UNIVERSITY OF CAMBRIDGE

# gRNAde

# Geometric deep learning for 3D RNA inverse design

**Chaitanya K. Joshi**, Arian R. Jamasb, Ramon Viñas, Charles Harris, Simon Mathis, Alex Morehead, Rishabh Anand, Pietro Liò

*Computational Biology Workshop, International Conference on Machine Learning, 2023*

Forthcoming book chapter in *Methods in Molecular Biology (RNA Design: Methods and Protocols)*

**Paper**: https://arxiv.org/abs/2305.14749

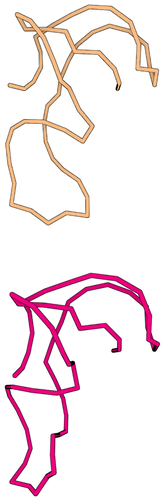
**Codebase**: github.com/chaitjo/geometric-rna-design

# Executive summary

## Inverse design of RNA sequence conditioned on backbone structure

RNA Conformational Ensemble

Extract Backbones



Set of Backbone Geometric Graphs

Multi-state Graph Neural Network Encoder

Equivariant to:

- 3D roto-translations
- node order
- conformation order

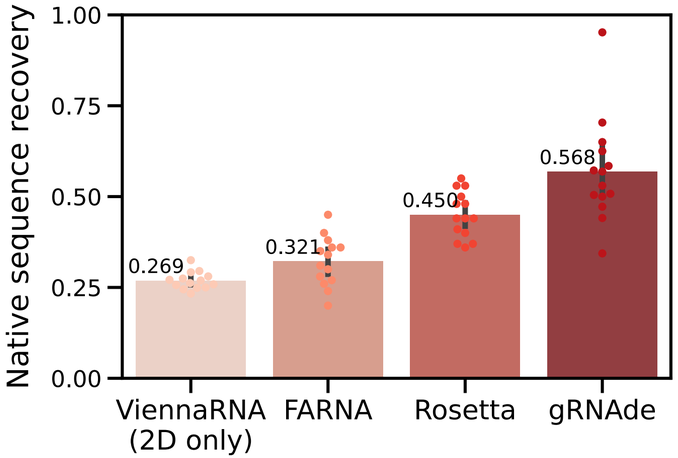
Fixed backbone re-design

Sequence Decoder

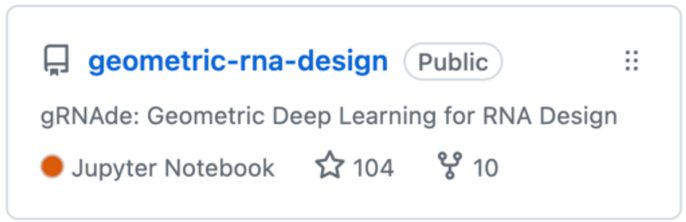
GAGCGU...

RNA Sequence

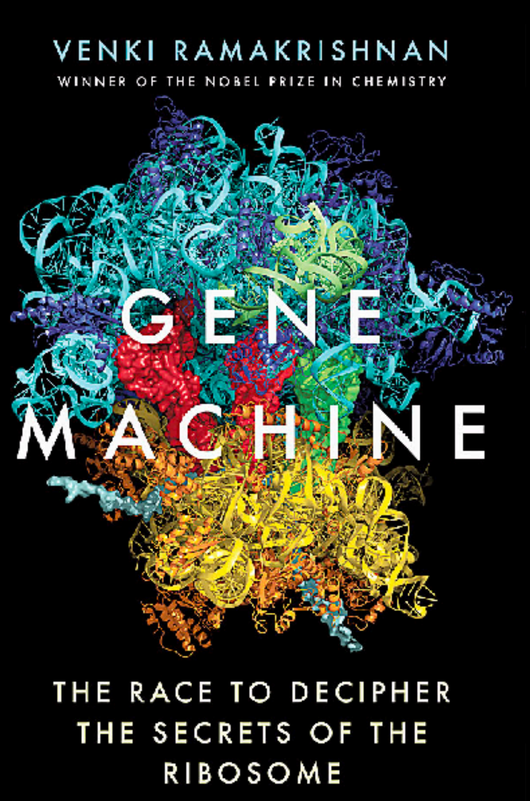
gRNAde improves **sequence recovery** and **speed** compared to Rosetta fixed-backbone RNA design

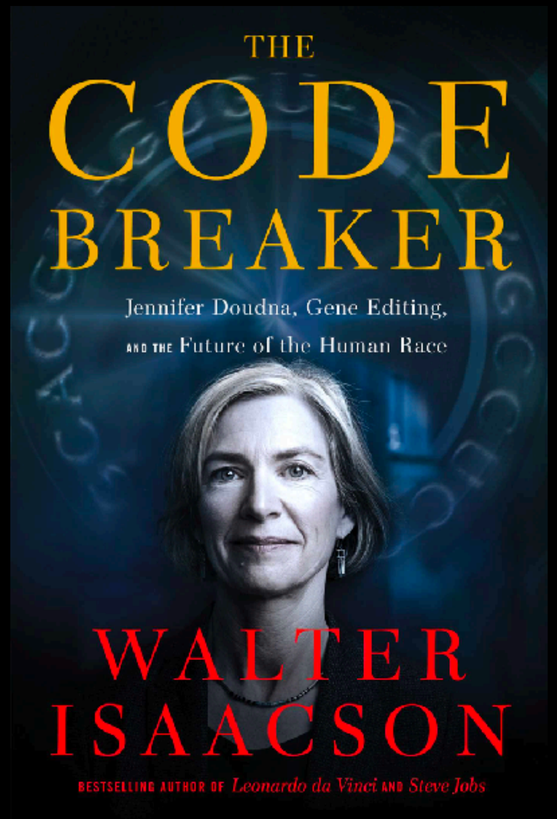


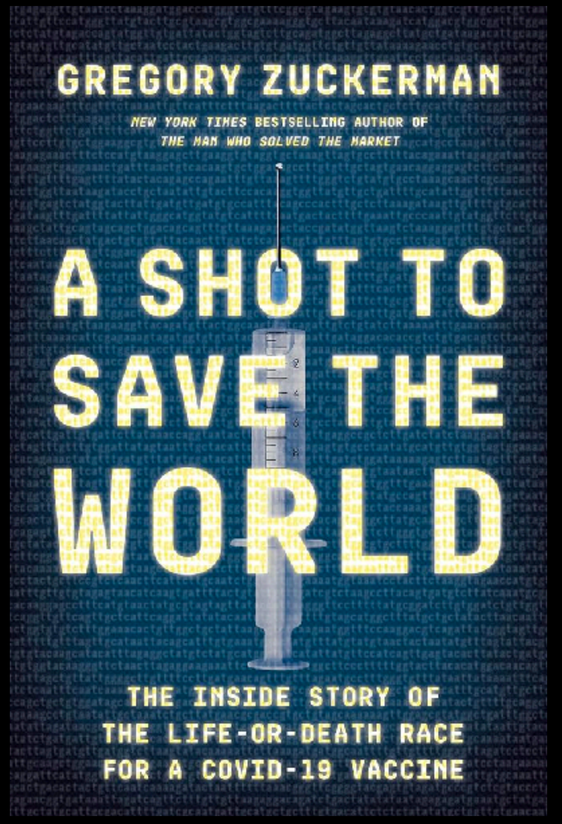

**ProteinMPNN** for RNA.
Open-source on GitHub:



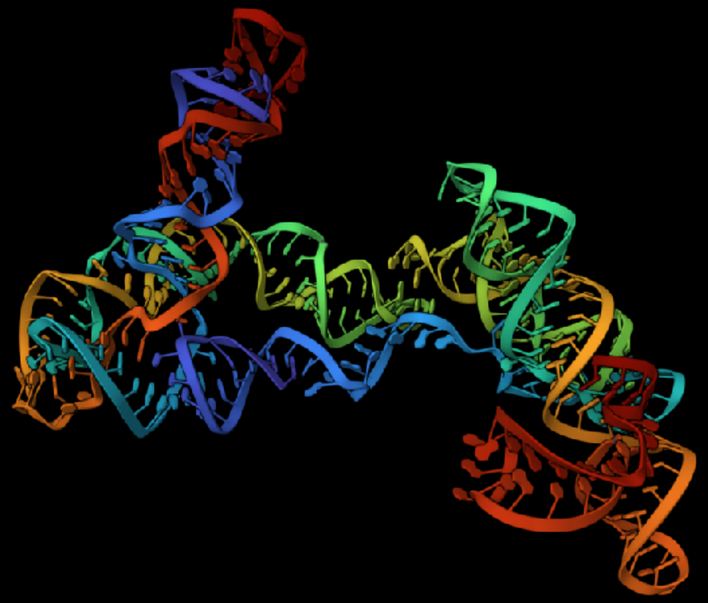
# RNA at the forefront of biotechnology



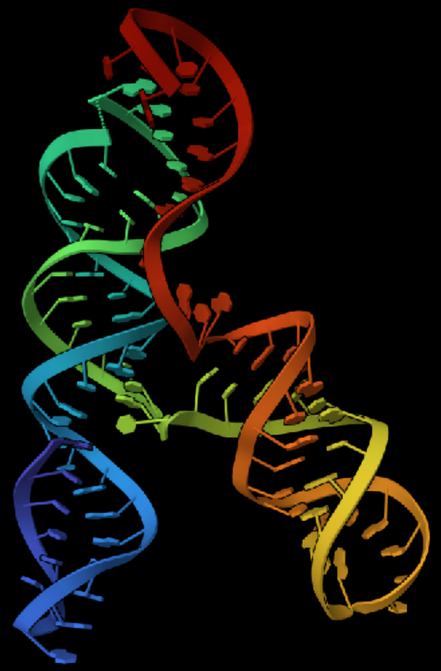



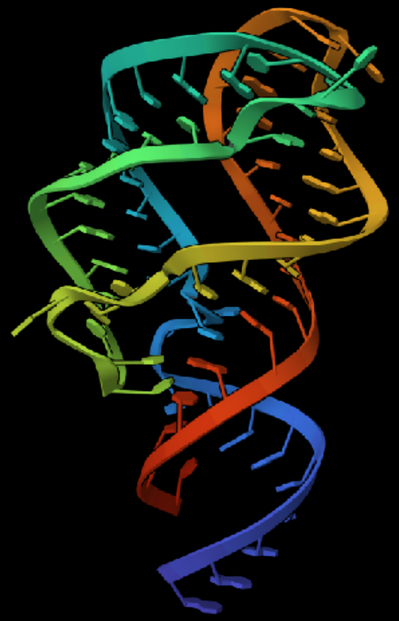
# And many RNA are structured



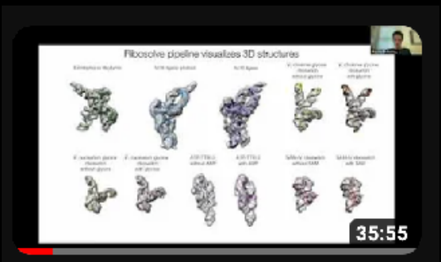
RNA polymerase
ribozyme
8T2P
McRae et al.



SARS-CoV-2
frameshift
element
6XRZ
Zhang et al.



Adenine
riboswitch
aptamer
5E54
Stagno et al.



