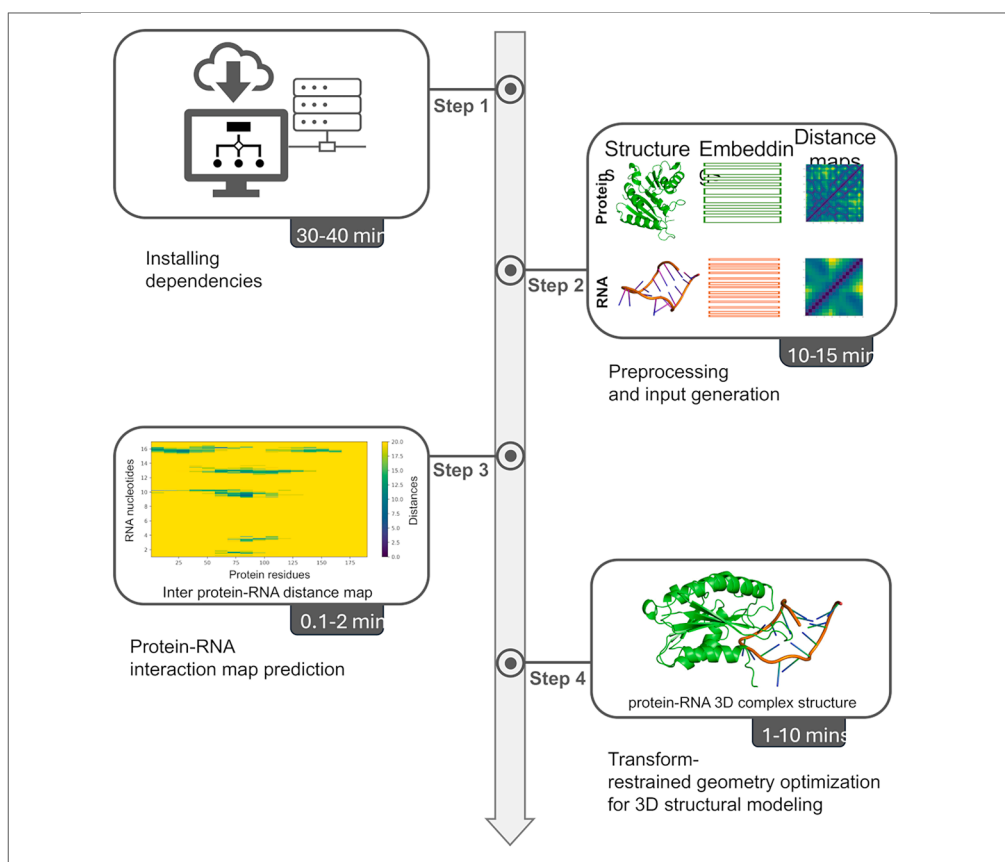


## Protocol

# A protocol for single-sequence protein-RNA complex structure prediction using ProRNA3D-single



Rahmatullah Roche,  
Sumit Tarafder,  
Debswapna  
Bhattacharya

dbhattacharya@vt.edu

### Highlights

A protocol for generating three-dimensional protein-RNA complex structural models

Steps for preprocessing and feature generation

Instructions for executing the inference script and optimization algorithms

Characterizing the interaction between protein and RNA is critically important for biology and medicine. Here, we present a protocol to predict three-dimensional protein-RNA complex structural models from their sequences using a deep learning-based approach named ProRNA3D-single. First, we delineate detailed steps for predicting protein-RNA interaction maps by leveraging pretrained language models. We then describe the procedure for generating the protein-RNA complex structure from the predicted interactions using optimization techniques. This protocol has potential applications in host-virus interactions and drug design.

Publisher's note: Undertaking any experimental protocol requires adherence to local institutional guidelines for laboratory safety and ethics.

Roche et al., STAR Protocols 7, 104502

June 19, 2026 © 2026 The Authors. Published by Elsevier Inc.

<https://doi.org/10.1016/j.xpro.2026.104502>



## Protocol

## A protocol for single-sequence protein-RNA complex structure prediction using ProRNA3D-single

Rahmatullah Roche,<sup>1</sup> Sumit Tarafder,<sup>2</sup> and Debswapna Bhattacharya<sup>2,3,4,\*</sup><sup>1</sup>TSYS School of Computer Science, Columbus State University, Columbus, GA 31906, USA<sup>2</sup>Department of Computer Science, Virginia Tech, Blacksburg, VA 24061, USA<sup>3</sup>Technical contact<sup>4</sup>Lead contact\*Correspondence: [dbhattacharya@vt.edu](mailto:dbhattacharya@vt.edu)  
<https://doi.org/10.1016/j.xpro.2026.104502>

## SUMMARY

Characterizing the interaction between protein and RNA is critically important for biology and medicine. Here, we present a protocol to predict three-dimensional protein-RNA complex structural models from their sequences using a deep learning-based approach named ProRNA3D-single. First, we delineate detailed steps for predicting protein-RNA interaction maps by leveraging pre-trained language models. We then describe the procedure for generating the protein-RNA complex structure from the predicted interactions using optimization techniques. This protocol has potential applications in host-virus interactions and drug design.

