder (VAE) experiment for the training, validation, and testing subsets. The metrics shown are the mean average error (MAE), root mean square error (RMSE), MAE after symmetrization, RMSE after symmetrization, TM-score after symmetrization, and GDT-score after symmetrization. All values are presented as the mean ± standard deviation.

| | **MAE** | **MAE+sym** | **RMSE** | **RMSE+sym** | **GDT-score** | **TM-score** |
|---|---|---|---|---|---|---|
| **Train** | 3.23 ±1.64 | 3.22 ±1.64 | 4.69 ±2.46 | 4.69 ±2.46 | 39.67±20.58 | 0.25±0.14 |
| **Validation** | 3.86 ±2.61 | 3.86 ±2.61 | 5.56 ±3.77 | 5.56 ±3.77 | 26.83±20.87 | 0.18±00,5 |
| **Test** | 3.13±1.69 | 3.12±1.69 | 4.49±2.42 | 4.48±2.43 | 32.13±20.19 | 0.19±0.07 |

Figure 6 presents the computational results for two RNA sequences. Within each trio of images, the first two represent the actual and predicted distance matrices, respectively. The third image depicts the 3D structure of the RNA molecule. We employed the distance geometry method to reconstruct the coordinates from the distance matrix, a classic technique proven effective in nuclear magnetic resonance spectroscopy [31]. The challenge is to identify a set of coordinates that optimally fits the given distance constraints. According to [32], this problem can be approached by reformulating it as the task of finding a matrix:

$$\min_{C} \|C - G^2\|, \quad \text{where} \quad G = -\tfrac{1}{2} H^{\mathrm{T}} D H$$

In this formulation, H is the centering matrix, which removes the translational component of the configuration; $D$ denotes the pairwise Euclidean distance matrix of $N * N$, whose elements correspond to the squared distances; $G$ is the Gram matrix (inner-product matrix); $C$ is the matrix of $3D$ Cartesian coordinates to be recovered.

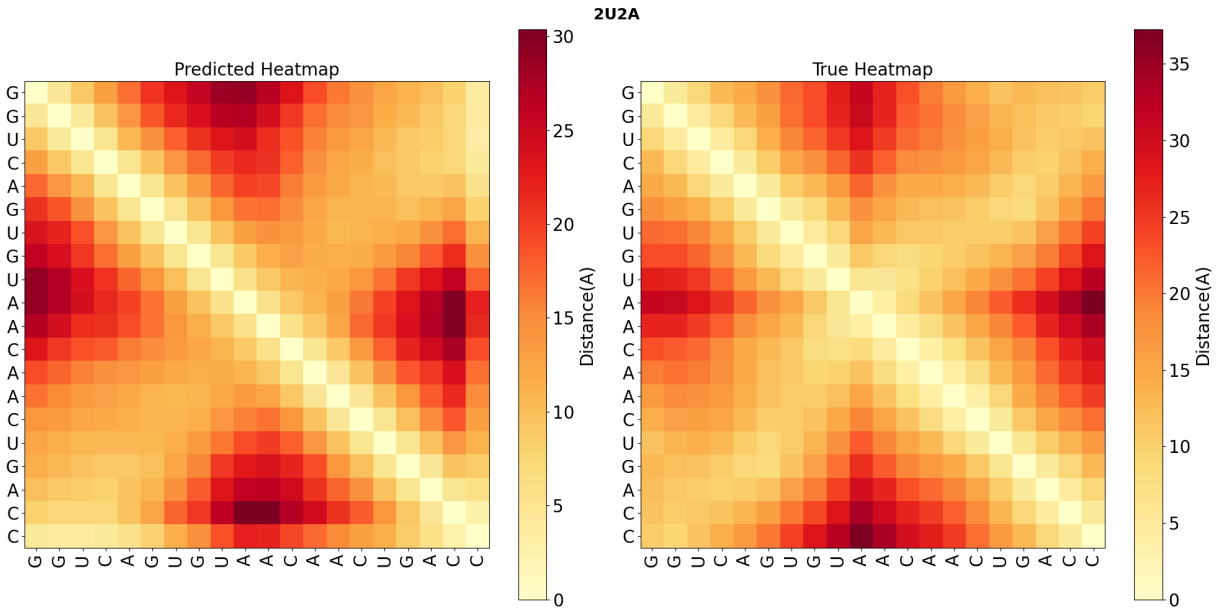
This problem could be solved via the computation of the spectral decomposition of distance matrices and computation of eigenvalues and eigenvectors. The components of this matrix are the coordinates we need. In addition to the basic experiments, we checked the variational concept of the VAE. We generated the distance matrices five times for several RNA sequences with different noise vectors, and the results are presented in Figure 8. The change in noise vectors did not substantially alter the

predicted distance matrix. Thus, the variational factor of the algorithm had a minimal impact on the outcome. If we calculate the average coefficient of variation as a ratio between the standard deviation of the error for the different noise vectors for all the test sequences, the resulting ratio is small:

$$\mathrm{cv} = \frac{\sigma}{\mu} = 0.13$$

It is important to note that applying distance geometry methods requires the distance matrix to be Euclidean—that is, it must satisfy symmetry, the triangle inequality, and have a rank not exceeding three. To assess this property, we computed the Negative Eigen-Fraction (NEF) for the distance matrices produced by the VAE, as NEF is commonly used to quantify how closely a matrix adheres to Euclidean structure. On the test set, the average NEF was 0.092, suggesting the presence of some noise while still indicating that the matrices generally satisfy the Euclidean criteria.

Figure 6. Distance matrices for three RNA sequences (actual and predicted). The image on the left represents the predicted heatmap, and the image on the right represents the actual heatmap. The presented sequences are 2U2A, 3DW7 (from top to bottom).