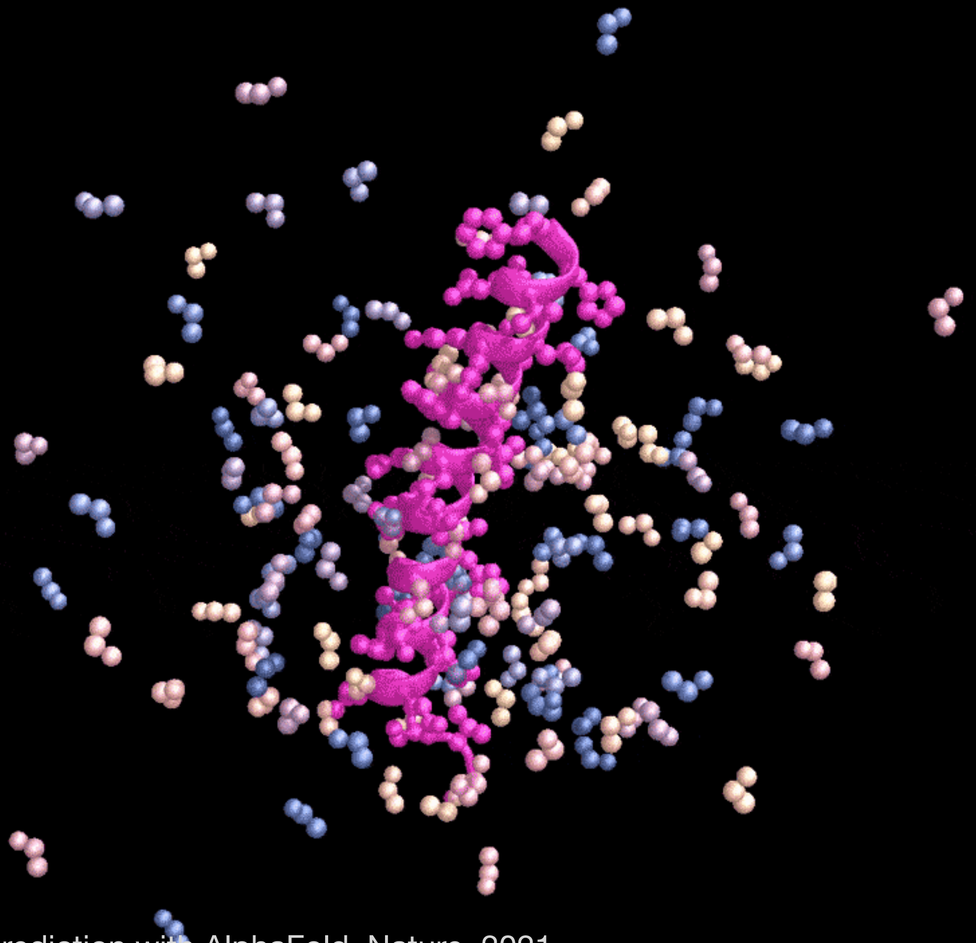

NGBS2022 Talk 10: RNA modelling and design - Rhiju Das

466 views • 4 months ago

# Meanwhile

## Generative models can design bespoke protein structure & function!

Prescient Design
A Genentech Accelerator



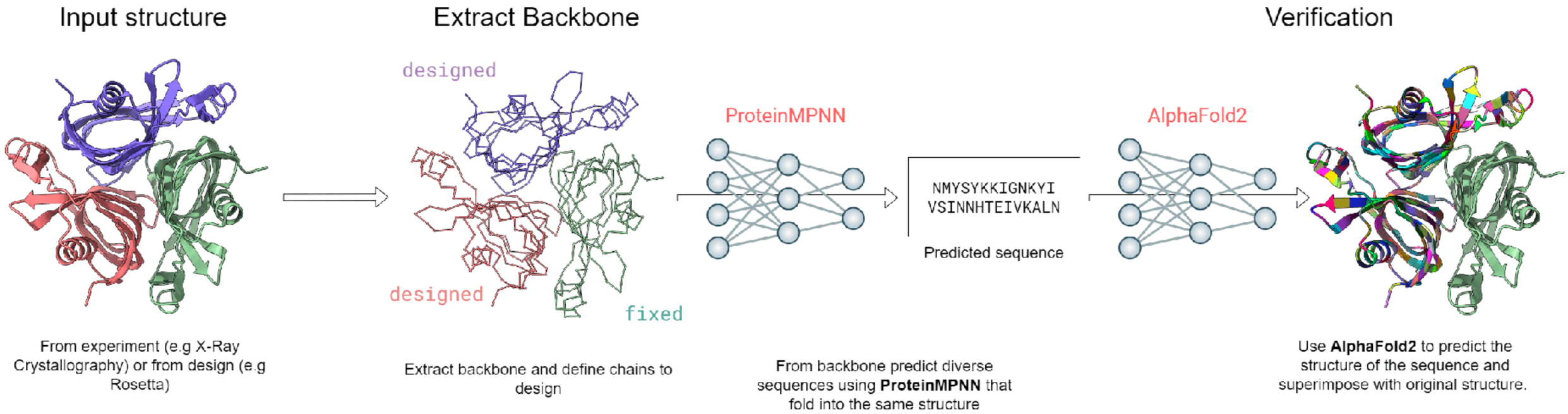
What about RNA?

Jumper et al. Highly accurate protein structure prediction with AlphaFold. Nature, 2021.
Dauparas et al. Robust deep learning–based protein sequence design using ProteinMPNN. Science, 2022.
Watson, Juergens et al. De novo design of protein structure and function with RFdiffusion. Nature, 2023.

‘Generative AI’
is starting to work for protein design

# Structure-based protein design workflow

## Assumption: Structure → Function




Not shown: protein Language Models (purely sequence-based)

Dauparas et al. Robust deep learning–based protein sequence design using ProteinMPNN. Science. 2022.
Figure: Simon Duerr

# Analogy to ChatGPT

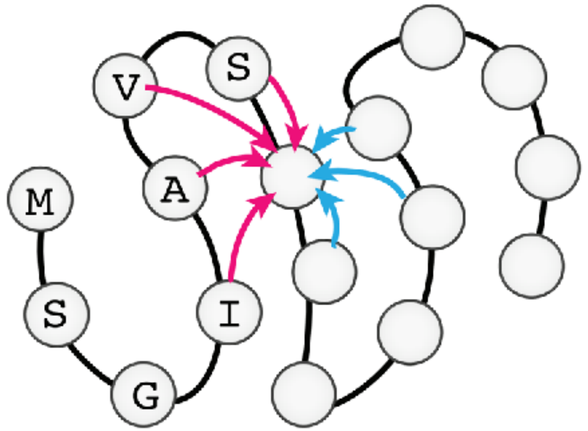
Natural language models

S = Where are we going

Previous words (Context)

Word being predicted

$$P(S) = P(\text{Where}) \times P(\text{are} \mid \text{Where}) \times P(\text{we} \mid \text{Where are}) \times P(\text{going} \mid \text{Where are we})$$




**Trained on PDB structures:**

Samples are biased towards thermal stability, expression.

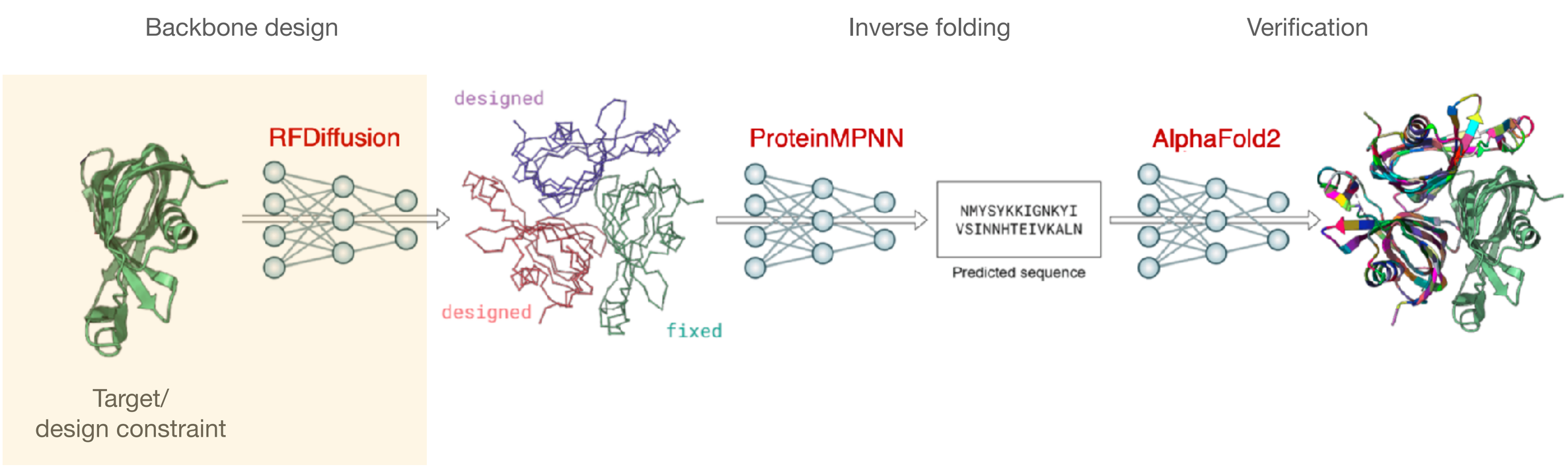

Sequence generation: **Language model**

Sequence generation conditioned on structure: **ProteinMPNN** (inverse folding)

Ingraham et al. Generative Models for Graph-Based Protein Design. NeurIPS, 2019.

# De-novo protein design workflow

## Starting from scratch

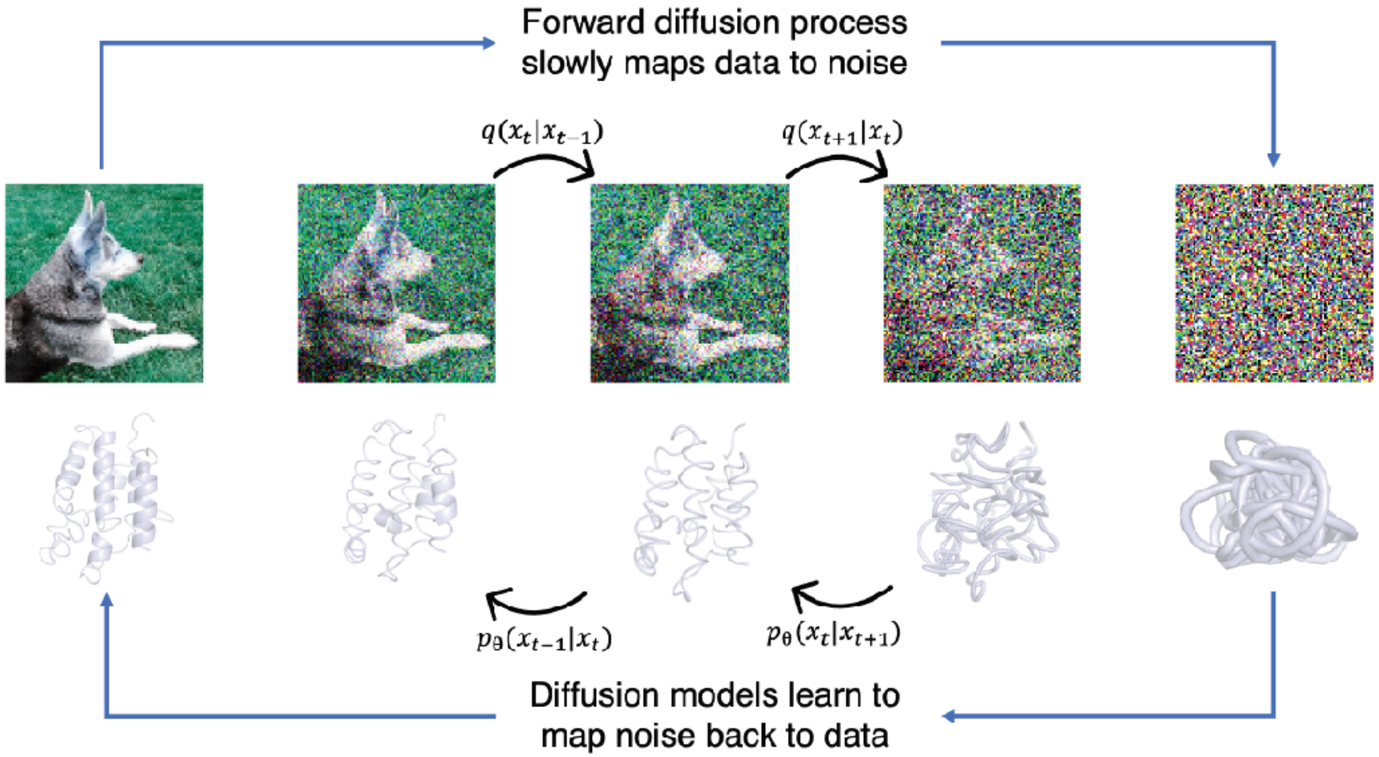



Watson, Juergens et al. De novo design of protein structure and function with RFdiffusion. Nature. 2023.

# Analogy to DALL-E

DALL·E

Image generation models




**Trained on PDB structures:**

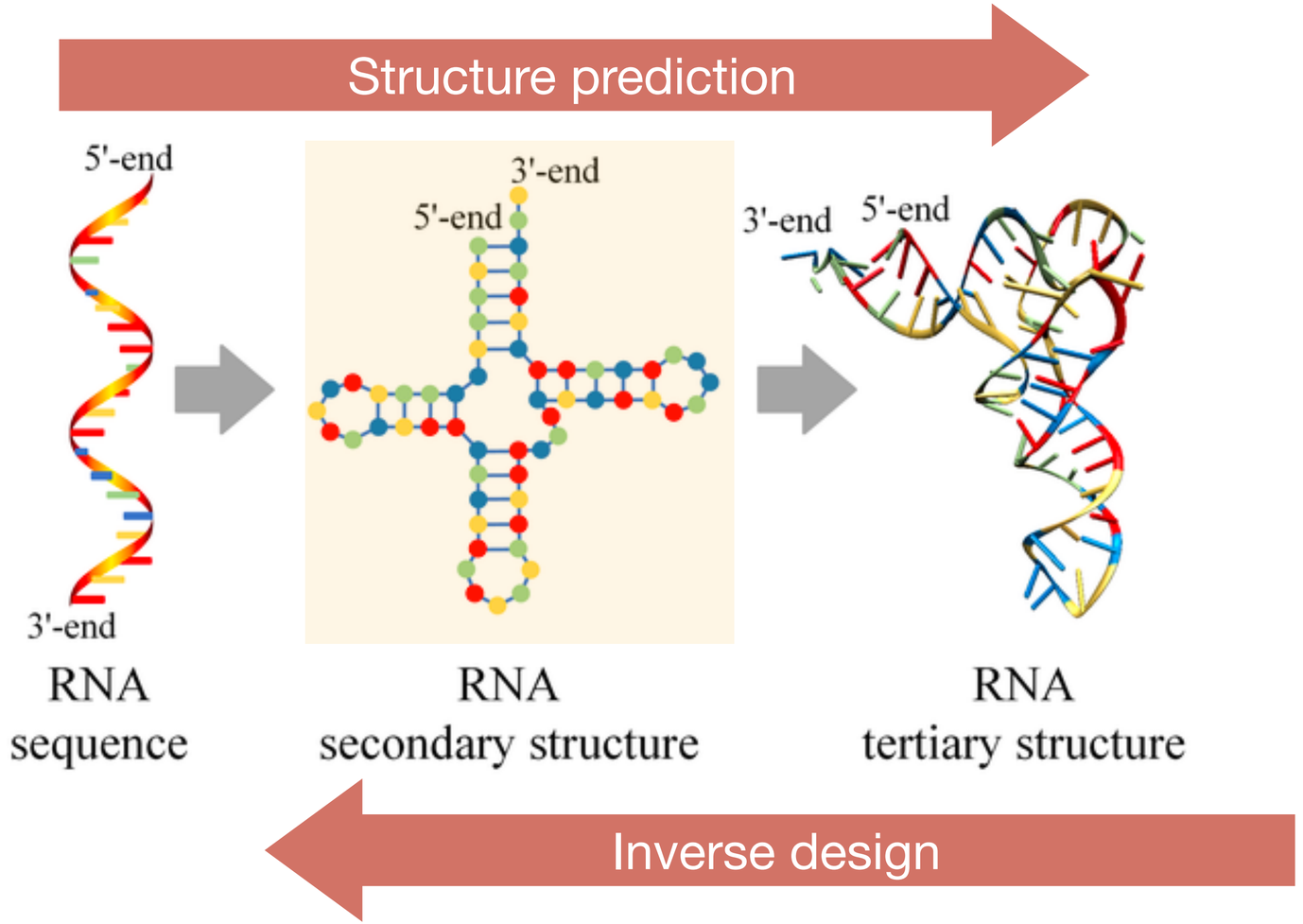
Learn to mix and match sub-structures.