For complete details on the use and execution of this protocol, please refer to Roche et al.<sup>1</sup>

## BEFORE YOU BEGIN

Knowledge of protein-RNA interactions is critical to our understanding of cellular biology,<sup>2–4</sup> with potential impact on drug design and discovery.<sup>5–8</sup> Numerous computational methods have been developed to predict the RNA interaction sites on protein<sup>9–13</sup> or protein interaction sites on RNA<sup>14–17</sup> attaining promising prediction performance. However, these methods do not capture the three-dimensional (3D) interactions between protein and RNA in Euclidean space, which convey finer-grained complex structural information. In recent years, several deep-learning-based methods have been developed for predicting complex biomolecular structural models.<sup>18–20</sup> However, their dependence on limited co-evolutionary information may affect their prediction performance. To address this issue, we developed ProRNA3D-single,<sup>1</sup> a method for predicting protein-RNA complex structural models from their single sequence information using pretrained language models, thereby bypassing the need for explicit co-evolutionary information. In the ProRNA3D-single pipeline, first protein and RNA structures and embeddings are predicted using their respective pretrained language models. Then using geometric pairing of language model embeddings, an inter-protein-RNA distance map is predicted. Finally, the distance map along with the predicted protein and RNA structures is optimized to generate a 3D protein-RNA complex structural model.

In this paper, we present detailed step-by-step protocol of ProRNA3D-single. We use a human RNA-binding protein (PDB ID: 7ZLQ, protein chain B and RNA chain C) to demonstrate the potential application of the protocol for predicting protein-RNA interaction.



### Innovation

Existing deep-learning-based techniques utilize the co-evolutionary information derived from multiple sequence alignments (MSAs). However, the scarcity of protein-RNA paired MSA information can negatively impact their performance. ProRNA3D-single utilizes pretrained protein and RNA language models, pairing their embeddings using symmetry-aware graph convolutions.<sup>21</sup> Particularly, we use ESM-2<sup>22</sup> and RNA-FM<sup>23</sup> to generate embeddings for proteins and RNAs. We generate the embeddings from the single amino acid sequence of protein and nucleotide sequence for RNA. The ESM-2 generates protein embedding with a shape of  $L1 \times 1280$ , and the RNA-FM generates embeddings with the shape of  $L2 \times 640$ , where  $L1$ , and  $L2$  are the length of the protein and RNA sequences, respectively. Additionally, we generate the monomeric 3D structures of the protein using ESMFold<sup>22</sup> and the RNA using E2EFold-3D (later evolved to RhoFold+).<sup>24</sup> Next, the embeddings and the 3D monomeric structures are transformed into structure-aware enriched representations through graph convolutions using four layers of EGNN.<sup>21</sup> The dimensions of the enriched protein and RNA representations are  $L1 \times D$  and  $L2 \times D$ , respectively. Through a pairwise combination, the protein-RNA pair embedding is generated, having a dimension of  $L1 \times L2 \times 2D$ . The resulting protein-RNA pair embedding eliminates the need for paired MSAs. Subsequently, the pair embedding is passed through the ResNet inception blocks,<sup>25</sup> followed by a geometric attention module to predict two-dimensional protein-RNA interaction maps. Finally, the interaction maps are utilized to generate 3D complex structural models via PyRosetta optimization.<sup>26</sup>

### QuickStart: Install dependencies

⌚ Timing: 30–40 min

On a Linux x86\_64 operating system, ProRNA3D-single was trained using a single NVIDIA A100-SXM4-80GB GPU and tested using CPUs (GPUs can be used if available). After preprocessing, the RAM and disk requirements for running the pipeline depend on the input size (protein and RNA lengths); however, a minimum of 32 GB RAM and 256 GB of disk storage is recommended for typical runs.

This section guides the setup and installation of dependencies to prepare ProRNA3D-single protocol ready to go. The detailed prediction steps are described in “[preprocessing](#)” and “[step-by-step method details](#)” sections.

1. Download this GitHub package of the protocol:
  - a. Run the following command:

```
>git clone https://github.com/Bhattacharya-Lab/ProRNA3D-single
```

- b. And then go to the directory:

```
>cd ProRNA3D-single
```

- c. Download the trained model for prediction. The model should be saved as ‘ProRNA3D\_model/model.pt’:

```
>curl --output ProRNA3D_model/model.pt
https://zenodo.org/records/11477127/files/model.pt?download=1
```

**Alternatives:** Use this link <https://zenodo.org/records/11477127/files/model.pt> to directly download the trained model.

2. Create Conda virtual environment to install dependencies.
  - a. Run the following command:

```
>conda env create -f ProRNA3D-single_environment.yml
```

- b. And then activate the virtual environment named as 'ProRNA3D-single'.

```
>conda activate ProRNA3D-single
```

**Note:** Specifically, Python version 3.10.4, PyTorch version 1.12.0, CUDA version 11.6 are used in this protocol.

3. Install the following packages for preprocessing and input feature generation:
  - a. Install PyRosetta<sup>26</sup> following the instructions from <https://www.pyrosetta.org/downloads#h.iwt5ktel05jc>.

△ **CRITICAL:** Use of PyRosetta requires a license. PyRosetta is free of charge for academic and non-profit organizations under a non-commercial license, and commercial use requires a paid license. Detailed licensing information is available at <https://els2.comotion.uw.edu/product/pyrosetta>.

- b. Install pretrained protein language models:
    - i. For protein embedding generation, install ESM-2<sup>22</sup> following the instructions from <https://github.com/facebookresearch/esm?tab=readme-ov-file#quickstart>.

**Note:** 'esm2\_t33\_650M\_UR50D' model is used for embedding generation, which is ESM-2 with 33 layer, 650M parameter and trained on uniref50.<sup>27</sup>

Protein monomeric structural model can be predicted from its sequence using ESMFold.<sup>22</sup> Use 'esmfold\_v1()', which is the ESMFold baseline model trained with 15B parameters for structure prediction purposes.