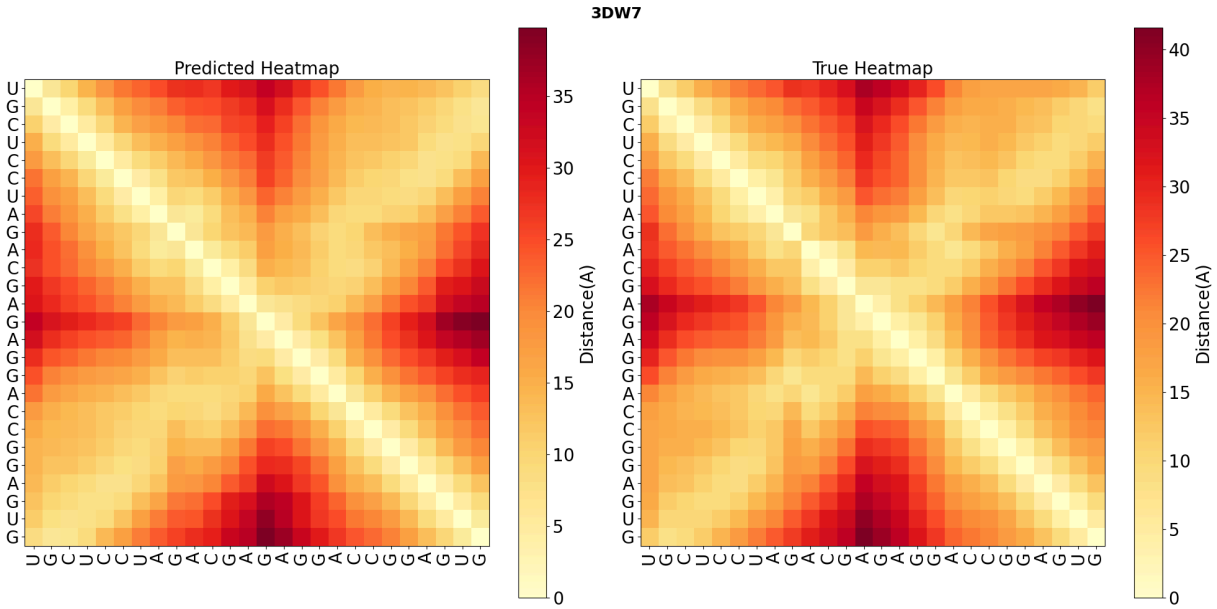
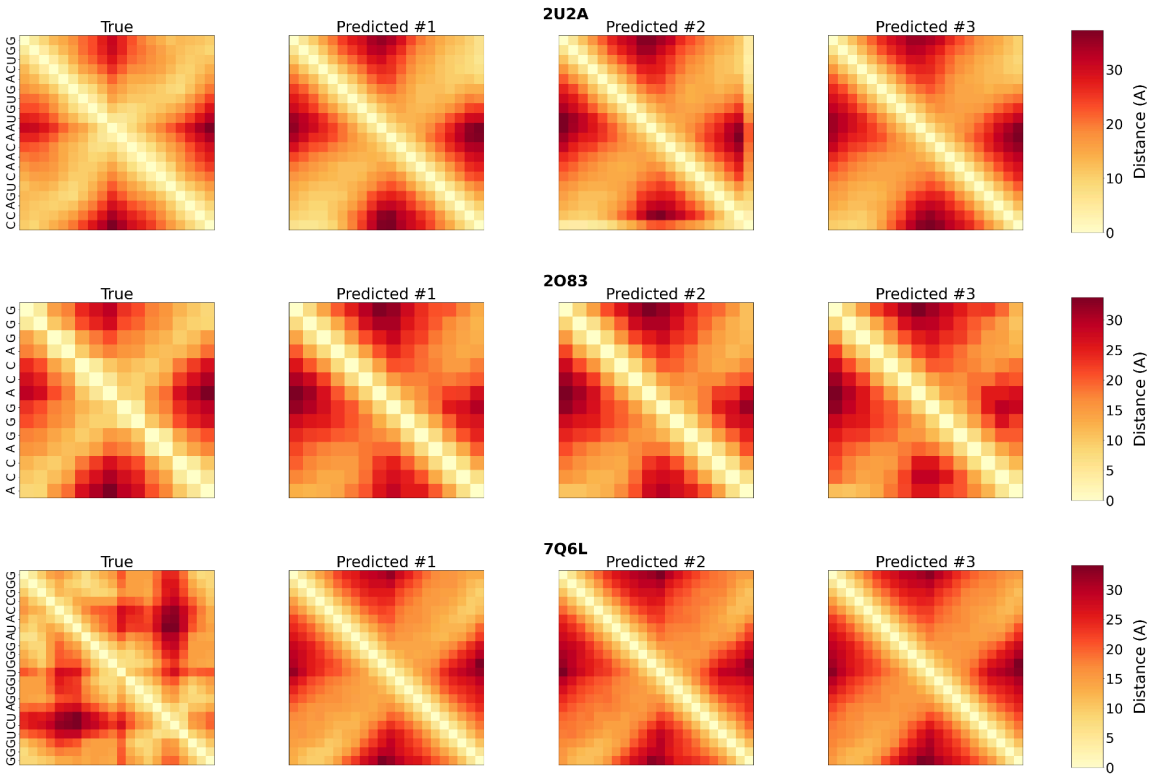



Figure 7. Structures generated with different noise vectors (three noise vectors for each sequence) for three RNA structures (from top to bottom : 2U2A, 2O83, 7Q6L). The image on the left represents the true distance matrices, and the images on the right represent the predicted structures with different initial noise vectors. The nucleotide labels corresponding to the sequences are shown on the left.




In the next experiment, we trained the same model but for longer RNA sequences, fixing the maximum length at 128 nucleotides while keeping the other parameters unchanged. The total dataset

in this case consisted of 925 sequences, compared to 842 sequences containing fewer than 64 nucleotides. As the training, validation, and testing subsets naturally differed from those when the maximum length was fixed at 64 nucleotides, we compared the results with those in Table 1, considering the stochastic component. Table 2 presents the results of the experiments in the testing subset. We believe that the decrease in accuracy for longer sequences arises primarily from data scarcity rather than from inherent architectural limitations of the model. To assess overfitting quantitatively, we compared performance across the training, validation, and held-out test subsets . The validation MAE (3.86 Å) and RMSE (5.56 Å) were moderately higher than the corresponding training values (3.23 Å and 4.69 Å), representing an approximate 18–20% increase in error on unseen data. Structural quality metrics exhibited a similar trend: the mean GDT score decreased from 39.7 to 26.8 and the TM-score from 0.25 to 0.18 between training and validation subsets, indicating reduced global structural alignment. Importantly, the test subset yielded performance comparable to the validation subset (MAE = 3.13 Å, RMSE = 4.49 Å), suggesting that the model generalizes reasonably well to unseen data despite moderate overfitting.