Backbone design: **RFdiffusion**

# What about RNA?

# RNA structure modelling and design

## Emphasis on secondary structure



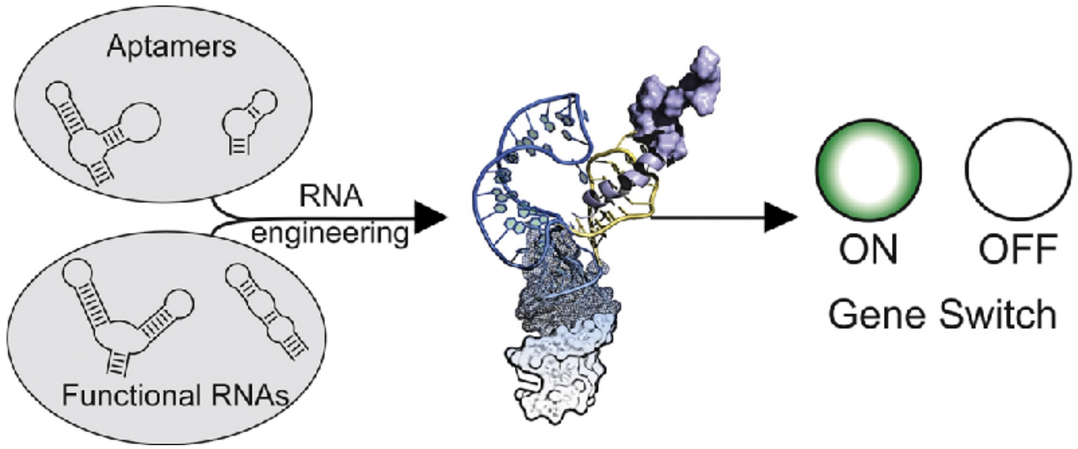

Thoughts on how to think (and talk) about RNA structure Quentin Vicens and Jeffrey S. Kieft. PNAS. 2022.

# Relatively fewer tools for 3D design

## Potential application: aptamers, riboswitches, ribozymes

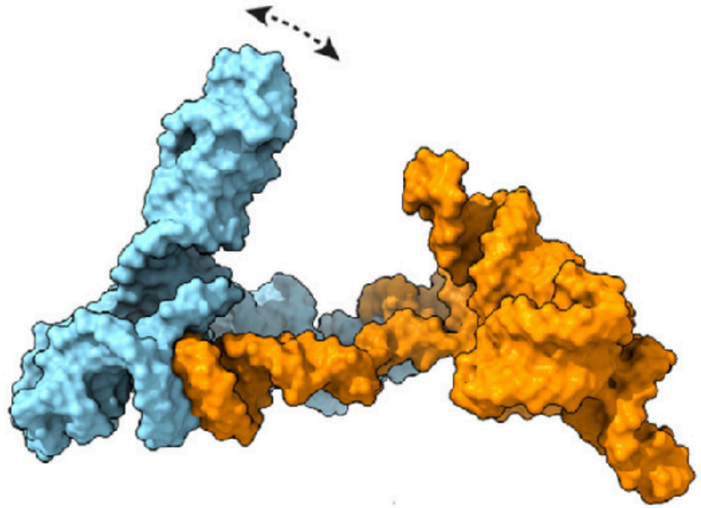
### Transient gene expression

Designing riboswitches in mRNAs




### RNA world

Self-replicating ribozymes

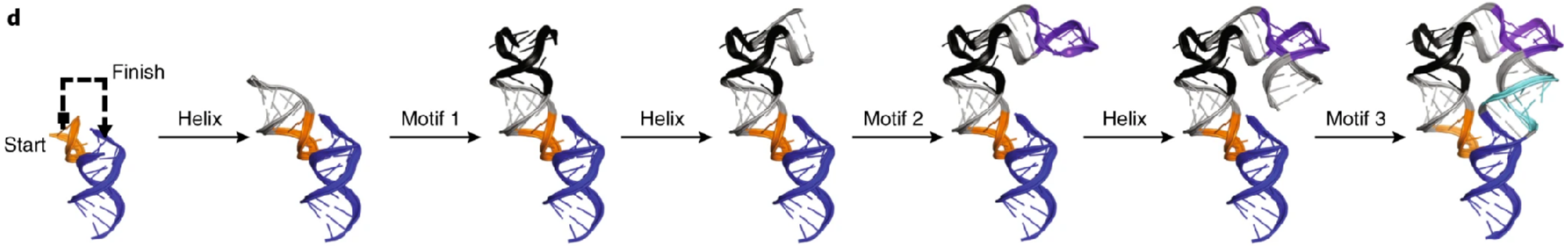


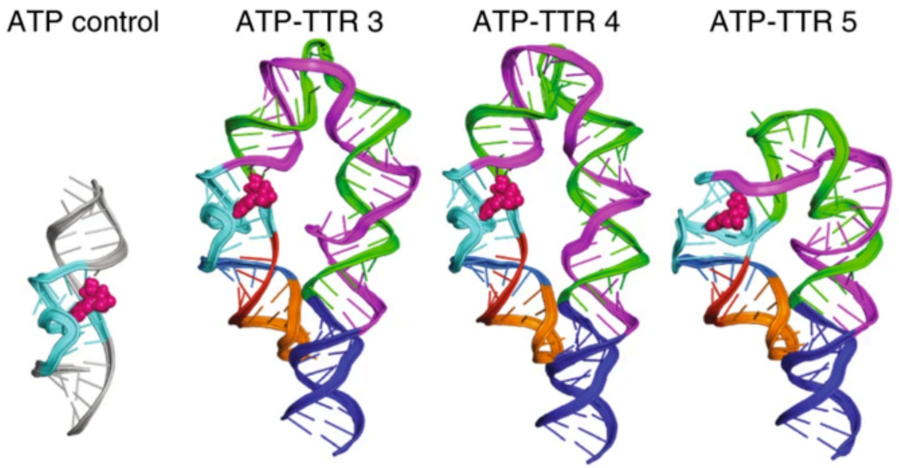
RNAs: carriers of information + play functional roles

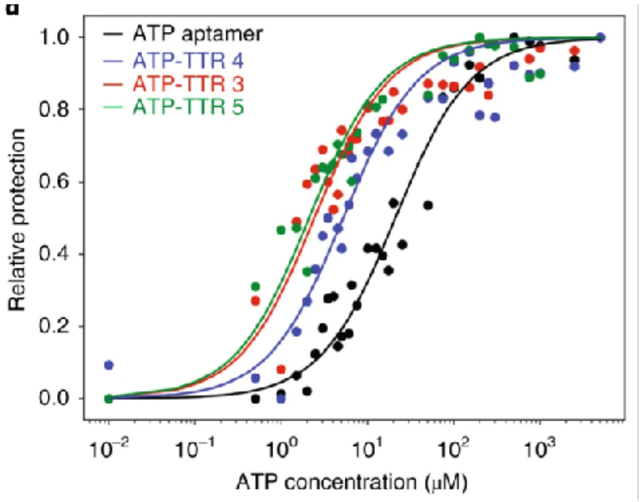
Synthetic RNA-based switches for mammalian gene expression control. Ausländer and Fussenegger. Curr. Opin. Biotechnology. 2017.
McRae et al. Cryo-EM structure and functional landscape of an RNA polymerase ribozyme. PNAS, 2024.

# RNAMake

## Uses classical algorithms for alignment between RNA motifs



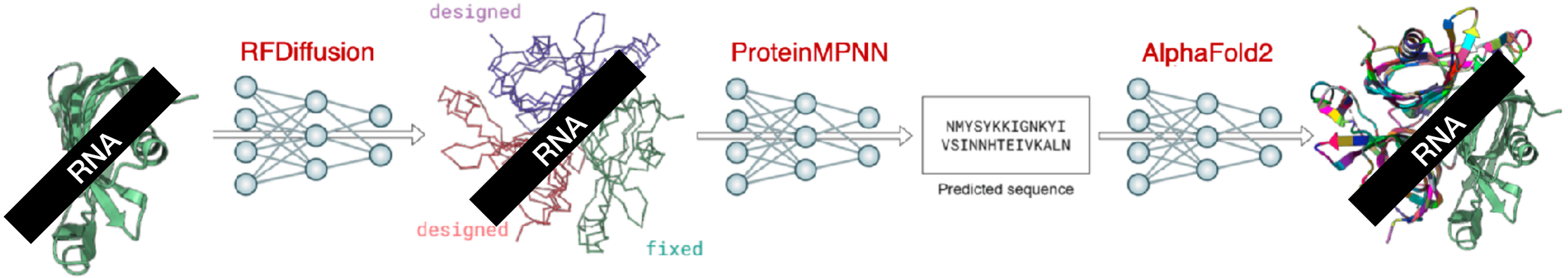




Yesselman et al. Computational Design of Three-dimensional RNA Structure and Function. Nature Nanotechnology, 2020.

# Deep learning toolkit for RNA design

## …work in progress




Nothing public yet

RNAMake, RNA origami (non-DL)

**gRNAde**

**This talk!**

DRFold, RhoFold, RF-NA

Several teams working on this.

Not shown: RNA Language Models — Several teams working on this.

eg. RiNaLMo

# Towards deep learning:
# What data exists?

# Geometric Deep Learning for RNA

## Main challenge: paucity of 3D structural data

"trained with only 18 known RNA structures"

> **ARES: Geometric deep learning of RNA structure.** *Science, 2021.*
>
> Raphael JL Townshend, Stephan Eismann, Andrew M Watkins, Ramya Rangan, Maria Karelina, Rhiju Das, and Ron O Dror.

"trained on 2,986 RNA chains, non-redundant to 122 test RNAs"

> **DRFold: Integrating end-to-end learning with deep geometrical potentials for ab initio RNA structure prediction.** *Nature Communications, 2023.*
>
> Yang Li, Chengxin Zhang, Chenjie Feng, Robin Pearce, Peter L. Freddolino, Yang Zhang.

# All RNA structures in the PDB

## RNAsolo: cleaned, PDB-derived RNA 3D structures

| | Solo RNAs | RNAs from protein–RNA complexes | RNAs from DNA–RNA hybrids | All RNAs |
|---|---|---|---|---|
| Total (today) | 2387 | 13218 | 136 | 15741 |

**3825** equivalence classes

vs.

ProteinMPNN, RFdiffusion: entire PDB
**208,659 proteins** ≤3.5Å → **25,361 clusters** at 30% seq.id.

One order of magnitude more proteins!

# Should we just wait?

## Not necessarily…

Other successful (in-silico) tools were trained on carefully chosen subsets:

- Chroma: 28819 structures ≤2.6Å
- Genie: 8766 domains
- FrameFlow: 3938 domains

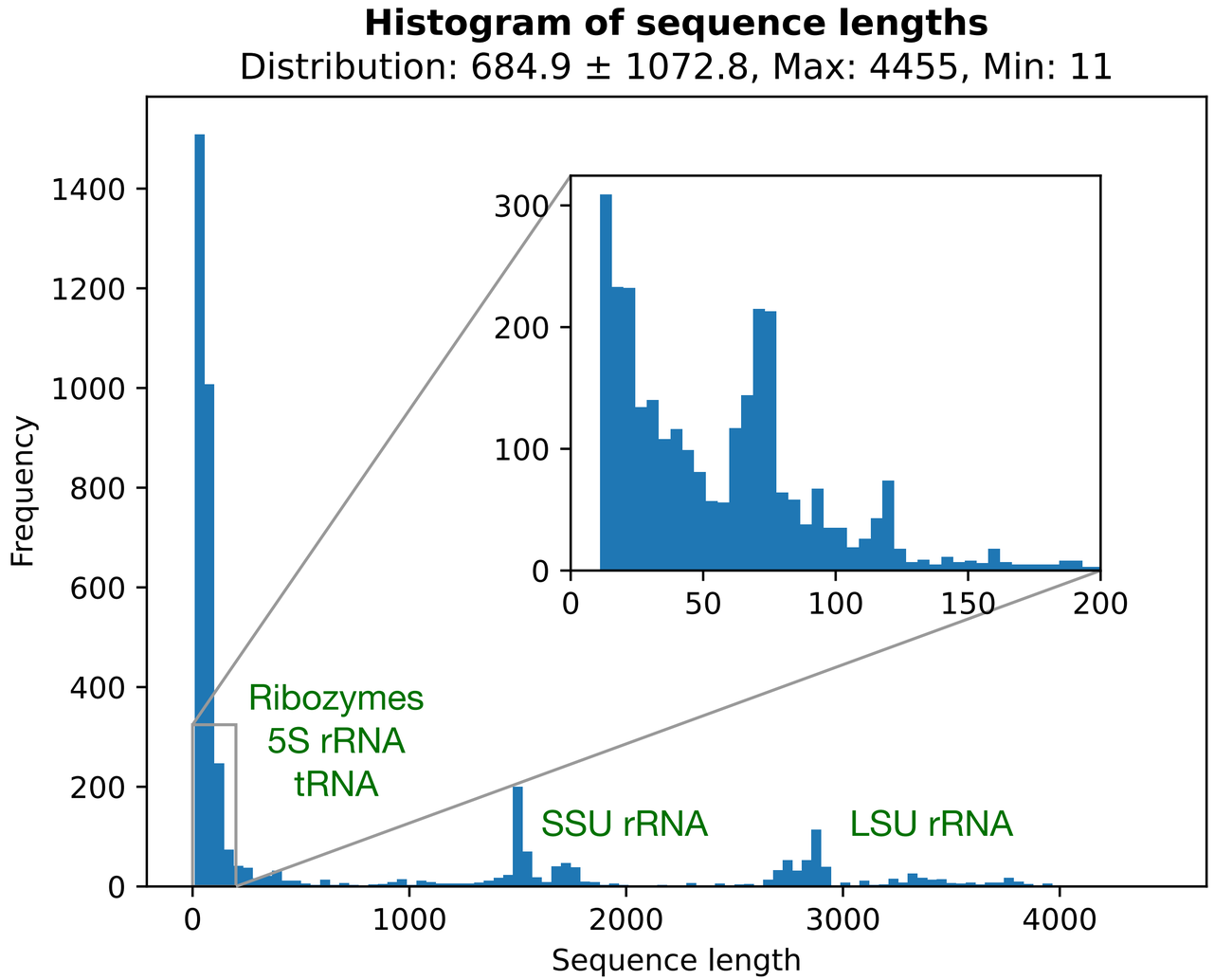
"...achieve similar in-silico performance to RFdiffusion with a quarter of the parameters – an important consideration…**models are often run tens of thousands of times**…"

– *Winnifrith et al. 2023.*

Winnifrith, Outeiral, Hie. Generative artificial intelligence for de novo protein design. 2023.