**Alternatives:** ProRNA3D-single is also compatible with native protein monomeric structures (bound or unbound) or structural models predicted by other methods.

- c. Install pretrained RNA language models:
    - i. For RNA embedding generation, install RNA-FM<sup>23</sup> following the instructions from <https://github.com/ml4bio/RNA-FM?tab=readme-ov-file#setup-and-usage>.
    - ii. For RNA monomeric structural model prediction from its sequence, install E2EFold-3D (later evolved to RhoFold+)<sup>24</sup> following the steps from here <https://github.com/ml4bio/RhoFold>.

**Alternatives:** ProRNA3D-single is also compatible with native RNA monomeric structures (bound or unbound) or structural models predicted by other methods.

## Preprocessing

⌚Timing: 10–15 min

This section describes the preprocessing required to generate the input features for the ProRNA3D-single model inference. The following features should be generated and placed inside the 'inputs/' directory. Here, we use an example of an RNA-binding protein in Homo sapiens (ID: 7ZLQ, protein chain B and RNA chain C) to generate input features, predict protein-RNA interaction map and 3D complex structural model. In the preprocessing stage, inputs are protein (and RNA) chain amino (and nucleic acid) sequences. The outputs are features that include protein and RNA embeddings, predicted structural models (when native structures are unavailable), and corresponding distance maps.

The preprocessing steps are as follows:

- Place the list of targets named as 'inputs.list' within the 'inputs/' directory.

⚠ **CRITICAL:** Each line of the file contains the <target id>\_<protein chain><RNA chain>. For this example, the 'inputs/inputs.list' contains only one line.

```
7ZLQ_BC
```

**Note:** Here, 7ZLQ is the target id, B is the protein chain and C is the RNA chain.

⚠ **CRITICAL:** The input list file must be named exactly 'inputs.list' and placed in the 'inputs/' directory. When the pipeline is executed, only the targets listed in this file are processed.

- Generate protein embedding using ESM-2 following the coding example from here <https://github.com/facebookresearch/esm#getting-started-with-this-repo>

Save the protein embedding as 'inputs/<protein>.rep\_1280.npy'.

⚠ **CRITICAL:** For this example, the generated protein embedding is saved as 'inputs/7ZLQB.rep\_1280.npy'.

- Generate RNA embedding using RNA-FM by following this coding example <https://github.com/ml4bio/RNA-FM?tab=readme-ov-file#quick-start-usage>.

⚠ **CRITICAL:** Save the RNA embedding as 'inputs/<RNA>\_RNA.npy'. For this example, the generated RNA embedding is saved as 'inputs/7ZLQC\_RNA.npy'.

- If the native protein or RNA monomeric structures are available, place them inside the 'inputs/' directory.

**Note:** The protein and/or RNA monomeric structures can be predicted from their sequences using different prediction methods. Here is an example for each:

- Predict protein monomeric structural models using ESMFold following the instructions from here <https://github.com/facebookresearch/esm?tab=readme-ov-file#esmfold>.

For this example, protein monomeric structure is generated using ESMFold, and saved as 'inputs/7ZLQB.pdb'.

- b. Predict RNA monomeric structural models using E2EFold-3D (later evolved to RhoFold+) following the instructions from here <https://github.com/ml4bio/RhoFold?tab=readme-ov-file#examples>.  
For this example, RNA monomeric structure is generated using E2EFold-3D, and saved as 'inputs/7ZLQC.pdb'.
8. Generate distance maps from protein and RNA structures and save as 'prot\_dist/<protein>\_prot.dist' and 'rna\_dist/<RNA>\_RNA.c4p.dist', respectively.

△ **CRITICAL:** For this example, protein and RNA distance maps are saved as 'prot\_dist/7ZLQB\_prot.dist', and 'rna\_dist/7ZLQC\_RNA.c4p.dist', respectively.

**Note:** The distance maps are obtained from the protein and RNA monomeric 3D structures. Distance maps are generally formatted with five columns. The first and second columns of the distance maps represent the residue (nucleotide for RNA) pairs, and the fifth column denotes the Euclidean pairwise distance between the corresponding residues (nucleotides for RNA). Specifically, the fifth column represents the distance between the C $\alpha$  (C4' for RNA) atoms of the corresponding residue (nucleotide for RNA) pairs. From the monomeric 3D structures, distance maps can be generated using any publicly available script such as this one <https://github.com/multicom-toolbox/CONFOLD2/blob/master/confold-v2.0/suppl-scripts/pdb2rr.pl>.

## KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
<b>Deposited data</b>		
ProRNA3D-single inference output for the example target	Roche et al. <sup>1</sup>	<a href="https://github.com/Bhattacharya-Lab/ProRNA3D-single/blob/main/out_inter_rr/7ZLQ_BC.10bins.out">https://github.com/Bhattacharya-Lab/ProRNA3D-single/blob/main/out_inter_rr/7ZLQ_BC.10bins.out</a>
ProRNA3D-single final output (3D protein-RNA complex structure) for the example target	Roche et al. <sup>1</sup>	<a href="https://github.com/Bhattacharya-Lab/ProRNA3D-single/blob/main/predictions/7ZLQ_BC.pdb">https://github.com/Bhattacharya-Lab/ProRNA3D-single/blob/main/predictions/7ZLQ_BC.pdb</a>
Preprocessed inputs	Roche et al. <sup>1</sup>	<a href="https://github.com/Bhattacharya-Lab/ProRNA3D-single/blob/main/inputs/">https://github.com/Bhattacharya-Lab/ProRNA3D-single/blob/main/inputs/</a>
Distance maps	Roche et al. <sup>1</sup>	<a href="https://github.com/Bhattacharya-Lab/ProRNA3D-single/blob/main/prot_dist/">https://github.com/Bhattacharya-Lab/ProRNA3D-single/blob/main/prot_dist/</a> <a href="https://github.com/Bhattacharya-Lab/ProRNA3D-single/blob/main/rna_dist/">https://github.com/Bhattacharya-Lab/ProRNA3D-single/blob/main/rna_dist/</a>
Pretrained model	Roche et al. <sup>1</sup>	<a href="https://zenodo.org/records/11477127">https://zenodo.org/records/11477127</a>
<b>Software and algorithms</b>		
PyTorch	Paszke et al. <sup>28</sup>	<a href="https://pytorch.org/">https://pytorch.org/</a>
ESMFold and ESM-2	Lin et al. <sup>22</sup>	<a href="https://github.com/facebookresearch/esm">https://github.com/facebookresearch/esm</a>
E2EFold-3D (later evolved to RhoFold+)	Shen et al. <sup>24</sup>	<a href="https://github.com/ml4bio/RhoFold">https://github.com/ml4bio/RhoFold</a>
RNA-FM	Cheng et al. <sup>23</sup>	<a href="https://github.com/ml4bio/RNA-FM">https://github.com/ml4bio/RNA-FM</a>
PyRosetta	Chaudhury et al. <sup>26</sup>	<a href="https://www.pyrosetta.org/">https://www.pyrosetta.org/</a>
Anaconda	Continuum Analytics Inc.	<a href="https://www.continuum.io/downloads">https://www.continuum.io/downloads</a>
ProRNA3D-single	Roche et al. <sup>1</sup>	Zenodo [ <a href="https://doi.org/10.5281/zenodo.16791755">https://doi.org/10.5281/zenodo.16791755</a> ] from <a href="https://github.com/Bhattacharya-Lab/ProRNA3D-single/tree/v1.0.0">https://github.com/Bhattacharya-Lab/ProRNA3D-single/tree/v1.0.0</a>
PyMol	Schrödinger LLC <sup>29</sup>	<a href="https://pymol.org/">https://pymol.org/</a>
Matplotlib	Hunter et al. <sup>30</sup>	<a href="https://matplotlib.org/">https://matplotlib.org/</a>

## STEP-BY-STEP METHOD DETAILS

After downloading and installing the dependencies and preprocessing the input features, the pipeline is ready to run. Below is an overview of the workflow (a more detailed workflow is shown in the graphical abstract):

Installing dependencies → Preprocessing → Performing inference → Generating restraints → Optimizing the restraints to generate the 3D protein–RNA complex structure.

**Note:** After completing the preprocessing, the entire ProRNA3D-single pipeline can be executed using the following command:

```
>python run_predictions.py
```

This ‘run\_predictions.py’ is a pipeline wrapper that first predicts the protein-RNA interaction maps and then executes the PyRosetta optimizations<sup>26</sup> to generate 3D protein-RNA complex structural model using the restraints derived from the predicted interaction maps. The above command is sufficient to generate final product (3D complex structural models).

The pipeline processes only the targets explicitly listed in the ‘inputs.list’ file. When multiple PDB files are present in the inputs/directory, only the PDBs corresponding to the target names specified in ‘inputs.list’ are selected for processing, regardless of other files in the directory. To analyze a different target, the corresponding target name must be added to (or replaced in) the ‘inputs.list’ file, and there is no changes to the command prompts.

Here is a detailed step-by-step discussion of the prediction pipeline:

### Prediction of protein-RNA interaction maps

⌚ Timing: 0.1–2 min

In this stage, inference is performed on the pretrained model. Feature files generated during the preprocessing step (protein and RNA embeddings, their monomeric structures, and distance maps) go as input, and the output is a predicted interaction map for each target.

1. Run the following command:

```
>python ProRNA3D-single.py --model_state_dict ProRNA3D_model/model.pt --outdir out_inter_rr/
```

**Note:** The inference program ‘ProRNA3D-single.py’ predicts the protein-RNA interaction maps for the targets listed in ‘inputs/inputs.list’, using the downloaded pretrained model saved in ‘ProRNA3D\_model/model.pt’.

⚠ **CRITICAL:** This program requires the protein and RNA distance maps, and the corresponding input features saved in ‘inputs/’ directory.

The generated output file is saved as 'out\_inter\_rr/<target\_id>\_<protein chain><RNA chain>.10bins.out'. In this example, for target 7ZLQ\_BC, the output is saved as 'out\_inter\_rr/7ZLQ\_BC.10bins.out' (see details in 'expected outcomes' section).

**Note:** Each output file contains 12 columns. The first and the second column correspond to the interacting protein residue (C $\alpha$  atom) and the RNA nucleotide (C4' atom), respectively. The remaining columns represent the likelihood of their interaction across 10 distance thresholds. Specifically, columns 3-12 indicate the propensity of the C $\alpha$ -C4' distance within 2.5 Å, 4 Å, 6 Å, 8 Å, 10 Å, 12 Å, 14 Å, 16 Å, 18 Å and 20 Å.

**△ CRITICAL:** For multiple targets, list all the targets in the 'inputs/inputs.list', and the command mentioned in Step 1 is executed only once.

### Restraint generation from output interaction maps

⌚ Timing: 0.1–1 min

In this stage, restraints are derived from the predicted protein-RNA interaction maps and constraint file is created for input into PyRosetta<sup>26</sup> optimizer for full-atom protein-RNA complex structural prediction.