Table 2. Comparison of the performance of the AutoRNA algorithm for small and medium-sized RNA sequences in the testing subset.

| | **Test dataset** | | |
|---|---|---|---|
| | **Number of sequences** | **RMSE** | **GDT-score** |
| **Sequences, less than 64 nts** | 30 | 4.49 ± 2.35 | 32.13 ± 20.19 |
| **Sequences, less than 128 nts and more than 64 nts** | 7 | 6.05 ± 1.17 | 14.72 ± 4.33 |

Table 3. The comparison of the algorithms (RhoFold, SimRNA) with our algorithm (AutoRNA) on the test dataset (for different sizes of the sequences). The RMSE score is displayed.

| | **Our algorithm (RMSE)** | **RhoFold (RMSE)** | **SimRNA (RMSE)** |
|---|---|---|---|
| **MAE (Å), less than 32 nts** | 3.475 | 3.414 | 3.990 |
| **MAE (Å), over 32 nts** | 4.248 | 3.692 | 5.125 |

We then compared our AutoRNA algorithm with several state-of-the-art models for 3D RNA structure prediction: AlphaFold3 [3], Boltz2 [33], SimRNA, and RhoFold [34]. These models were selected because they are the most well-known and frequently cited. RhoFold was chosen because it combines physical and ML approaches. For simulations with RhoFold, we used standard parameter settings (10000 iterations).

It is also worth noting that not all methods use molecular dynamics to refine results. Although this approach is common in protein and RNA folding and is used we performed calculations for AutoRNA with and without molecular dynamics. Methods such as AlphaFold3 or RhoFold do not use molecular dynamics or any physical-based methods at any stage. However, other methods, such as Boltz-2 or SimRNA, use either scoring with physical potentials or methods similar to MD.

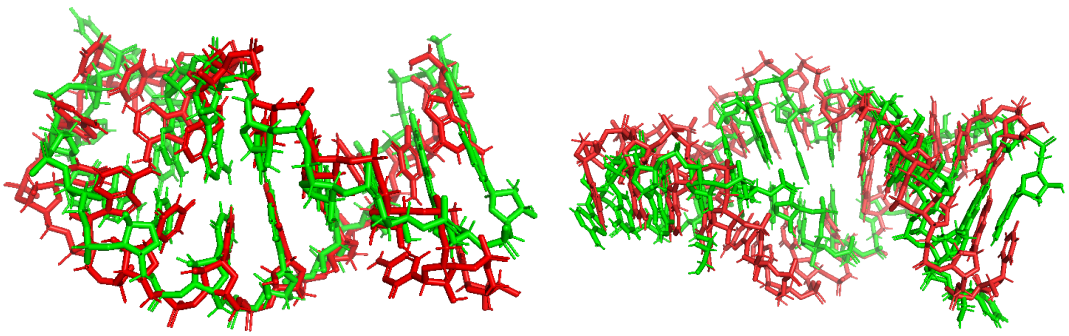
For modeling with AlphaFold3 and Boltz2, we used the AF3 server from Google and the Boltz2-NIM server from Nvidia, respectively. We selected structures with fewer than 64 nucleotides and only one chain from those provided for the RNA-Puzzle competition [4], identifying nine such structures in the entire dataset. Only one RNA (puzzle 16) in our dataset was shorter than 32 nucleotides. The RNA

sequences had different origins: they were catalytic RNAs (puzzle 16), riboswitches (puzzles 29 and 38), and aptamers (puzzle 32). The root mean square deviation (RMSD) metric was used for comparison. We used the PyMol [35] package to calculate the RMSD between the experimental and modeled RNA structures, and the results for each structure are presented in Table 4. The PDB files with the structures and the FASTA file with the sequences are provided in the Supplementary Materials.

Table 4. The comparison of the algorithms (AlphaFold3, Boltz2, RhoFold, SimRNA) with our algorithm (AutoRNA) on the subset of RNA-puzzle dataset. Each line corresponds to the individual sequence. The RMSE score is displayed.

| **Sequence ID** | **AlphaFold3(A)** | **Boltz2 (A)** | **RhoFold(A)** | **SimRNA(A)** | **AutoRNA with MD (A)** | **AutoRNA without MD (A)** |
|---|---|---|---|---|---|---|
| rep16 | 0.659 | 0.318 | 1.901 | 3.618 | 1.671 | 1.576 |
| rep19 | 5.095 | 3.415 | 12.791 | 15.090 | 7.512 | 10.755 |
| rep23 | 9.64 | 0.535 | 2.811 | 9.353 | 6.404 | 6.577 |
| rep29 | 3.625 | 0.486 | 17.089 | 3.233 | 2.202 | 2.211 |
| rep32 | 9.553 | 1.063 | 2.024 | 4.135 | 4.026 | 5.146 |
| rep38 | 10.08 | 2.202 | 8.987 | 9.830 | 3.404 | 3.958 |
| repRP11 | 3.832 | 6.315 | 5.562 | 8.537 | 3.587 | 4.755 |
| repRP14 | 1.017 | 1.302 | 15.302 | 21.930 | 7.302 | 11.589 |
| repRP17 | 3.579 | 2.524 | 6.532 | 26.337 | 4.191 | 17.120 |
| **Average RMSE** | **5.231 ± 3.864** | **2.018 ± 2.045** | **8.111 ± 6.151** | **11.34 ± 8.235** | **4.478 ± 2.271** | **7.076±5.096** |

Figure 8. Comparison of the experimental and predicted structures for RNA stem-loops 6XXB (left) and 7L3R (right). The actual structure is shown in green, and the predicted structure is shown in red.