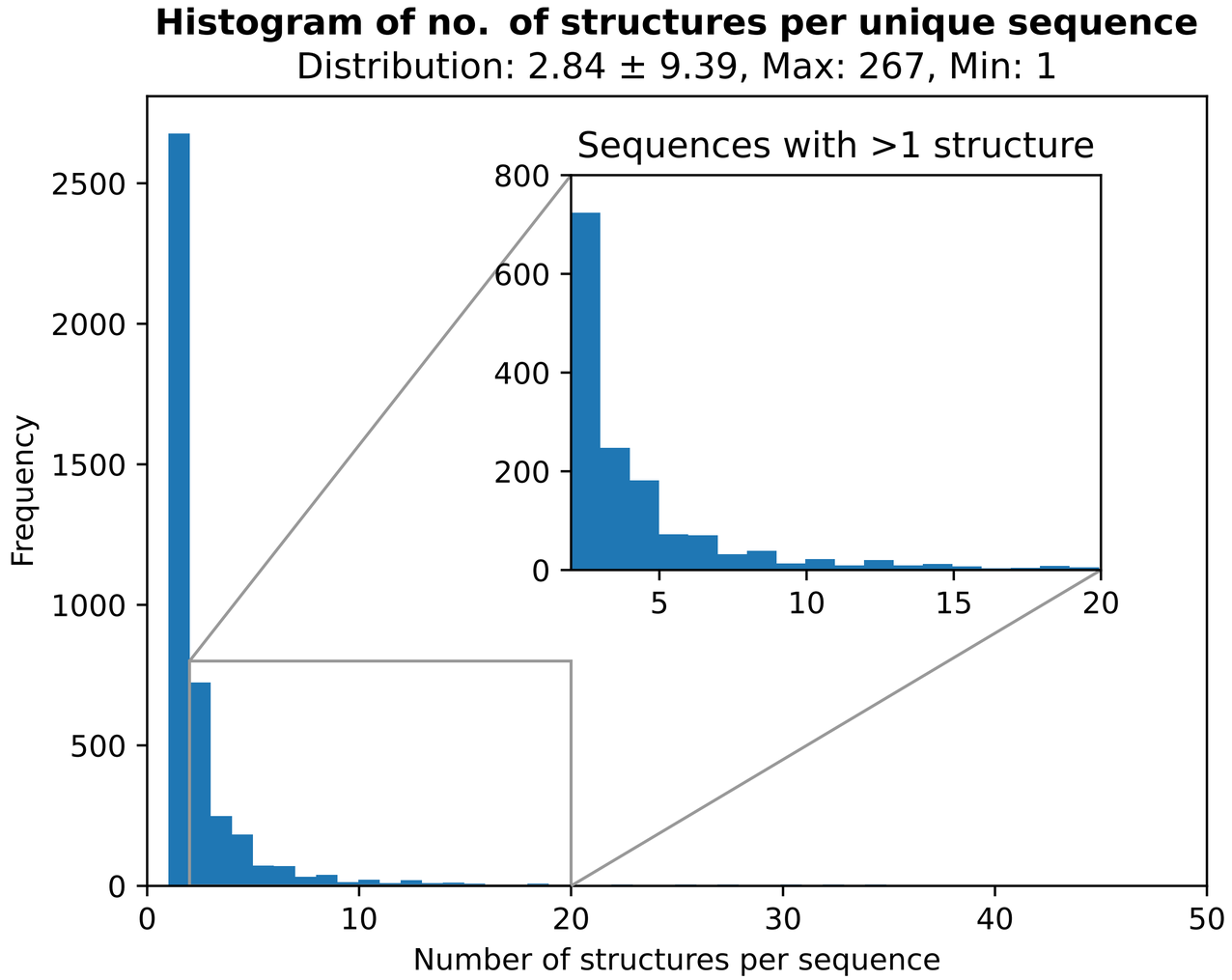
# Distribution of sequence lengths

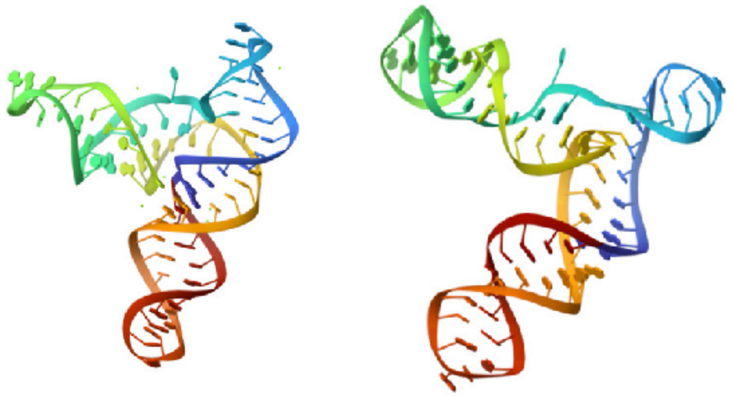
## Mostly shorter than 500 nucleotides



# Many RNAs have multiple structures

## Multiple conformations are important for functionality

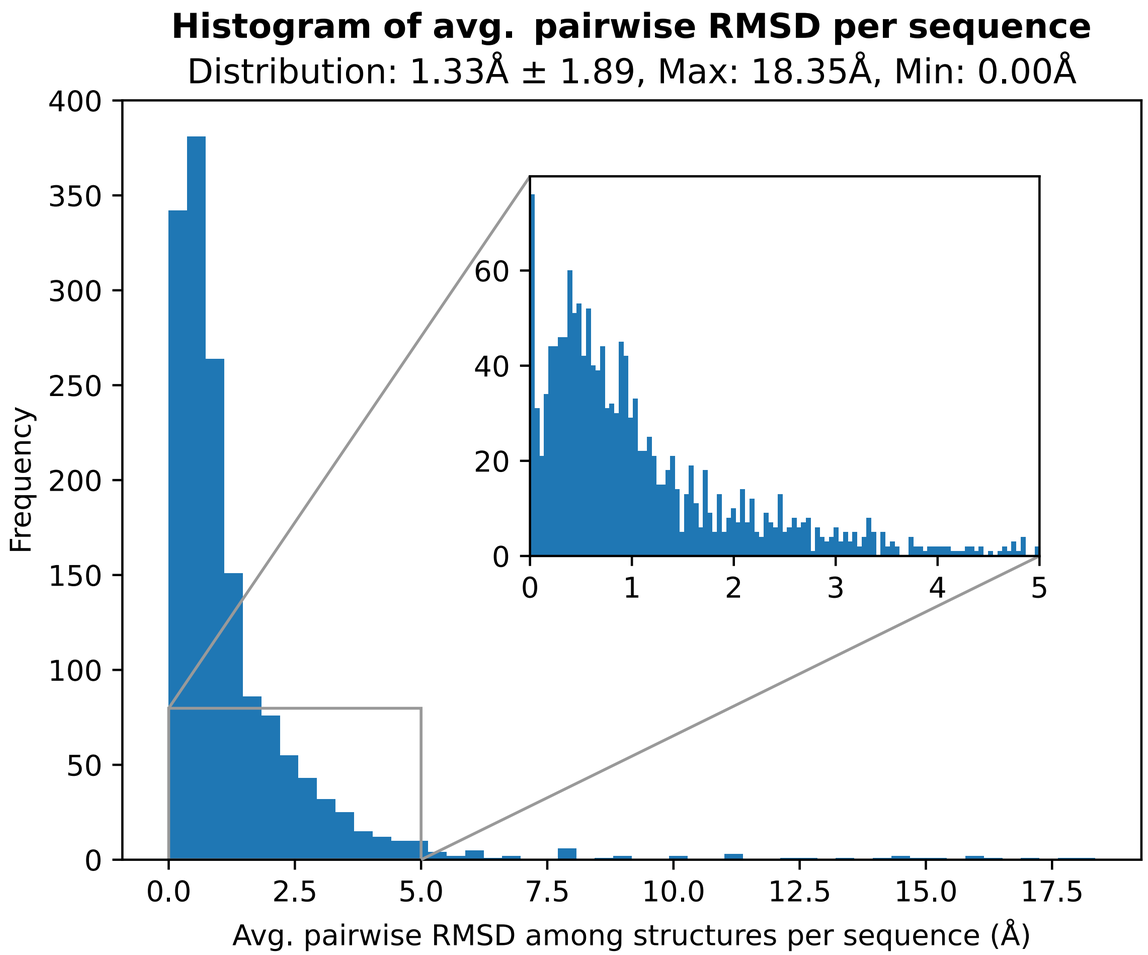


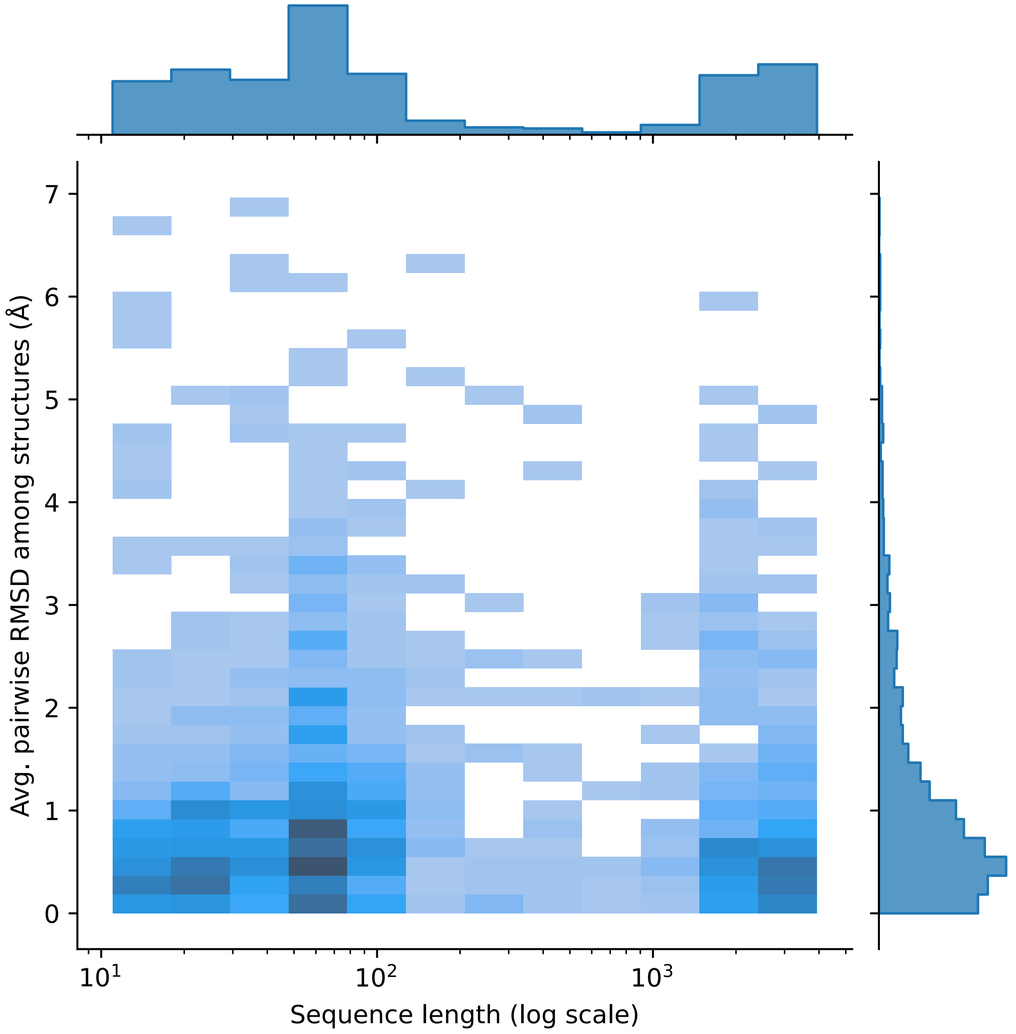



L1 ligase ribozyme:
**~15Å**
(PDB 2OIU)