2. Generate the constraint file containing the restraints for PyRosetta,<sup>26</sup> using gen\_rst.py script by executing the following command:

```
>python gen_rst.py --rrfile out_inter_rr/<target id>_<protein chain><RNA chain>.10bins.out  
--prot_chain <protein chain> --rna_chain <RNA chain> --rstfile inputs/tmp/<target id>_<protein  
chain><RNA chain>.rst
```

**△ CRITICAL:** The output file is saved as 'inputs/tmp/<target id>\_<protein chain><RNA chain>.rst'. For this example, the command is:

```
>python gen_rst.py --rrfile out_inter_rr/7ZLQ_BC.10bins.out --prot_chain B --rna_chain C  
--rstfile inputs/tmp/7ZLQ_BC.rst
```

The constraint file is saved as 'inputs/tmp/7ZLQ\_BC.rst'.

**Note:** AtomPair 'BOUNDED' restraints with 'SCALARWEIGHTEDFUNC' function is used. For each of the C $\alpha$ -C4' pair, the propensity of the interatomic distance falling within thresholds ranging from 2.5 Å to 20 Å is evaluated. If the propensity is greater than or equal to 0.5 for a particular threshold, treat this as the upper bound of the restraint. The lower bound is fixed to 2.0 Å. The propensity value itself is used as the scalar weight. Here is an example of restraint:

```
AtomPair CA 19 B C4' 8C SCALARWEIGHTEDFUNC 0.83 BOUNDED 2.00 8.00 1.0 0.5 tag.
```

Here, 'CA 19B' represents the C $\alpha$  atom of residue 19 of protein chain B, and 'C4' 8C' represents the C4' atom of nucleotide 8 of RNA chain C. The scalar weight is 0.83, which will be multiplied by the

BOUNDED function during optimization, by definition. For the BOUNDED distances, the upper limit is 8 Å and the lower limit is 2 Å. The standard deviation is 1.0, and 'rswitch' is set to 0.5.

### Folding 3D complex structures

⌚ Timing: 0.1–10 min

The constraint file along with protein and RNA monomeric 3D structures, is passed as input into PyRosetta<sup>26</sup> optimization algorithm to generate the three-dimensional protein-RNA complex structural model as the final output.

⚠ **CRITICAL:** Before starting to generate the 3D complex structures, ensure that the constraint files as well as the initial structures are created to start with the process.

3. Create initial structure by executing the following command:

```
>python preprocess_monomers.py --prot inputs/<target id><protein chain>.pdb --rna inputs/
<target id><RNA chain>.pdb --combined inputs/tmp/<target id>_<protein chain><RNA chain>_
init.pdb --prot_chain <protein chain> --rna_chain <RNA chain>
```

⚠ **CRITICAL:** The 'preprocess\_monomers.py' script appends the coordinates of RNA to the end of protein coordinates and updates the corresponding chain identifiers in the newly generated PDB file.

In our example, the command is:

```
>python preprocess_monomers.py --prot inputs/7ZLQB.pdb --rna inputs/'7ZLQC.pdb --combined
inputs/tmp/7ZLQ_BC_init.pdb --prot_chain B --rna_chain C
```

The initial complex structure is saved as 'inputs/tmp/<target id>\_<protein chain><RNA chain>\_init.pdb'. In our example, it is saved as 'inputs/tmp/7ZLQ\_BC\_init.pdb'.

4. Generate protein-RNA complex structure by executing the following command:

```
>python folding.py --rst_file inputs/tmp/<target id><protein chain><RNA chain>.rst --start_pdb
inputs/tmp/<target id>_<protein chain><RNA chain>_init.pdb --refined_pdb predictions/<target
id>_<protein chain><RNA chain>.pdb
```

⚠ **CRITICAL:** The folding script, 'folding.py' (<https://rosettacommons.github.io/PyRosetta.notebooks/>) is used to generate the final product ('predictions/<target id>\_<protein chain><RNA chain>.pdb') of this pipeline. In our example, protein-RNA complex structure ('7ZLQ\_BC.pdb') is generated with the command:

```
>python folding.py --rst_file inputs/tmp/7ZLQ_BC.rst --start_pdb inputs/tmp/7ZLQ_BC_
init.pdb --refined_pdb predictions/7ZLQ_BC.pdb'
```

**Note:** For the PyRosetta initialization parameters, we use '-hb\_cen\_soft', a softer version of 'cen\_soft' with constant seed. Then the 'pose' (3D atomic position) is initialized as follows:

```
>pose = pose_from_pdb(start_pdb)
```

Here, in our example, the 'start\_pdb' is 'inputs/tmp/7ZLQ\_BC\_init.pdb'. PyRosetta Energy function 'REF2015' is used as the scoring function as follows:

```
>sf_fa = create_score_function('ref2015')
```

The score function utilizes the atom pair restraints derived from the generated constraint file, and the initial pose is converted to an all-atom protein-RNA complex as follows:

```
>sf_fa.set_weight(rosetta.core.scoring.atom_pair_constraint, rst_weight)
>switch = SwitchResidueTypeSetMover("fa_standard")
>switch.apply(pose)
```

Here, 'rst\_weight' is initialized as 10.

For the PyRosetta optimization using the defined score function, a FastRelax mover object is created to perform iterative energy minimization on torsion angles and atomic coordinates of the initial pose of the complex structure. The 'MoveMap()' object keeps protein and RNA backbones and sidechains fixed, treating protein and RNA structures rigid during docking. The restraints generated from inference are applied to the initial pose, which drives the re-orientation of the protein and the RNA, resulting their complex 3D structural model. The maximum iteration of the FastRelax is set to 500.