The presented results demonstrate that incorporating MD leads to improved structural predictions. While the enhancement is modest in most cases, three previously poorly predicted structures show a substantial improvement of approximately 3–4 Å. This effect is likely due to the fact that, when the initial prediction is accurate, nucleotides are already positioned near local energy minima, so molecular dynamics only slightly adjusts their configuration. In contrast, for poorly predicted structures, MD induces more pronounced rearrangements of both nucleotides and the RNA backbone, resulting in a noticeably better final structure. Although the positions of individual nucleotides may vary markedly, the backbone positions almost always coincide with sufficient accuracy. Figure 8 shows that the backbone positions in the modeled structures almost completely align with those in the experimental structures.

## Discussion

The proposed AutoRNA algorithm has several shortcomings despite its simplicity and robust performance. The main drawback is that the predicted and learning set distance matrices have a fixed and limited size. Thus, the PDB dataset contains insufficient long-RNA structures for proper training.

Although the current performance of the AutoRNA algorithm (average RMSE ≈ 4.5 Å, TM score < 0.25) does not achieve near-native accuracy, it is sufficient for capturing coarse-grained structural features. Importantly, TM-scores below 0.25 and GDT values around 30 are not indicative of correct tertiary RNA structures, and thus the predictions should not be interpreted as near-native models. Instead, AutoRNA is best suited for exploratory structural analyses, generating functional hypotheses, and pre-screening candidate conformations.

Larger RNAs (>64 nts) were briefly tested, but the accuracy of the AutoRNA algorithm decreased substantially (RMSE > 6 Å, GDT score < 15), indicating limited generalization to longer sequences. Thus, we focused the primary experiment on RNAs ≤64 nts because the dataset was heavily skewed toward short sequences (e.g., miRNAs and piRNAs), which provide more reliable statistics and simpler structural contexts. An additional experiment on RNA sequences containing 65–128 nucleotides served as a generalization test, showing the expected decline in performance due to data sparsity and longer-range dependencies.

The architecture of the VAE is one of the possible approaches for resolving RNA folding. Additionally, incorporating RNA-specific features into the model based on the biophysical properties of RNA molecules could be attempted. For example, we could consider the secondary structure of RNA [36] or the functional class of predicted RNA molecules or features obtained from RNA templates as additional features.

The idea of extending the developed method to other computational biology problems is also quite promising. For example, investigating the interactions between RNA-binding proteins (RBP) and RNA [37, 38] at the level of structural patterns. A combined approach involving the use of tools based on different computational approaches can also be quite interesting [39]. For example, in the task of predicting the affinity of RNA binding to a protein, one can try to build an ensemble of sequence-based models, physics-based models [40], and structural ML models [41].

Another extremely important task is predicting the structure of long non-coding RNAs [42,43,38]. The role of these RNAs is not fully understood, but the correlation between the expression of such RNAs and cancer is widely known [44, 45]. Understanding the three-dimensional structure of long non-coding RNA may be key to understanding its functions, just as structural annotation of proteins sheds light on how they work. In addition, the growth and development of tumors can be influenced by chemical modifications of various RNAs, so another direction could be to add information about nucleotide modifications to the model [46, 47, 48].

As shown for proteins [2, 4], model quality could be improved by incorporating evolutionary information. However, there is no adequate model of nucleotide coevolution for RNA molecules or the connection of coevolution to the RNA tertiary structure. Therefore, before adding coevolution information, we must understand the applicability of such a model to our problem. Moreover, we could use physically informed neural networks [37] to gain additional information about atomic interactions.

Another important direction is to enhance the dataset. Modeling new RNA structures using physics-based methods (e.g., molecular dynamics or Monte Carlo sampling) could yield good-quality structures but is likely to be slow and computationally expensive. Thus, using a rotamer library may be a practical solution, as it would allow changing one or several nucleotides in an RNA molecule into other nucleotides. Their local conformation must be in accordance with the rotamer library, after which several molecular dynamics steps could be performed to remove clashes and refine the structure.

Our AutoRNA algorithm generates RNA conformations that are broadly consistent with known structural principles of RNA folding. In particular, the predicted structures retain realistic geometric organization and recover characteristic structural motifs for several RNA classes. For example, most miRNA sequences folded by our approach form canonical hairpin-like architectures, while tRNA sequences adopt conformations compatible with their characteristic cloverleaf secondary structure organization. Overall, for RNA classes with well-defined structural motifs, AutoRNA is able to reproduce key topological features, suggesting that the model captures important aspects of RNA structural organization, even though the resulting models should not be interpreted as near-native tertiary structures.

## Conclusions

This study developed a novel algorithm for predicting the tertiary structure of RNA molecules based on a VAE architecture. Unlike traditional computational methods, such as molecular dynamics simulations [29], our AutoRNA algorithm leverages the complex, nonlinear relationships embedded in RNA sequences to generate accurate structural predictions. This data-driven approach enables the efficient modeling of RNA-folding mechanisms and offers a scalable alternative to physics-based simulations, which are often computationally expensive.

Understanding the 3D conformation of RNA molecules is vital for numerous biomedical applications, including the design of RNA-based therapeutics and the development of next-generation RNA vaccines. Our AutoRNA algorithm represents a promising step in this direction, achieving an RMSE

of approximately 4.5 Å when predicting the center of mass positions of individual nucleotides, highlighting the potential of deep generative models in structural bioinformatics.

Nonetheless, the AutoRNA algorithm has limitations regarding the specific RNA sequences because of the need for appropriate training data. For some RNA structures, sub-optimal conformations that could vary from the original tertiary structures were found. However, the proposed AutoRNA algorithm could be used to predict the RNA tertiary structure. A Python implementation of the AutoRNA algorithm is provided on GitHub: https://github.com/quantori/AutoRNA.

## Contributions

M.K. contributed to planning, discussion, pre-processing the dataset, creating models, creating visualisations, performing experiments, writing manuscript.

L.U. contributed to planning, discussion, pre-processing the dataset, reconstruction of a 3d structure, and writing the manuscript.