# High RMSDs between multiple states

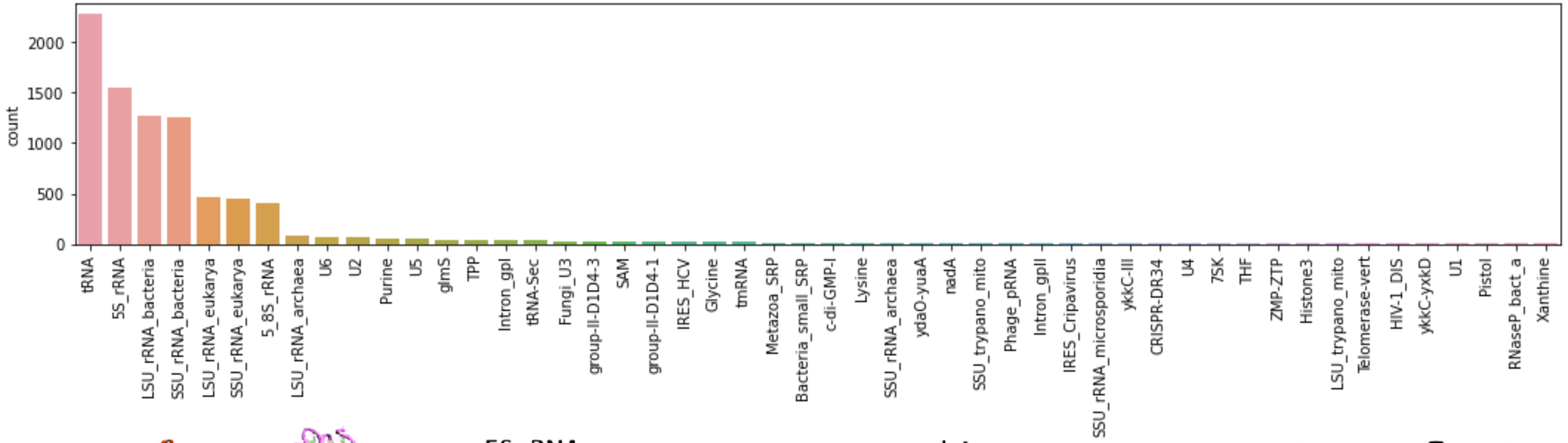
## Same sequence can have very different structures

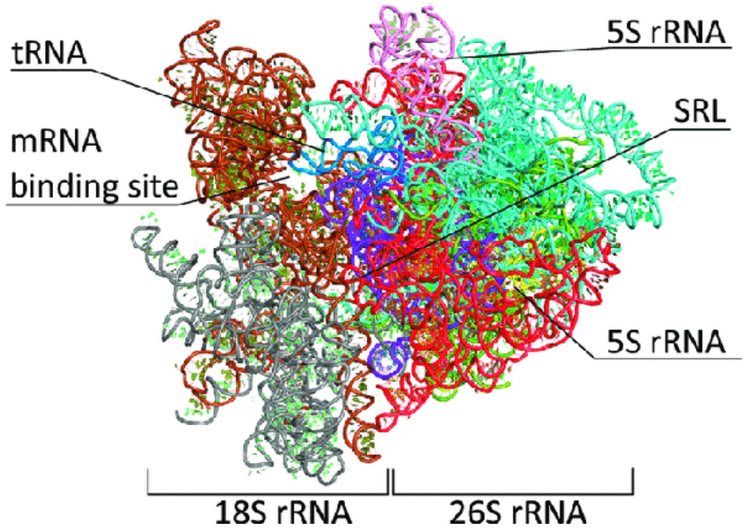


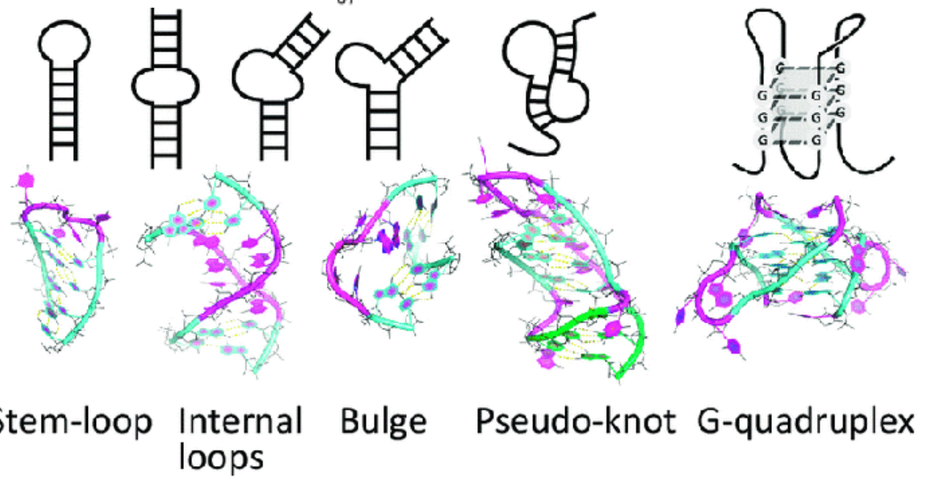



# RFam families in the PDB

## Majority from protein-RNA complexes, tRNAs, ribosomal RNAs



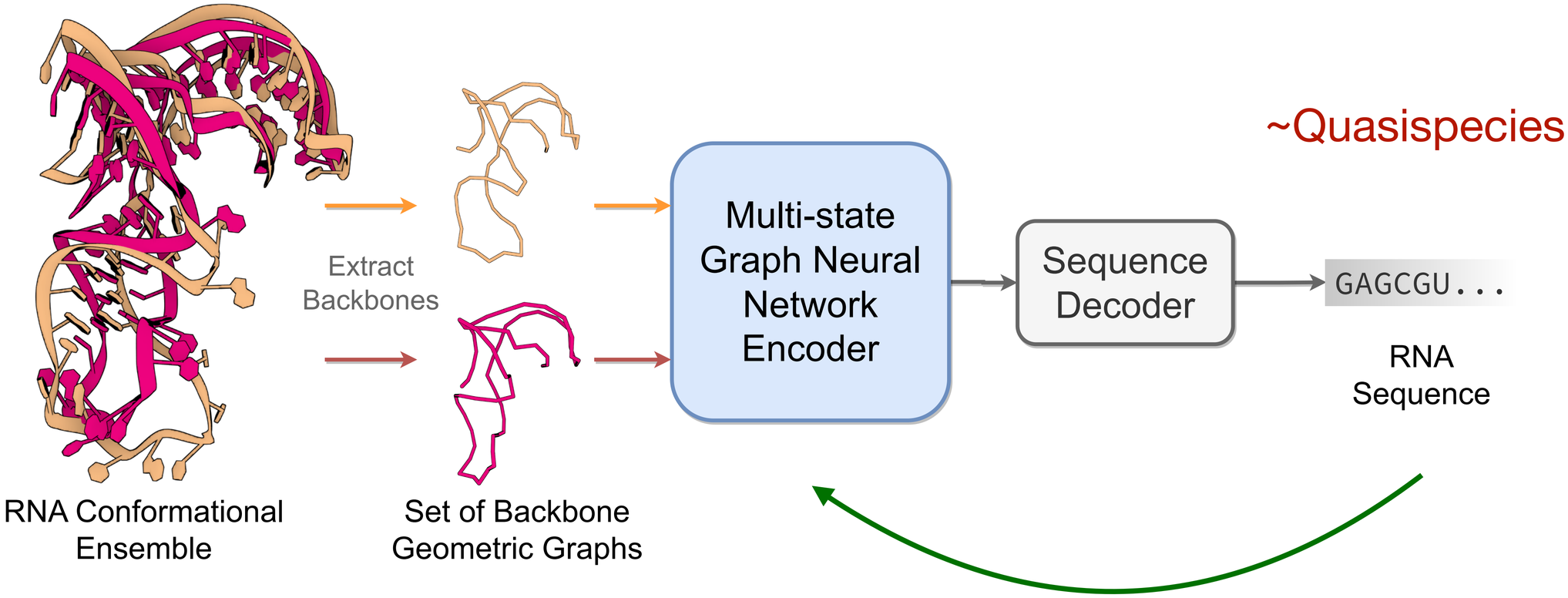



# The gRNAde pipeline for RNA inverse folding

# Fixed backbone re-design

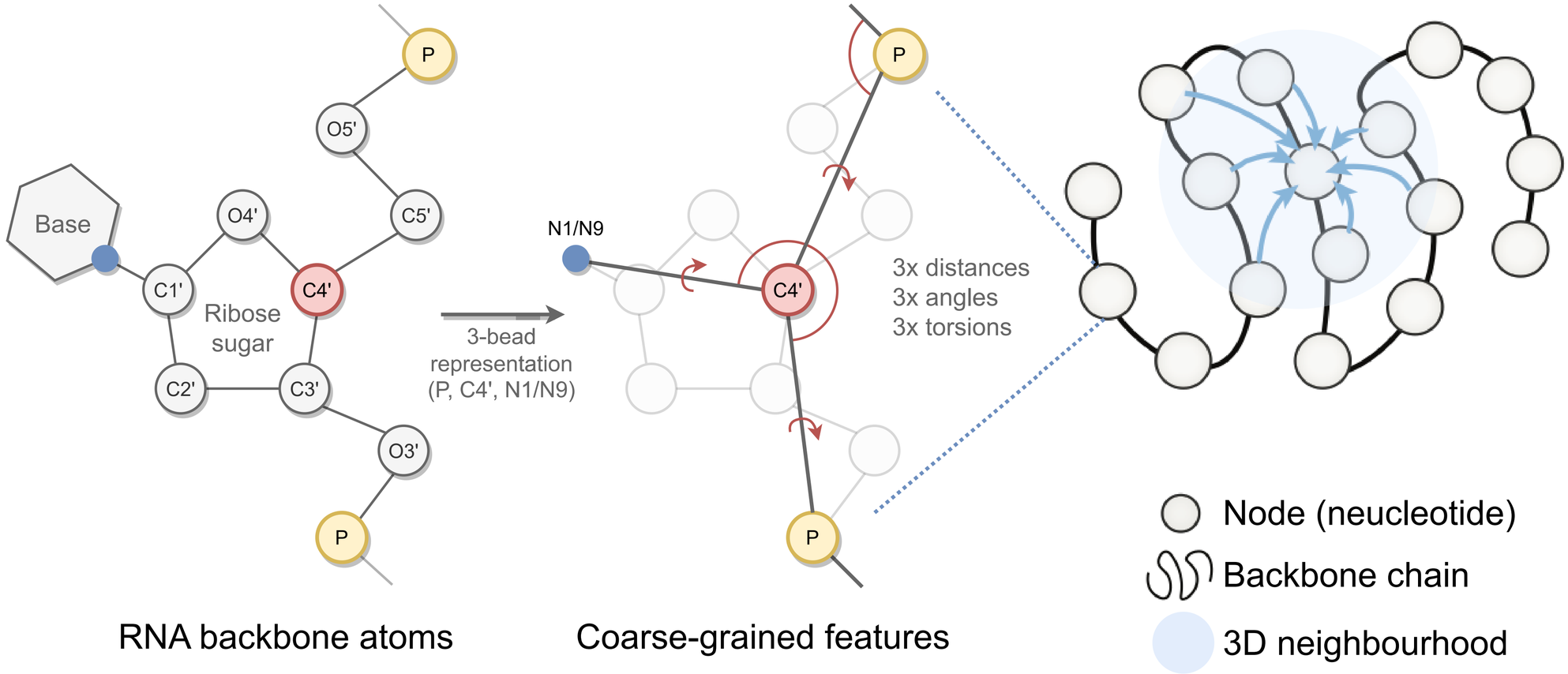
## Input: native PDB file → Output: designed sequences