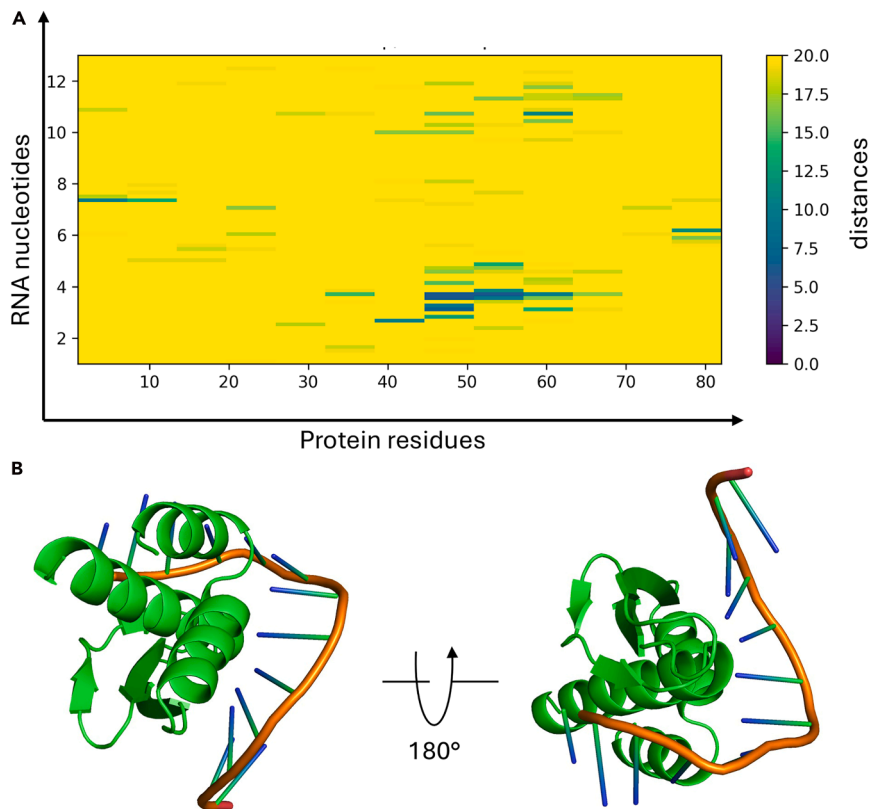
**△ CRITICAL:** For multiple targets, Steps 2-4 should be executed iteratively for each target listed in the input file 'inputs/inputs.list', as specified in the pipeline script 'run\_prediction.py'.

### EXPECTED OUTCOMES

The final output PDB formatted file, consisting of all-atomic 3D coordinates of protein-RNA dimer is saved in 'predictions/' directory. In this example, the final output is saved as 'predictions/7ZLQ\_BC.pdb'. The fnat score (fraction of native interface contact present in the predicted structure, where an interface contact is defined as residue-nucleotide heavy atom distance within 5 Å) for the predicted is 0.385, indicating reasonable agreement with the experimentally validated structure. In addition to the final complex structural model, an intermediate interaction map and the corresponding restraint (constraint) file are generated.

**Optional:** To generate distance maps at the inference, modify the inference script 'ProRNA3D-single.py' after line 97. Create a new distance list as:

```
>list = [2.5, 3, 3.5, 4, 4.5, 5, 5.5, 6, 6.5, 7, 7.5, 8, 8.5, 9, 9.5, 10, 10.5, 11, 11.5, 12, 12.5, 13, 13.5, 14, 14.5, 15, 15.5, 16, 16.5, 17, 17.5, 18, 18.5, 19, 19.5, 20, 100000]
```



**Figure 1. Protein-RNA interaction map and the complex structure predicted using ProRNA3D-single**

(A) The visualization of the finer-grained inter-protein-RNA interaction (distance) map with distances ranging from 0 to 20 Å. The X-axis corresponds to protein residues, and the Y-axis corresponds to RNA nucleotides. Darker regions indicate closer proximity between the corresponding residue-nucleotide pairs.

(B) Cartoon representation of the predicted 3D protein-RNA complex structure of the target (PDB ID:7ZLQ, protein chain B and RNA chain C) using ProRNA3D-single protocol. The interacting protein and RNA are colored green and orange, respectively.

This list specifies the finer-grained distance bins employed in the inference stage. The final value is set to '100000', denoting the absence of interaction between the corresponding protein residue and RNA nucleotide. Then examine the likelihood of each residue-nucleotide pair not being in contact by checking the predicted probability of the last class (distance greater than 20 Å). If that value is greater or equal to 0.5, the residue-nucleotide pair is classified as non-interacting. Otherwise, determine the most probable distance bin among the remaining classes (for example, by using 'numpy.argmax()'). Once a distance bin is determined based on prediction probability, subtract 0.25 from the bin value to obtain the midpoint of upper and lower bounds, which is considered as an approximate distance prediction between the corresponding residue (C $\alpha$  atom) and nucleotide (C4') pair. Finally, save the finer-grained inter-protein-RNA distance information for all residue-nucleotide pairs to a separate file.

As an example, for residue 19 and nucleotide 8, 'numpy.argmax()' returns 8, indicating the distance bin with value 6.5. After subtracting 0.25, the estimated distance (Å) for the residue-nucleotide pair is 6.25.

Figure 1 demonstrates the expected outputs for the example target 7ZLQ\_BC. The 2D heatmap (Figure 1A) represents the predicted inter-protein-RNA distance map for the target. In the predicted distance map, the darker region corresponds to the residue-nucleotide pairs in closer proximity. This distance or interaction map is ultimately translated into the 3D complex structure (Figure 1B).