F. Z. contributed to pre-processing the dataset, discussion, creating visualization

I. P. contributed to discussion, creating visualizations

O. K. contributed to the discussion, writing a manuscript.

Y. G. contributed to planning, discussion, and organizing the research.

Acknowledgements: A. Kasianov, A. Krasnaya, S. Semyonov, A. Safaraleev

## Data availability

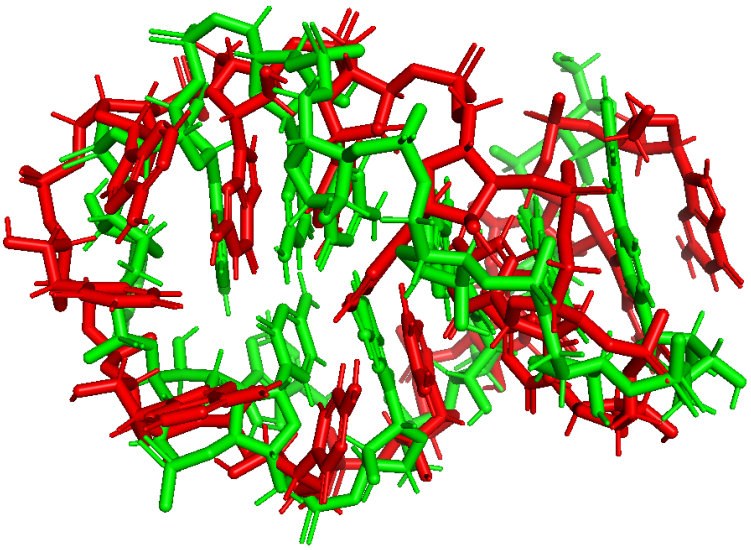
The data and code that support the findings of this study are available on Zenodo and GitHub, respectively. The code used for this study can be accessed at https://github.com/quantori/AutoRNA. The dataset used in this study can be accessed at https://doi.org/10.5281/zenodo.16887142.

**Supplementary material**

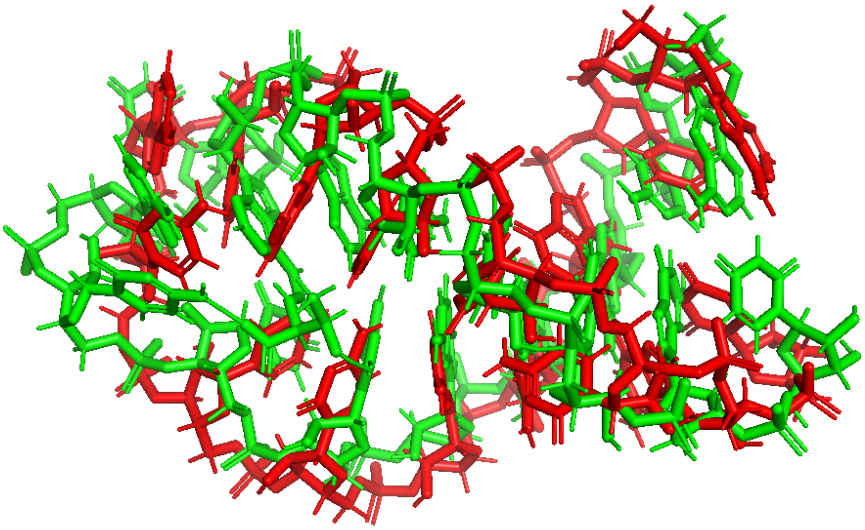
Nucleotide predictions (additional examples).