**Self-supervised learning:**
(backbone, sequence) pairs from PDB

# RNA backbones as 3D graphs

## Preparing input: PDB file(s) → geometric graph in 3D

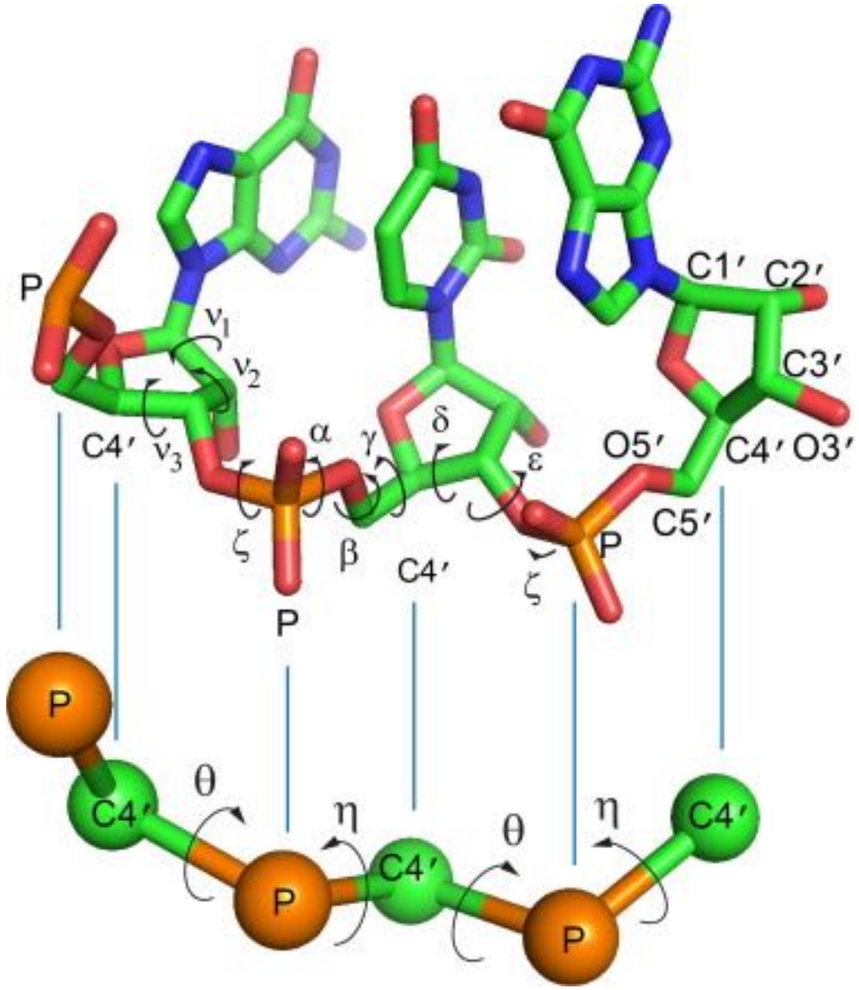



RNA backbone atoms

Coarse-grained features

# Why the 3-bead representation?

## P, C4', N1 (pyrimidine) or N9 (purine)



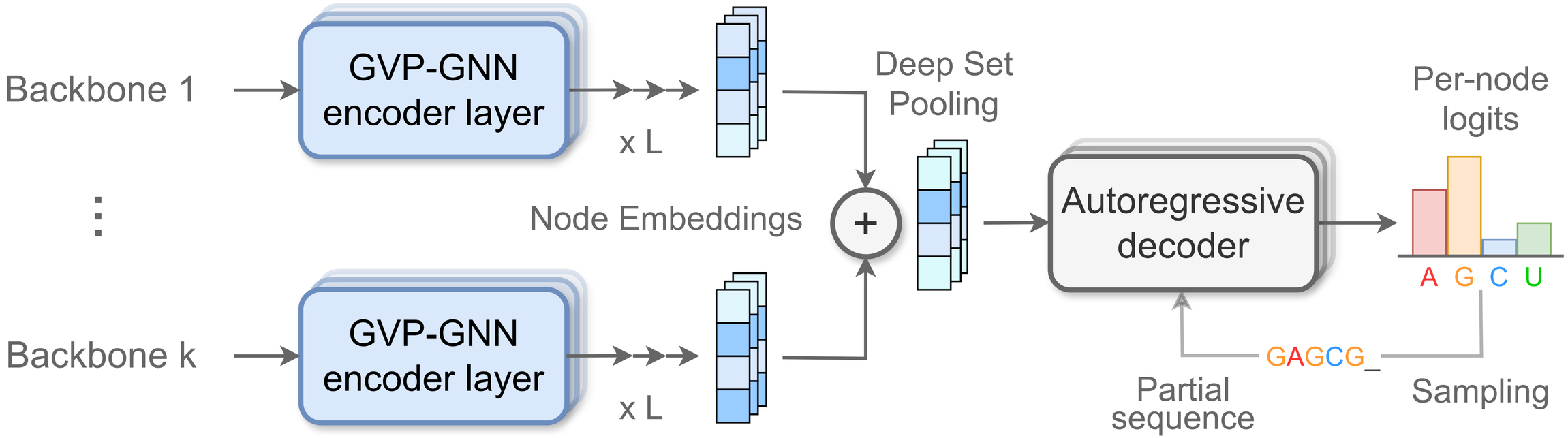

Intuition: Reduce the degrees of freedom as input to gRNAde.

*"The pseudotorsional descriptors η and θ, together with sugar pucker,* ***are sufficient to describe RNA backbone conformations fully*** *in most cases."*

Coarse-grained modeling of RNA 3D structure. Dawson et al. Methods. 2016.

# gRNAde model architecture

**One or more featurized graphs → per-node probability over 4 bases**

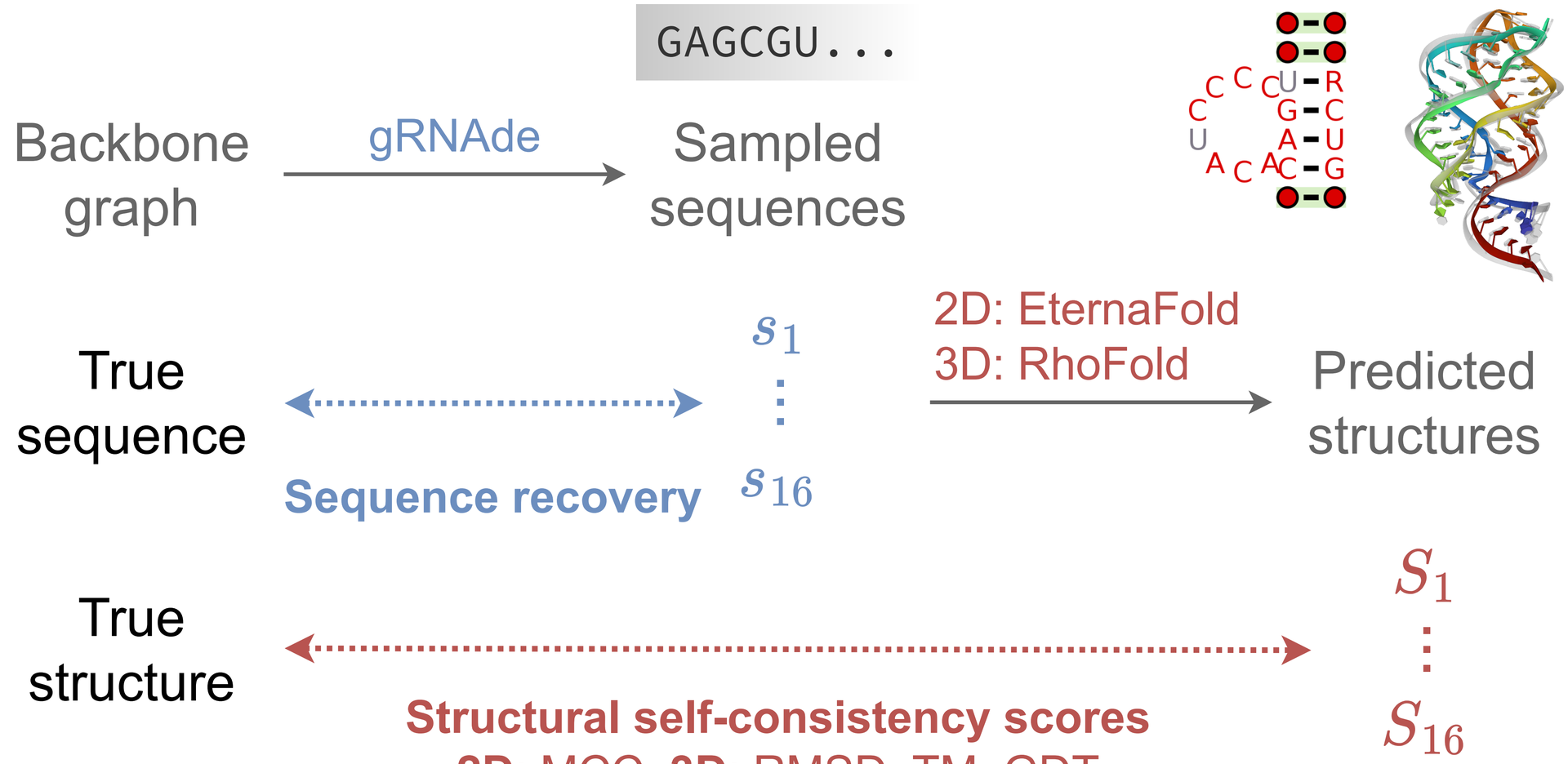




Jing et al. Learning from Protein Structure with Geometric Vector Perceptrons. ICLR 2020.
Zaheer et al. Deep sets. NeurIPS 2017.

# What is a good designs?

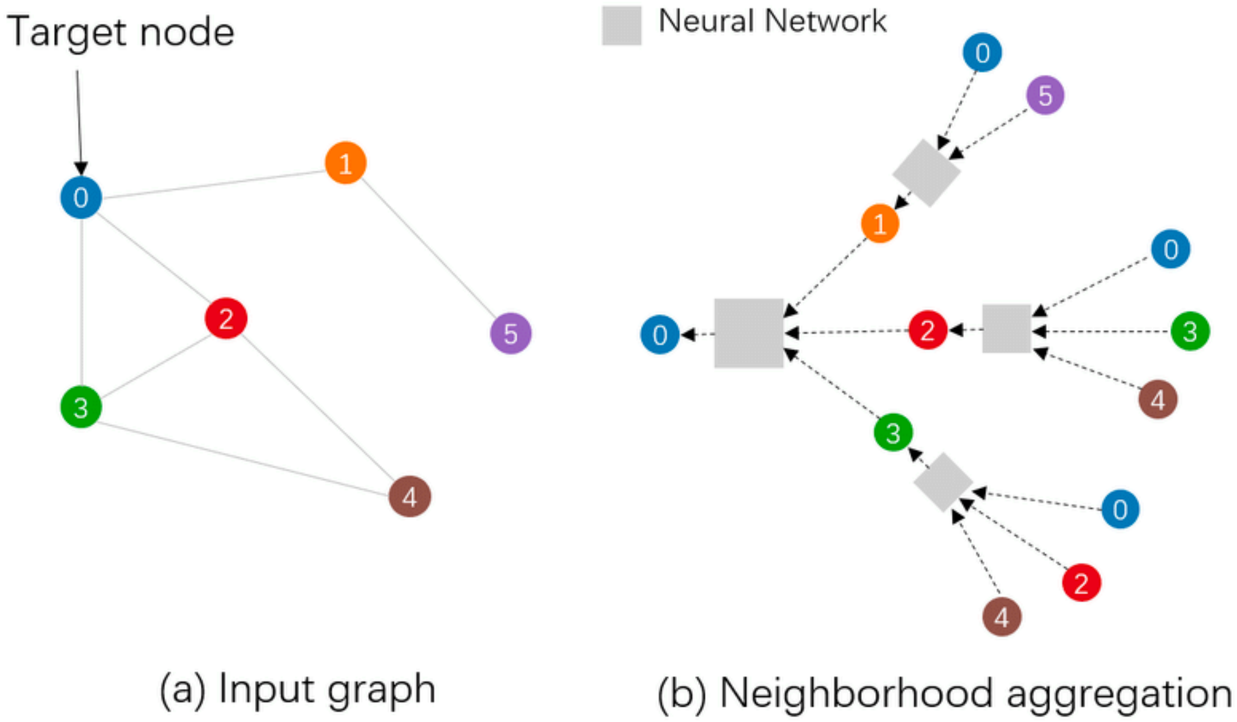
## In-silico evaluation metrics to prioritise designs



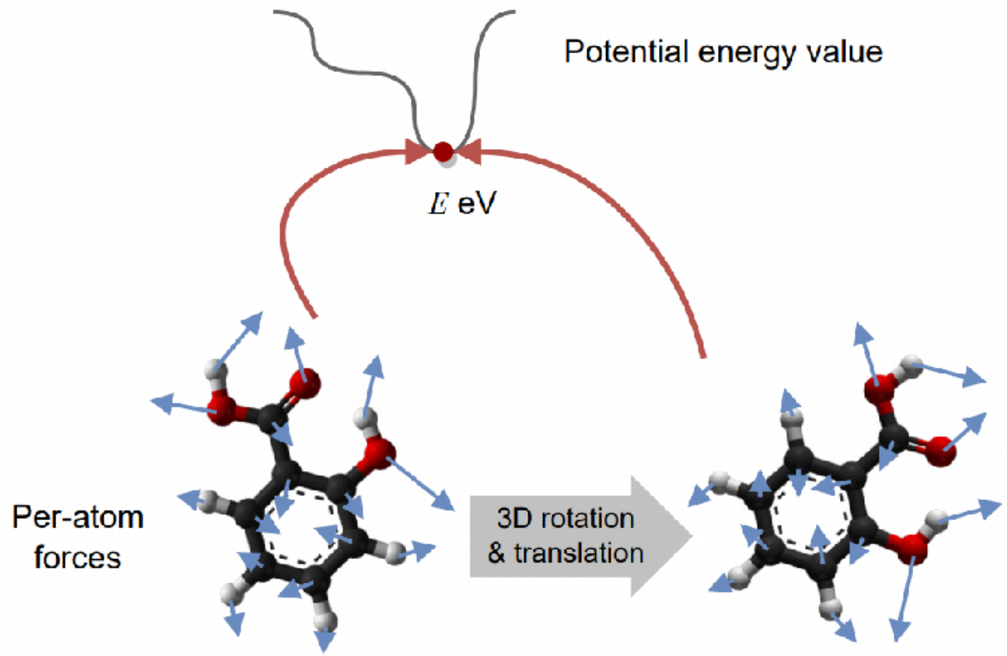
# Graph Neural Networks for 3D structure