### LIMITATIONS

ProRNA3D-single demonstrates robustness and resilience in prediction performance.<sup>1</sup> However, this method has several limitations. First, the pipeline relies on protein and RNA monomeric structures generated independently by other methods from their respective sequences.<sup>23,24</sup> In many cases, the monomeric structural models are generated with suboptimal accuracy, which may negatively affect the overall accuracy of the predicted protein-RNA complex structural model. Secondly, the optimization stage that converts the predicted interaction map into all-atomic complex structural model could be improved by integrating this into a unified deep-learning-based predictive modeling framework. Finally, in its current form, ProRNA3D-single pipeline is limited to predicting protein-RNA dimeric complex structures. In future, we plan to extend the prediction pipeline to support multi-chain and multi-molecular complex structural prediction.

### TROUBLESHOOTING

#### Problem 1

Only the protein or RNA component appears in the output complex structural model.

#### Potential solution

Verify that both the protein and RNA monomeric structures are generated/saved and the files are non-empty before starting [step-by-step methods details](#), Step 1.

#### Problem 2

Folding ([step-by-step methods details](#), Step 4) is taking longer than expected.

#### Potential solution

Check if the constraint file contains no restraints (i.e., the file is empty). If the constraint file is empty, it suggests that the protein-RNA interaction map is predicted with very low propensity. Consider lowering the probability threshold from the default value of 0.5 in the 'gen\_rst.py' file by modifying lines 20-49.

If the constraint file is not empty, consider lowering the default maximum number of iteration for FastRelax ([step-by-step methods details](#), Step 4, script folding.py at line 30).

### RESOURCE AVAILABILITY

#### Lead contact

Further information and requests for resources and reagents should be directed to and will be fulfilled by the lead contact, Debswapna Bhattacharya ([dbhattacharya@vt.edu](mailto:dbhattacharya@vt.edu)).

#### Technical contact

Technical questions on executing this protocol should be directed to and will be answered by the technical contact, Debswapna Bhattacharya ([dbhattacharya@vt.edu](mailto:dbhattacharya@vt.edu)).

#### Materials availability

This study did not generate new materials.

#### Data and code availability

- This paper analyzed existing, publicly available data from <https://www.rcsb.org/structure/7ZLO>.
- All original code has been deposited at Zenodo (DOI: <https://doi.org/10.5281/zenodo.16791755>) and at GitHub (<https://github.com/Bhattacharya-Lab/ProRNA3D-single/tree/v1.0.0>), and both are publicly available as of the date of publication. DOIs are also listed in the [key resources table](#). To generate figures, PyMol version 2.5.2 (<https://www.pymol.org/>) and Matplotlib version 3.7.4 are used.
- Any additional information required to reanalyze the data reported in this paper is available from the [lead contact](#) upon request.

### ACKNOWLEDGMENTS

This work was made possible in part by a grant of high-performance computing resources and technical support from the National AI Research Resource Pilot award (NAIRR240093 to D.B.). This work was partially supported by the National

Institute of General Medical Sciences (R35GM138146 to D.B.) and the National Science Foundation (DBI2208679 to D.B.).

## AUTHOR CONTRIBUTIONS

R.R., S.T., and D.B. conceptualized this study. R.R. implemented the computational method, performed the experiments, and analyzed the results. S.T. conducted validation and visualizations. R.R. wrote the original draft, and all other authors reviewed and edited the manuscript. D.B. provided funding and supervision.

## DECLARATION OF INTERESTS

The authors declare no competing interests.