** Green color reflects the actual structure, red color reflects the predicted structure.*

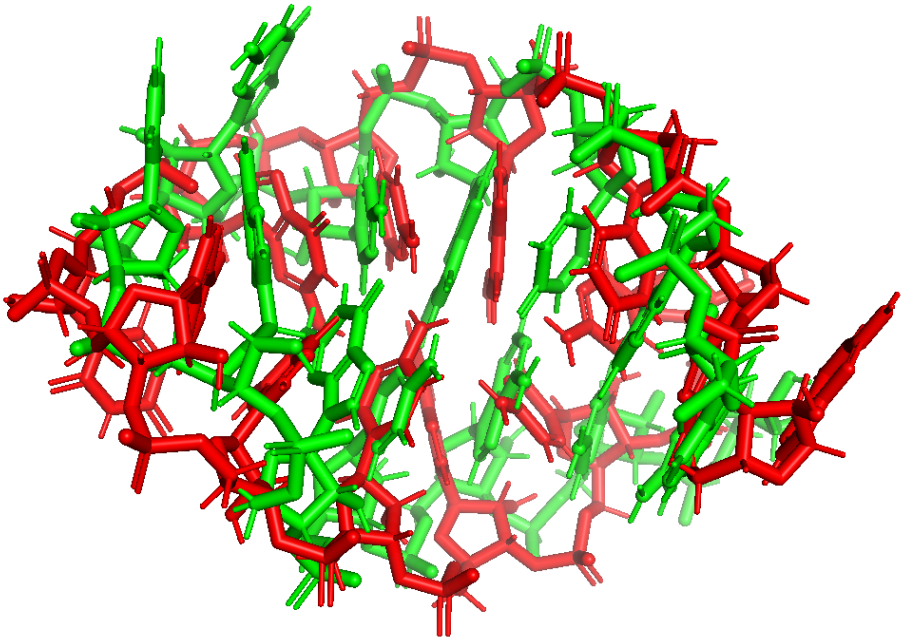
1ESH ↓



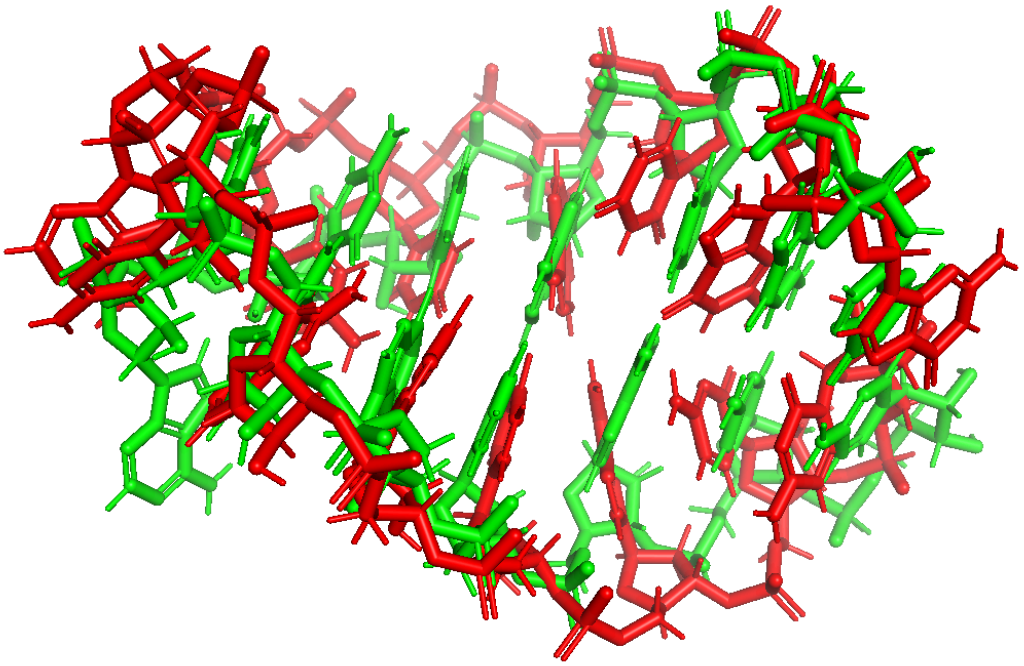
2G1G ↓



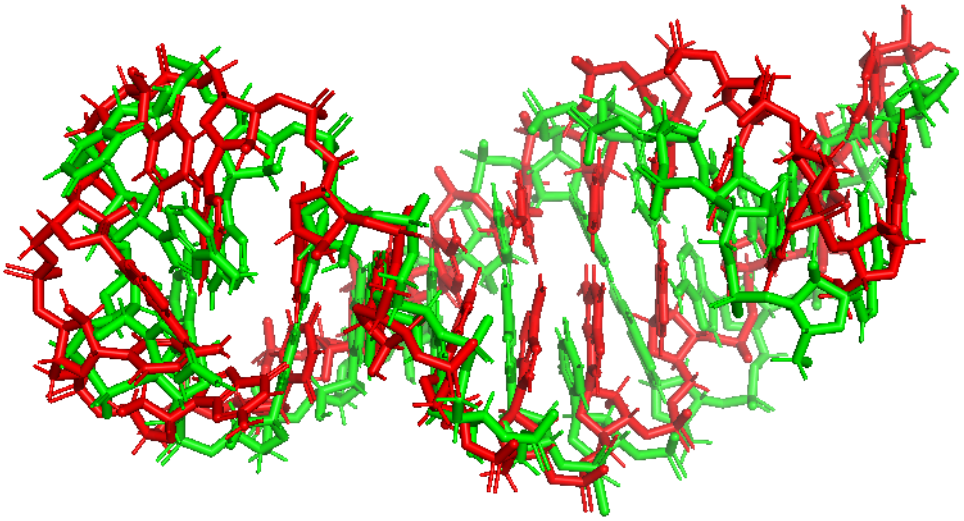
1I46 ↓



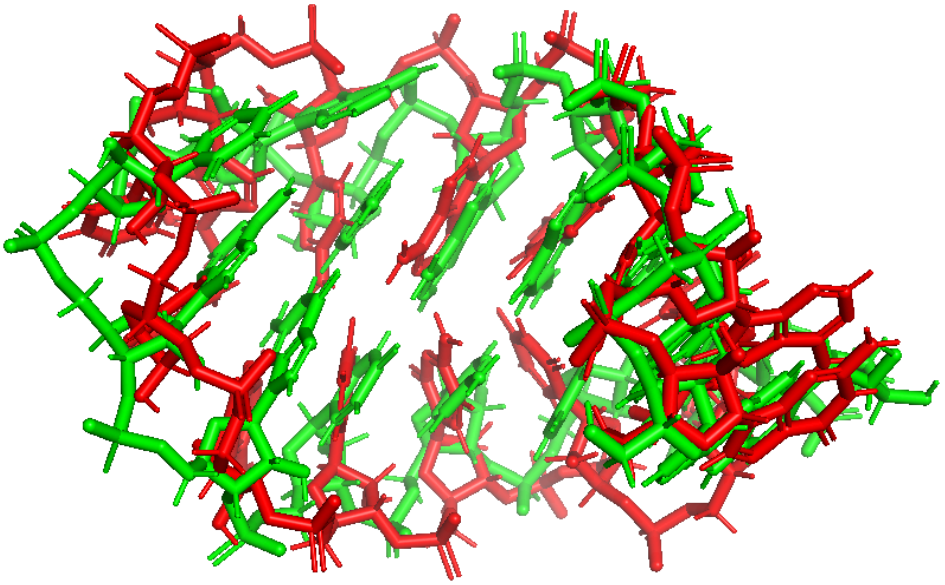
1ROQ ↓



1NZ1 ↓



2LPA ↓



## References


[1] Yu A.M., Choi Y.H., Tu M.J. (2020) RNA drugs and RNA targets for small molecules: principles, progress, and challenges. *Pharmacological Rev.*, 72(4), 862-898.

[2] Jumper J., Evans R., Pritzel A., Green T., Figurnov M., Ronneberger O., Tunyasuvunakool K., Bates R., Žídek A., Potapenko A., Bridgland A., Meyer C., Kohl S.A.A., Ballard A.J., Cowie A., Romera-Paredes B., Nikolov S., Jain R., Adler J., Back T., Petersen S., Reiman D., Clancy E., Zielinski M., Steinegger M., Pacholska M., Berghammer T., Bodenstein S., Silver D., Vinyals O., Senior A.W., Kavukcuoglu K., Kohli P., Hassabis D. (2021) Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873), 583-589.