Extra

## Learn to propagate information along the graph



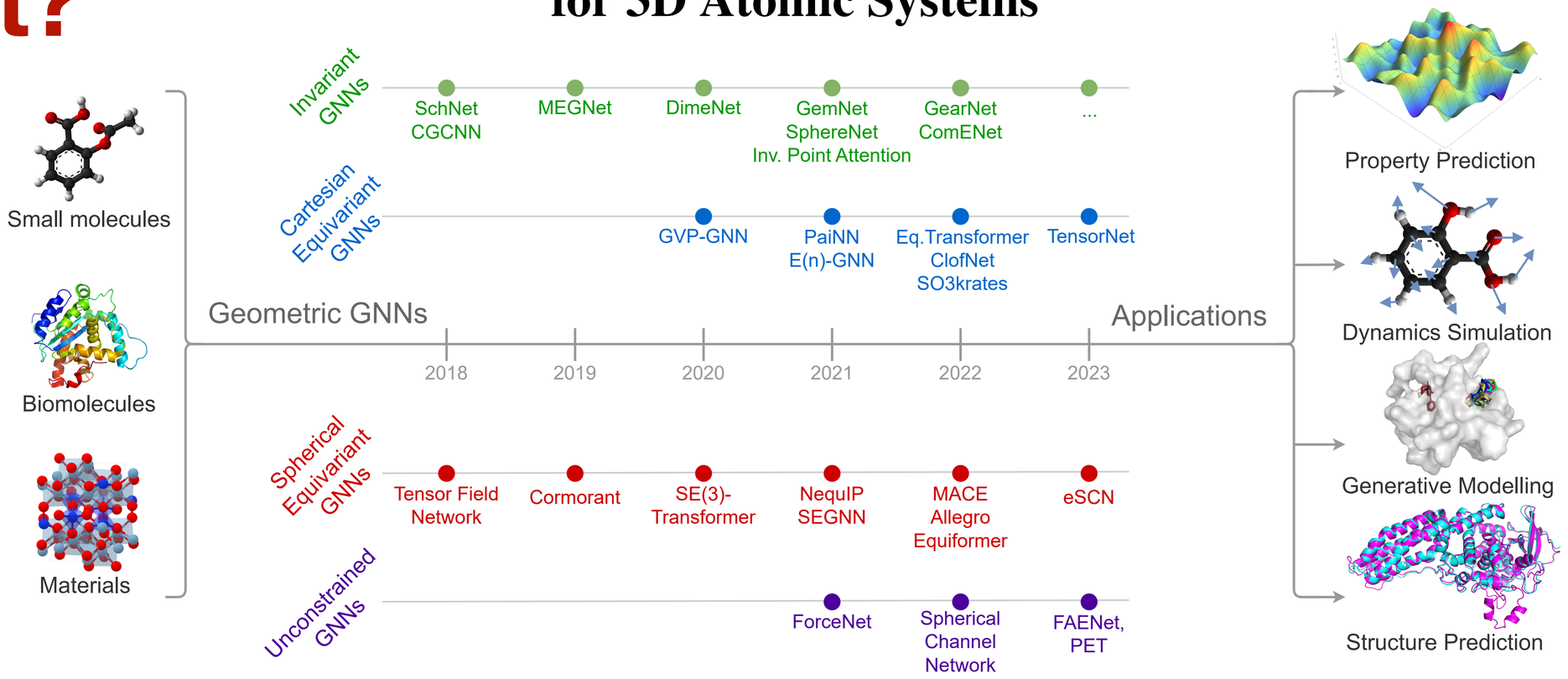

## Account for 3D symmetries



Extra

# Where to start?

## A Hitchhiker's Guide to Geometric GNNs for 3D Atomic Systems





**Alexandre Duval**[*,1,2] **Simon V. Mathis**[*,3] **Chaitanya K. Joshi**[*,3] **Victor Schmidt**[*,1,4]
**Santiago Miret**[5] **Fragkiskos D. Malliaros**[2] **Taco Cohen**[6]
**Pietro Liò**[3] **Yoshua Bengio**[1,4] **Michael Bronstein**[7]

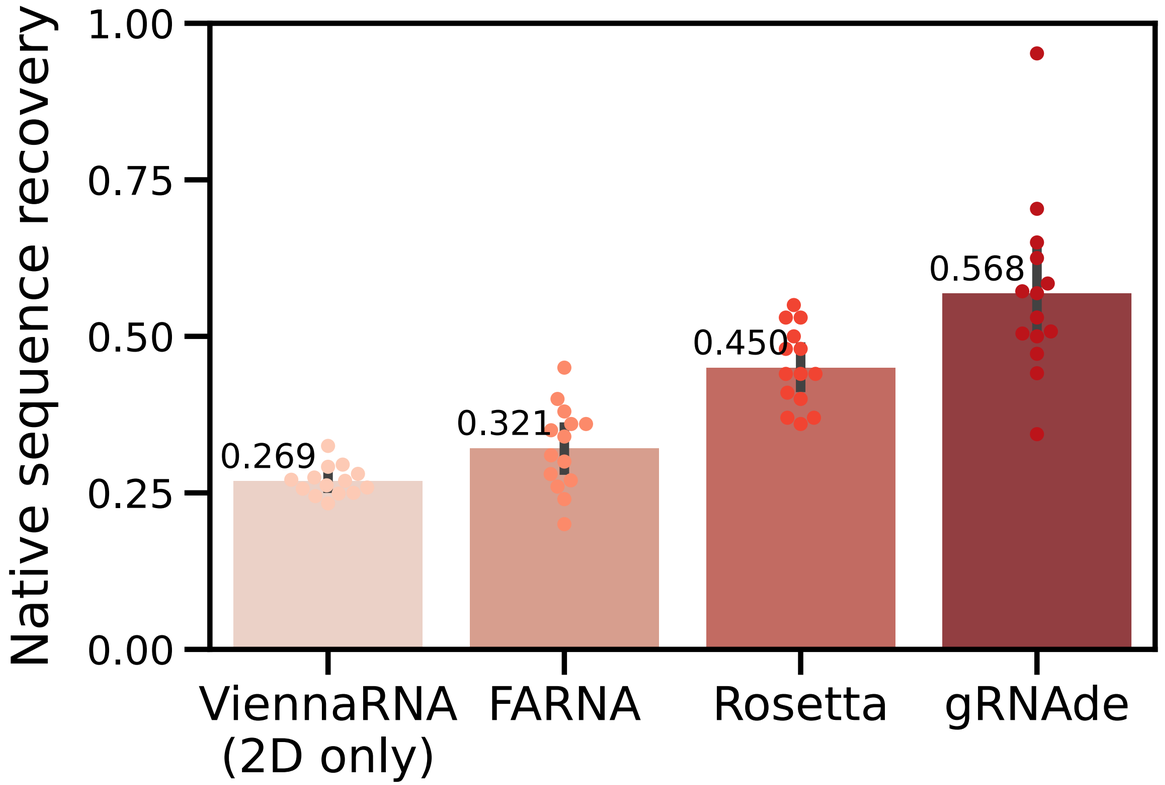
[1]Mila [2]Université Paris-Saclay [3]University of Cambridge [4]Université de Montréal
[5]Intel Labs [6]Qualcomm AI Research [7]University of Oxford
[*]Equal first authors.

# What can we do with gRNAde?

# Benchmarking single-state design

## Re-design 14 RNAs of interest from the PDB by Das et al.

### Improved sequence recovery




### Faster inference speed

- gRNAde: under 1 second for 100s of nts.
- Rosetta: order of hours…

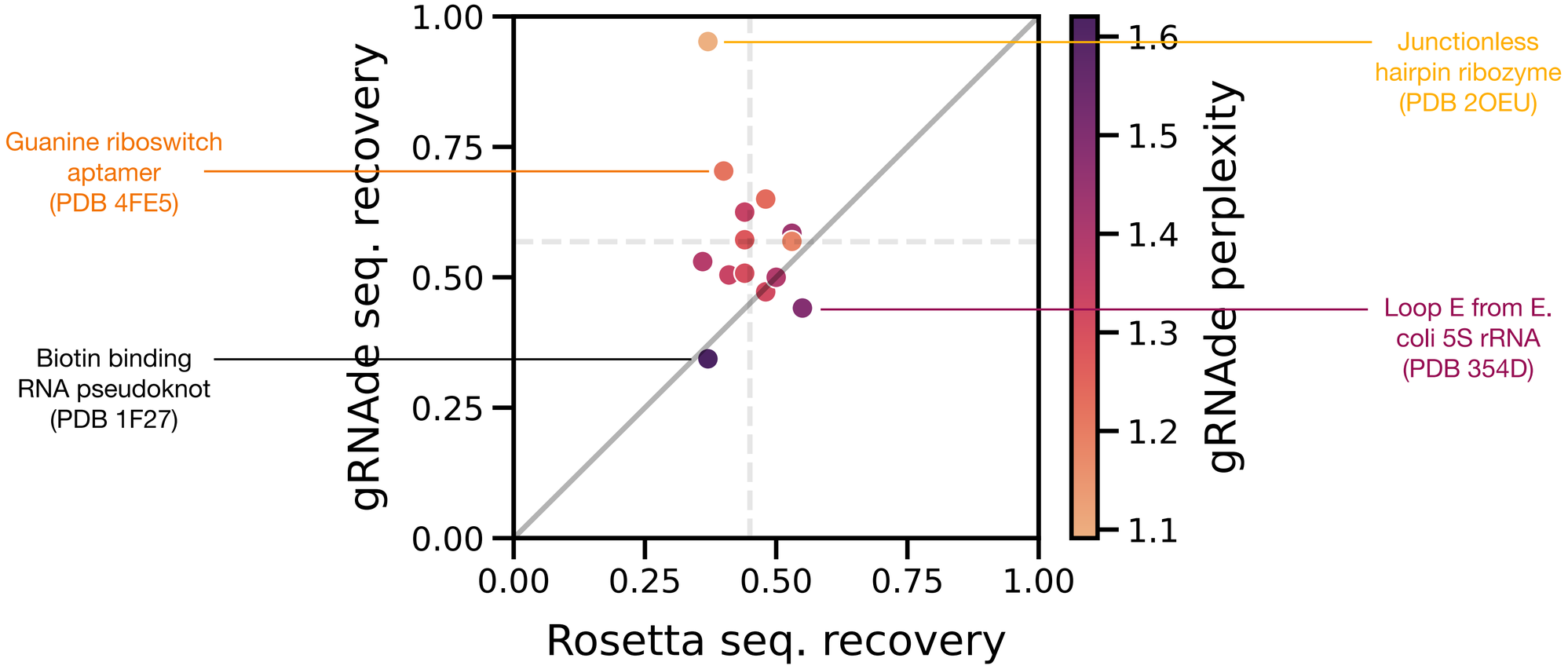
Rosetta documentation:
*“runs on RNA backbones longer than ~ten nucleotides take many minutes or hours”*

**Tried to evaluate for generalisation:**
Training data excluded all 14 RNAs and structurally identical RNAs (TM-score >0.45).

Das, Karanicolas, Baker. Atomic accuracy in predicting and designing noncanonical RNA structure. Nature Methods, 2010.

# Perplexity correlates well with recovery

## Indicator of model's confidence in its own prediction




Could perplexity be correlated with **fitness/function**, too?

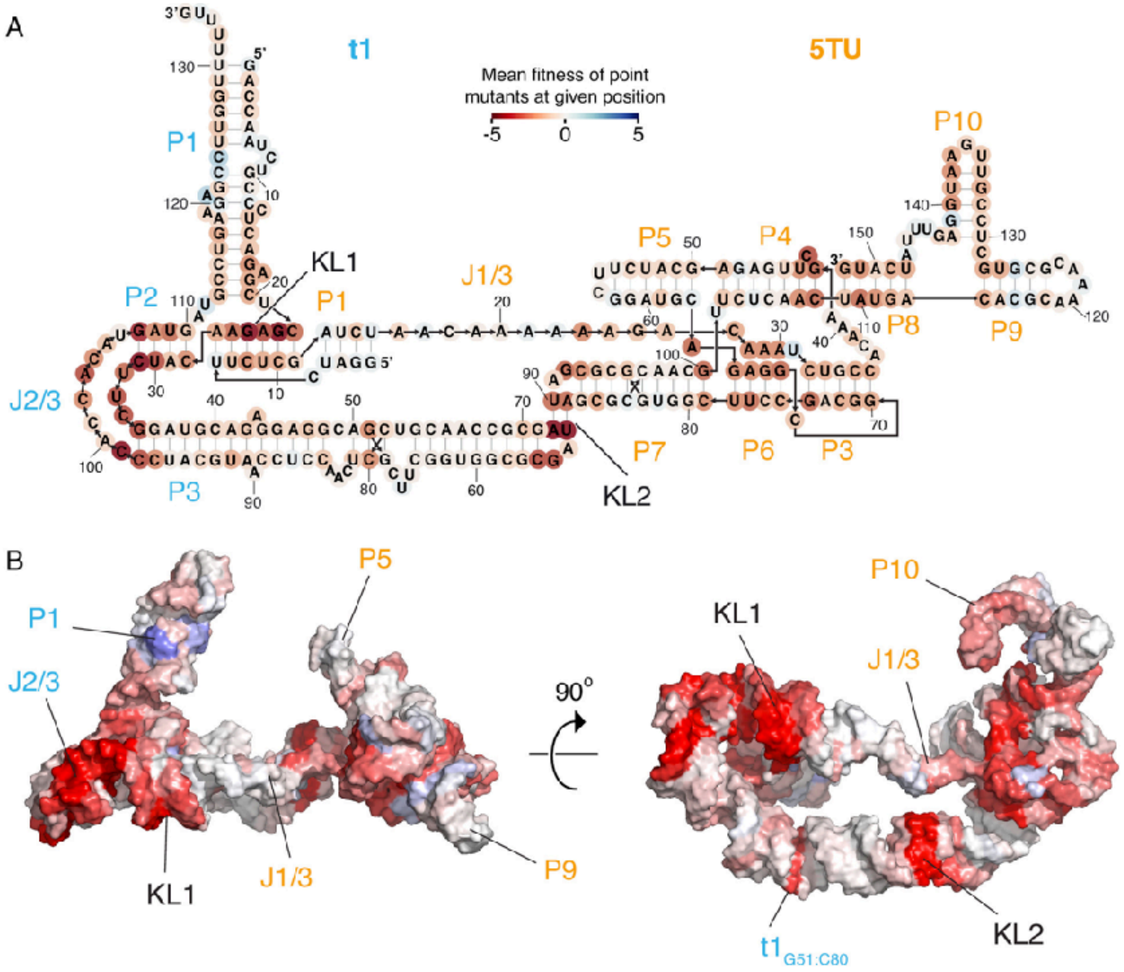
# Can gRNAde understand RNA fitness landscapes?

## A retrospective analysis on an RNA Polymerase Ribozyme (data from Phil Holliger's lab at MRC LMB)

McRae, Wan, Kristoffersen, Hansen, Gianni, Gallego, Curran, Attwater, Holliger, and Andersen.
Cryo-EM structure and functional landscape of an RNA polymerase ribozyme. PNAS, 2024.