## REFERENCES

- Roche, R., Tarafder, S., and Bhattacharya, D. (2025). Single-sequence protein-RNA complex structure prediction by geometric attention-enabled pairing of biological language models. *Cell Syst.* *16*, 101400. <https://doi.org/10.1016/j.cels.2025.101400>.
- Caudron-Herger, M. (2025). Uncovering the complexity of RNA-protein interactions. *Nat. Rev. Mol. Cell Biol.* *26*, 499.
- Glisovic, T., Bachorik, J.L., Yong, J., and Dreyfuss, G. (2008). RNA-binding proteins and post-transcriptional gene regulation. *FEBS Lett.* *582*, 1977–1986.
- König, J., Zarnack, K., Luscombe, N.M., and Ule, J. (2012). Protein-RNA interactions: new genomic technologies and perspectives. *Nat. Rev. Genet.* *13*, 77–83.
- Park, H.-J., Bang, Y.-J., Kwon, S.P., Kwak, W., Park, S.-I., Roh, G., Bae, S.-H., Kim, J.-Y., Kwak, H.W., Kim, Y., et al. (2023). Analyzing immune responses to varied mRNA and protein vaccine sequences. *npj Vaccines* *8*, 84.
- Willis, J.R., Prabhakaran, M., Muthui, M., Naidoo, A., Sincomb, T., Wu, W., Cottrell, C.A., Landais, E., deCamp, A.C., Keshavarzi, N.R., et al. (2025). Vaccination with mRNA-encoded nanoparticles drives early maturation of HIV bnAb precursors in humans. *Science* *389*, eadr8382.
- Grippin, A.J., Marconi, C., Copling, S., Li, N., Braun, C., Woody, C., Young, E., Gupta, P., Wang, M., Wu, A., et al. (2025). SARS-CoV-2 mRNA vaccines sensitize tumours to immune checkpoint blockade. *Nature* *647*, 488–497.
- Steinmetz, B., Smok, I., Bikaki, M., and Leitner, A. (2023). Protein-RNA interactions: from mass spectrometry to drug discovery. *Essays Biochem.* *67*, 175–186.
- Walia, R.R., Xue, L.C., Wilkins, K., El-Manzalawy, Y., Dobbs, D., and Honavar, V. (2014). RNABindRPlus: a predictor that combines machine learning and sequence homology-based methods to improve the reliability of predicted RNA-binding residues in proteins. *PLoS one* *9*, e97725.
- Wu, Q., Peng, Z., Zhang, Y., and Yang, J. (2018). COACH-D: improved protein-ligand binding sites prediction with refined ligand-binding poses through molecular docking. *Nucleic Acids Res.* *46*, W438–W442.
- Li, S., Yamashita, K., Amada, K.M., and Standley, D.M. (2014). Quantifying sequence and structural features of protein-RNA interactions. *Nucleic Acids Res.* *42*, 10086–10098.
- Xia, Y., Xia, C.-Q., Pan, X., and Shen, H.-B. (2021). GraphBind: protein structural context embedded rules learned by hierarchical graph neural networks for recognizing nucleic-acid-binding residues. *Nucleic Acids Res.* *49*, e51.
- Roche, R., Moussad, B., Shuvo, M.H., Tarafder, S., and Bhattacharya, D. (2024). EquipNAS: improved protein-nucleic acid binding site prediction using protein-language-model-informed equivariant deep graph neural networks. *Nucleic Acids Res.* *52*, e27. <https://doi.org/10.1093/nar/gkac039>.
- Hiller, M., Pudimat, R., Busch, A., and Backofen, R. (2006). Using RNA secondary structures to guide sequence motif finding towards single-stranded regions. *Nucleic Acids Res.* *34*, e117.
- Kazan, H., Ray, D., Chan, E.T., Hughes, T.R., and Morris, Q. (2010). RNAcontext: a new method for learning the sequence and structure binding preferences of RNA-binding proteins. *PLoS Comput. Biol.* *6*, e1000832.
- Maticzka, D., Lange, S.J., Costa, F., and Backofen, R. (2014). GraphProt: modeling binding preferences of RNA-binding proteins. *Genome Biol.* *15*, R17–R18.
- Paz, I., Kosti, I., Ares, M., Jr., Cline, M., and Mandel-Gutfreund, Y. (2014). RBPmap: a web server for mapping binding sites of RNA-binding proteins. *Nucleic Acids Res.* *42*, W361–W367.
- Abramson, J., Adler, J., Dunger, J., Evans, R., Green, T., Pritzel, A., Ronneberger, O., Willmore, L., Ballard, A.J., Bambrick, J., et al. (2024). Accurate structure prediction of biomolecular interactions with AlphaFold 3. *Nature* *630*, 493–500. <https://doi.org/10.1038/s41586-024-07487-w>.
- Krishna, R., Wang, J., Ahern, W., Sturmfels, P., Venkatesh, P., Kalvet, I., Lee, G.R., Morey-Burrows, F.S., Anishchenko, I., Humphreys, I.R., et al. (2024). Generalized biomolecular modeling and design with RoseTTAFold All-Atom. *Science* *384*, eadl2528. <https://doi.org/10.1126/science.adl2528>.
- Baek, M., McHugh, R., Anishchenko, I., Jiang, H., Baker, D., and DiMaio, F. (2024). Accurate prediction of protein-nucleic acid complexes using RoseTTAFoldNA. *Nat. Methods* *21*, 117–121.
- Satorras, V.c.G., Hoogeboom, E., and Welling, M. (2021). E(n) Equivariant Graph Neural Networks. In Proceedings of the 38th International Conference on Machine Learning, M. Marina and Z. Tong, eds. (PMLR).
- Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., Kabeli, O., Shmueli, Y., et al. (2023). Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science* *379*, 1123–1130.
- Chen, J., Hu, Z., Sun, S., Tan, Q., Wang, Y., Yu, Q., Zong, L., Hong, L., Xiao, J., Shen, T., et al. (2022). Interpretable RNA Foundation Model from Unannotated Data for Highly Accurate RNA Structure and Function Predictions. Preprint at arXiv. <https://doi.org/10.48550/arXiv.2204.00300>.
- Shen, T., Hu, Z., Peng, Z., Chen, J., Xiong, P., Hong, L., Zheng, L., Wang, Y., King, I., and Wang, S. (2022). E2Efold-3D: end-to-end deep learning method for accurate de novo RNA 3D structure prediction preprint at arXiv. <https://doi.org/10.48550/arXiv.2207.01586>.
- Si, Y., and Yan, C. (2021). Improved protein contact prediction using dimensional hybrid residual networks and singularity enhanced loss function. *Briefings Bioinf.* *22*, bbab341. <https://doi.org/10.1093/bib/bbab341>.
- Chaudhury, S., Lyskov, S., and Gray, J.J. (2010). PyRosetta: a script-based interface for implementing molecular modeling algorithms using Rosetta. *Bioinformatics* *26*, 689–691.
- Suzek, B.E., Wang, Y., Huang, H., McGarvey, P.B., and Wu, C.H.; UniProt Consortium (2015). UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics* *31*, 926–932.
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library preprint at arXiv. <https://doi.org/10.48550/arXiv.1912.01703>.
- DeLano, W.L. (2002). Pymol: An open-source molecular graphics tool. *CCP4 Newsl. protein crystallogr* *40*, 82–92.
- Hunter, J.D. (2007). Matplotlib: A 2D graphics environment. *Comput. Sci. Eng.* *9*, 90–95.