[3] Abramson, J., Adler, J., Dunger, J., Evans, R., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. (2024). Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature*, 630(8017), 493–500. https://doi.org/10.1038/s41586-024-07487-w

[4] Cruz, J.A., Blanchet, M.F., Boniecki, M., Bujnicki, J.M., Chen, S.J., Cao, S., Das, R., Ding, F., Dokholyan, N.V., Flores, S.C., Huang, L., Lavender, C.A., Lisi, V., Major, F., Mikolajczak, K., Patel, D.J., Philips, A., Puton, T., Santalucia, J., Sijenyi, F., et al. (2012). RNA-Puzzles: a CASP-like evaluation of RNA three-dimensional structure prediction. *RNA*, 18(4), 610–625.

[5] Danaee, P., Rouches, M., Wiley, M., Deng, D., Huang, L., Hendrix, D. (2018). bpRNA: large-scale automated annotation and analysis of RNA secondary structure. *Nucleic Acids Research*, 46(11), 5381–5394. https://doi.org/10.1093/nar/gky285

[6] Gillespie J., Mayne M., Jiang M. (2009) RNA folding on the 3D triangular lattice. *BMC Bioinformatics*, 10, 1-17.

[7] Sharma S., Ding F., Dokholyan N.V. (2008) iFoldRNA: three-dimensional RNA structure prediction and folding. *Bioinformatics*, 24(17), 1951-1952.

[8] Tan Y.L., Liu H., Tan Z.J., Shi Y., Zhou P., Lu Z.J. (2022) rsRNASP: A residue-separation-based statistical potential for RNA 3D structure evaluation. *Biophys. J.*, 121(1), 142-156.

[9] Rahmanov S., Kulakovskiy I., Uroshlev L., Makeev V. (2010) Empirical potentials for ion binding in proteins. *Journal of Bioinformatics and Computational Biology*, 8(03), 427-435.

[10] Case D.A., Cheatham III T.E., Darden T., Gohlke H., Luo R., Merz Jr K.M., Woods R.J. (2005) The Amber biomolecular simulation programs. *Journal of Computational Chemistry*, 26(16), 1668-1688.

[11] Brooks B.R., Brooks C.L. III, MacKerell A.D. Jr., Nilsson L., Petrella R.J., Roux B., Won Y., Archontis G., Bartels C., Boresch S., Caflisch A., Caves L., Cui Q., Dinner A.R., Feig M., Fischer S., Gao J., Hodoscek M., Im W., Kuczera K., Lazaridis T., Ma J., Ovchinnikov V., Paci E., Pastor R.W., Post C.B., Pu J.Z., Schaefer M., Tidor B., Venable R.M., Woodcock H.L., Wu X., Yang W., York

D.M., Karplus M. (2009) CHARMM: The biomolecular simulation program. *Journal of Computational Chemistry*, 30(10):1545–1614. doi:10.1002/jcc.21287.

[12] Al-Hashimi H.M., Walter N.G. (2008) RNA dynamics: it is about time. *Current Opinion in Structural Biology*, 18(3), 321-329.

[13] Li J., Chen S.J. (2021) RNA 3D structure prediction using coarse-grained models. *Frontiers in Molecular Biosciences*, 8, 720937.

[14] Wang W., Feng C., Han R., Wang Z., Ye L., Du Z., Yang J. (2023) trRosettaRNA: automated prediction of RNA 3D structure with transformer network. *Nature Communications*, 14(1), 7266.

[15] Watkins A.M., Rangan R., Das R. (2020) FARFAR2: improved de novo rosetta prediction of complex global RNA folds. *Structure*, 28(8), 963-976.

[16] Fu L., Cao Y., Wu J., Peng Q., Nie Q., Xie X. (2022) UFold: fast and accurate RNA secondary structure prediction with deep learning. *Nucleic Acids Research*, 50(3), e14-e14.

[17] Burley S.K., Berman H.M., Kleywegt G.J., Markley J.L., Nakamura H., Velankar S. (2017) Protein Data Bank (PDB): The single global macromolecular structure archive. In: Read R.J., Schmidt H. (eds). *Methods in Molecular Biology*. Springer, New York, pp. 627-641. https://doi.org/10.1007/978-1-4939-7000-1_26.

[18] Hong, X., Zheng, J., Xie, J., Tong, X., Liu, X., Song, Q., Liu, S., & Liu, S. (2021). RR3DD: an RNA global structure-based RNA three-dimensional structural classification database. *RNA Biology*, *18*(sup2), 738–746. https://doi.org/10.1080/15476286.2021.1989200

[19] Rivas, E. (2020). RNA structure prediction using positive and negative evolutionary information. *PLoS Computational Biology*, *16*(10), e1008387. https://doi.org/10.1371/journal.pcbi.1008387

[20] Petegrosso, R., Li, Z. and Kuang, R. (2019). Machine learning and statistical methods for clustering single-cell RNA-sequencing data. *Briefings in Bioinformatics*. doi:https://doi.org/10.1093/bib/bbz063.

[21] Hong, X., Zheng, J., Xie, J., Tong, X., Liu, X., Song, Q., Liu, S., & Liu, S. (2021). RR3DD: an RNA global structure-based RNA three-dimensional structural classification database. *RNA Biology*, *18*(sup2), 738–746. https://doi.org/10.1080/15476286.2021.1989200

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

[23] Doersch C. (2016) Tutorial on Variational autoencoders. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1606.05908.

[24] Asperti A., Trentin M. (2020) Balancing reconstruction error and kullback-leibler divergence in variational autoencoders. *IEEE Access*, 8, 199440-199448.

[25] Burgess C.P., Higgins I., Pal A., Matthey L., Watters N., Desjardins G., Lerchner A. (2018) Understanding disentangling in beta-VAE. *arXiv.org*. https://arxiv.org/abs/1804.03599.

[26] Eastman, P., Swails, J., Chodera, J.D., McGibbon, R.T., Zhao, Y., Beauchamp, K.A., Wang, L.P., Simmonett, A.C., Harrigan, M.P., Stern, C.D., Wiewiora, R.P., Brooks, B.R., & Pande, V.S. (2017). *OpenMM 7: Rapid development of high performance algorithms for molecular dynamics*. PLOS Computational Biology, **13**(7), e1005659. https://doi.org/10.1371/journal.pcbi.1005659

[27] Kingma D.P., Ba J. (2014) Adam: A method for stochastic optimization. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1412.6980.

[28] Zemla, A. (2003). LGA: a method for finding 3D similarities in protein structures. *Nucleic Acids Research*, *31*(13), 3370–3374. https://doi.org/10.1093/nar/gkg571

[29] Zhang, Y. (2005). TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Research*, *33*(7), 2302–2309. https://doi.org/10.1093/nar/gki524

[30] *tmscoring*. (2019, March 22). PyPI. https://pypi.org/project/tmscoring/

[31] Mucherino A., Papajorgji P.J., Pardalos P.M. (2012) Distance geometry: theory, methods, and applications. Springer Science & Business Media.

[32] Trosset M.W. (1998) A new formulation of the nonmetric strain problem in multidimensional scaling. *J. Classif.*, 15(1), 15-35.

[33] Bassaro, S., Corso, G., Wohlwend, J., et al. (2025). *Boltz-2: Towards accurate and efficient binding affinity prediction.* bioRxiv. https://doi.org/10.1101/2025.02.14.123456

[34] Shen, T., Hu, Z., Sun, S., Liu, D., Wong, F., Wang, J., Chen, J., Wang, Y., Hong, L., Xiao, J., Zheng, L., Krishnamoorthi, T., King, I., Wang, S., Yin, P., Collins, J.J., Li, Y. (2024). Accurate RNA 3D structure prediction using a language model-based deep learning approach. *Nature Methods*, 21, 2287–2298. https://doi.org/10.1038/s41592-024-02487-0

[35] DeLano W.L., Ultsch M.H., Palo Alto C.A. (2002) Pymol: An open-source molecular graphics tool. *CCP4 Newsl. Protein Crystallogr.*, 40(1), 82-92.

[36] Vicens Q., Kieft J.S. (2022) Thoughts on how to think (and talk) about RNA structure. *Proc. Natl. Acad. Sci. U.S.A.*, 119(17), e2112677119.

[37] Qiao, Y., Yang, R., Liu, Y., Chen, J., Zhao, L., Huo, P., Wang, Z., Bu, D., Wu, Y., & Zhao, Y. (2024). DeepFusion: A deep bimodal information fusion network for unraveling protein-RNA interactions using in vivo RNA structures. *Computational and Structural Biotechnology Journal*, 23, 617–625.

[38] Florentino, B. R., Bonidia, R. P., Sanches, N. H., da Rocha, U. N., & de Carvalho, A. C.P.L.F. (2024). BioPrediction-RPI: Democratizing the prediction of interaction between non-coding RNA and protein with end-to-end machine learning. *Computational and Structural Biotechnology Journal*, 23, 2267–2276.

[39] Krautwurst, S., & Lamkiewicz, K. (2024). RNA-protein interaction prediction without high-throughput data: An overview and benchmark of in silico tools. *Computational and Structural Biotechnology Journal, 23*, 4036-4046.

[40] Sun, L. Z., Jiang, Y., Zhou, Y., & Chen, S. J. (2020). RLDOCK: a new method for predicting RNA–ligand interactions. *Journal of chemical theory and computation, 16*(11), 7173-7183.

[41] Pan, X., Fang, Y., Li, X., Yang, Y., & Shen, H. B. (2020). RBPsuite: RNA-protein binding sites prediction suite based on deep learning. *BMC genomics, 21*(1), 884.

[42] Cicconetti, C., Lauria, A., Proserpio, V., Masera, M., Tamburrini, A., Maldotti, M., Oliviero, S., & Molineris, I. (2023). 3plex enables deep computational investigation of triplex-forming lncRNAs. *Computational and Structural Biotechnology Journal,* 21, 3091–3102.

[43]Ballarino, M., Pepe, G., Helmer-Citterich, M., & Palma, A. (2023). *E*xploring the landscape of tools and resources for the analysis of long non-coding RNAs. *Computational and Structural Biotechnology Journal,* 21, 4706–4716.

[44] Cheetham, S. W., Gruhl, F., Mattick, J. S., & Dinger, M. E. (2013). *Long noncoding RNAs and the genetics of cancer. British journal of cancer, 108*(12), 2419-2425.

[45] Gupta, R. A., Shah, N., Wang, K. C., Kim, J., Horlings, H. M., Wong, D. J., ... & Chang, H. Y. (2010). Long non-coding RNA HOTAIR reprograms chromatin state to promote cancer metastasis. *Nature, 464*(7291), 1071-1076.

[46] Braga, E. A., Burdennyy, A. M., Uroshlev, L. A., Zaichenko, D. M., Filippova, E. A., Lukina, S. S., ... & Kushlinskii, N. E. (2024). Ten hypermethylated lncRNA genes are specifically involved in the initiation, progression, and lymphatic and peritoneal metastasis of epithelial ovarian cancer. I*nternational Journal of Molecular Sciences*, *25*(21), 11843.

[47] Brovkina, O. I., Pronina, I. V., Uroshlev, L. A., Fridman, M. V., Loginov, V. I., Kazubskaya, T. P., ... & Braga, E. A. (2021). Identification of novel differentially expressing long non-coding RNAs with oncogenic potential. *Molecular Biology, 55*(4), 548-554.

[48] Pichot, F., Hogg, M. C., Marchand, V., Bourguignon, V., Jirström, E., Farrell, C., ... & Helm, M. (2023). Quantification of substoichiometric modification reveals global tsRNA hypomodification, preferences for angiogenin-mediated tRNA cleavage, and idiosyncratic epitranscriptomes of human neuronal cell-lines. *Computational and Structural Biotechnology Journal*, *21*, 401-417.

[49] Moafinejad, S. N., de Aquino, B. R., Boniecki, M. J., Pandaranadar Jeyeram, I. P., Nikolaev, G., Magnus, M., ... & Bujnicki, J. M. (2024). SimRNAweb v2. 0: a web server for RNA folding simulations and 3D structure modeling, with optional restraints and enhanced analysis of folding trajectories. *Nucleic Acids Research*, *52*(W1), W368-W373.