# Structure + Functional landscape

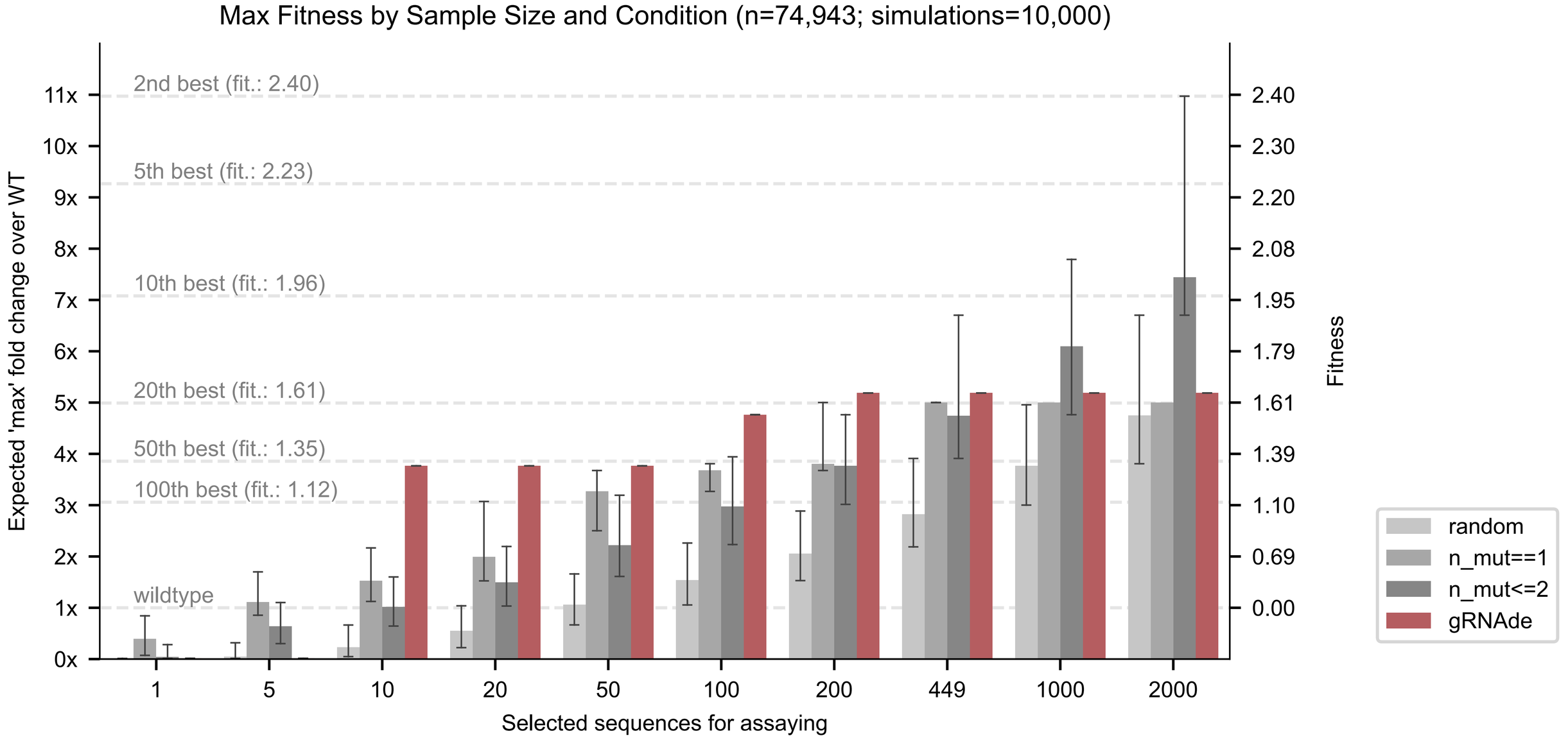
## Allows retrospectively analysis of gRNAde for RNA engineering




- **Cryo-EM structure** at 5Å resolution (not in gRNAde's training set).
- **75,000+ data points** of (mutant sequence, fitness).
- **gRNAde's perplexity**: likelihood of sequence folding into given backbone; can be used for zero-shot ranking of mutants for a given structure.
- Latent features can be used for finetuning (supervised learning), too.

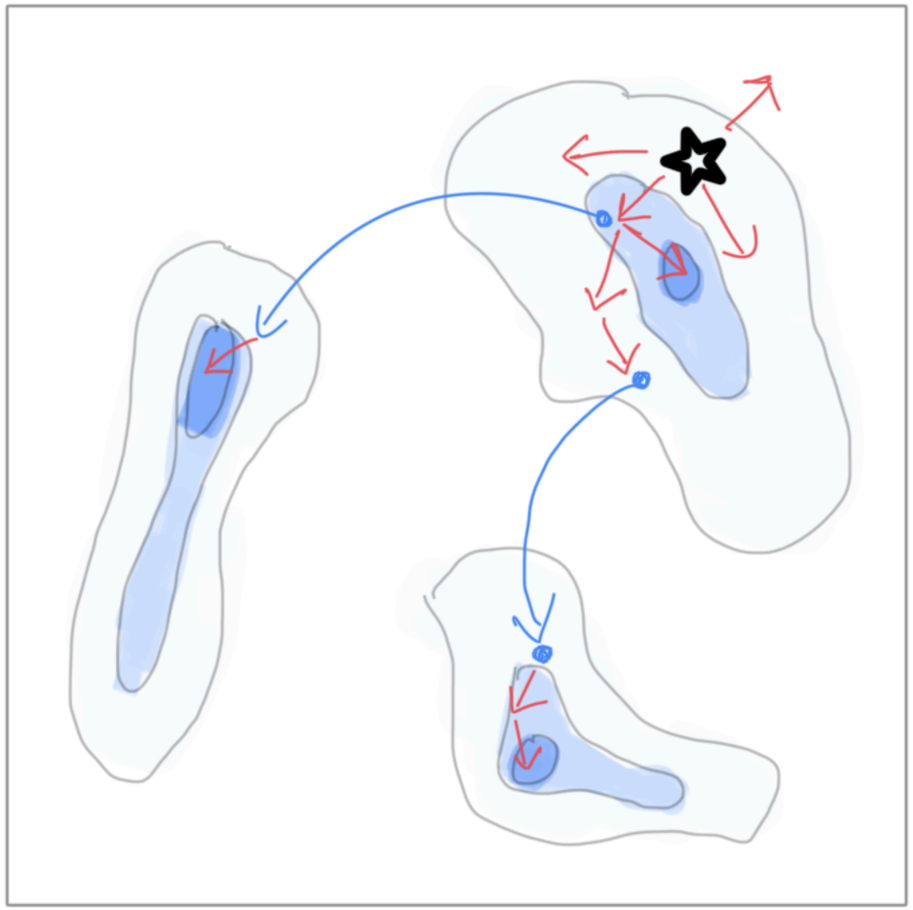
McRae et al. Cryo-EM structure and functional landscape of an RNA polymerase ribozyme. PNAS, 2024.

# Unsupervised learning of Ribozyme fitness



# A vision for AI-augmented biomolecule design

**Evolution:** local exploration, **gRNAde:** global jumps in sequence space

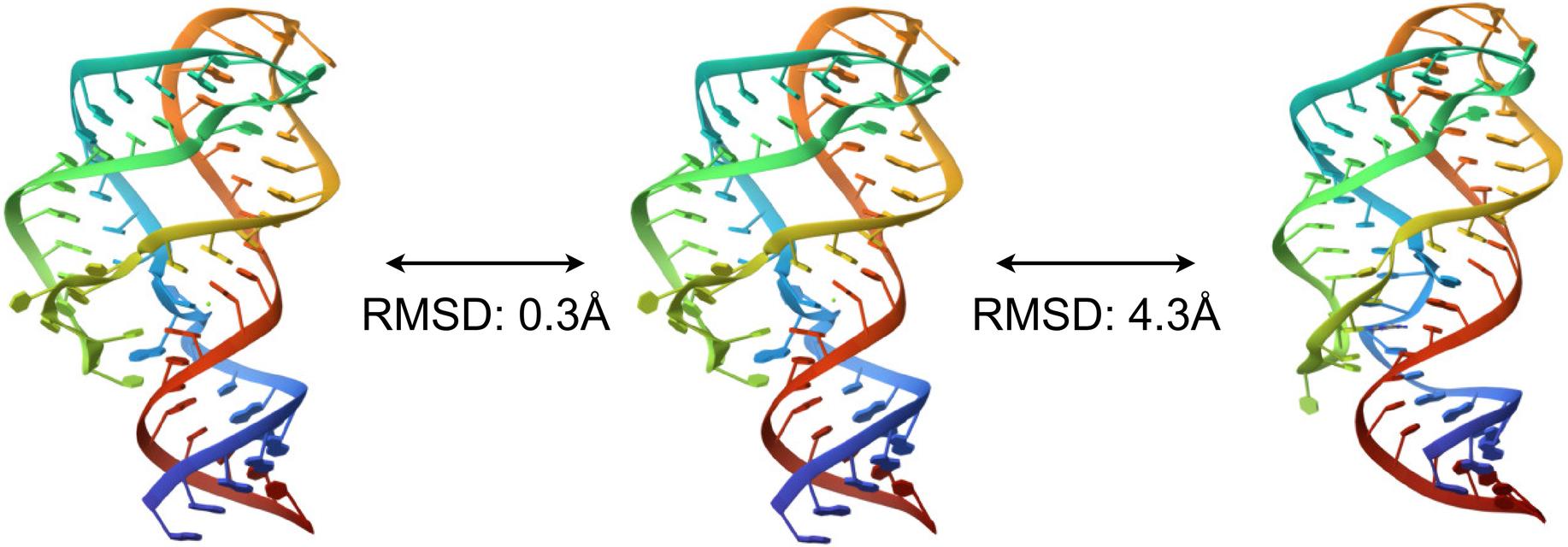



Huang, Boyken, Baker. The coming of age of de novo protein design. Nature. 2016.

# Multi-state RNA design

# Explicitly designing conformational ensembles

## Single-state design can be ambiguous




Stagno et al. Structures of riboswitch RNA reaction states by mix-and-inject XFEL serial crystallography. Nature, 2017.
Hoetzel, Suess. Structural changes in aptamers are essential for synthetic riboswitch engineering. Journal of Molecular Biology, 2022.
Ken et al. RNA conformational propensities determine cellular activity. Nature, 2023.

# Benchmarking multi-state design

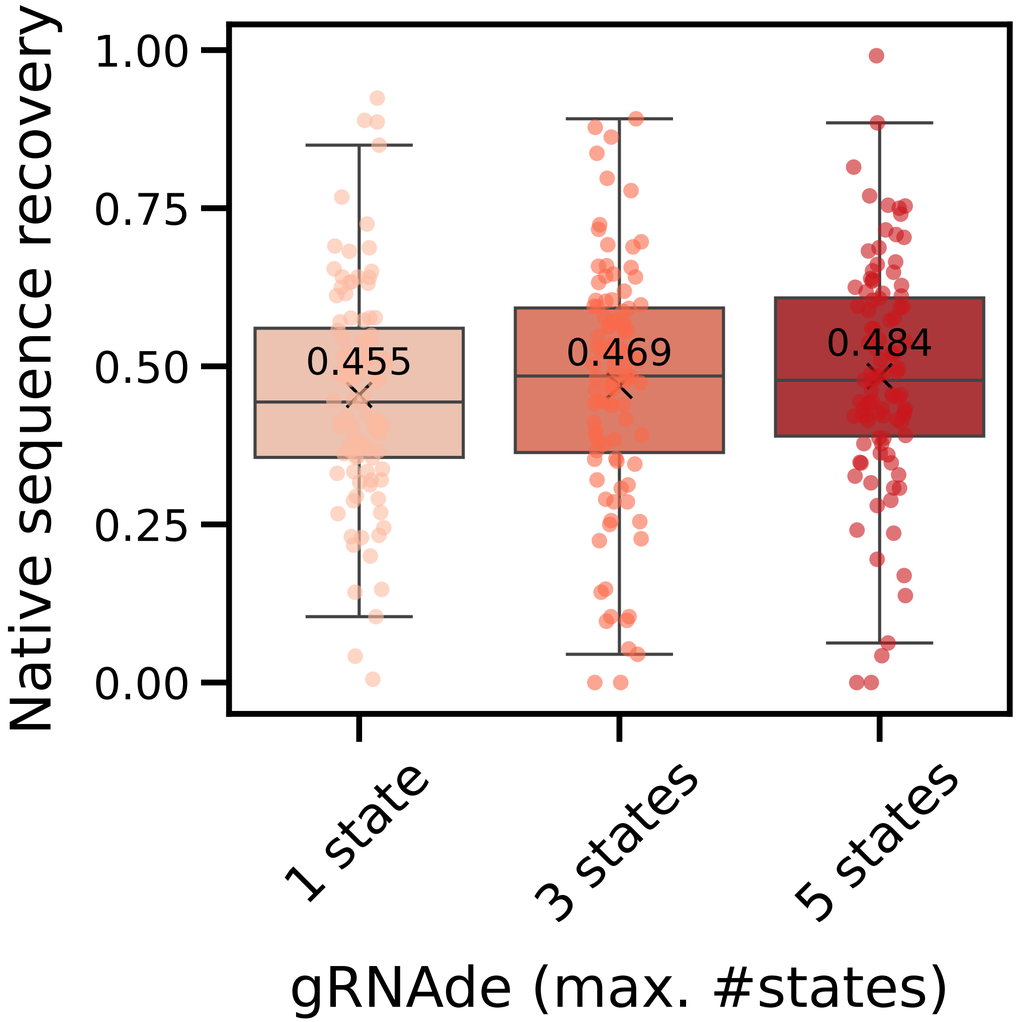
## Creating a challenging set of structurally flexible RNAs

1. **Cluster RNAsolo** based on **structural similarity** — US-align with TM-score threshold 0.45.
2. Order clusters based on **median intra-sequence RMSD** among available structures in the cluster.
3. Training, validation, and test splits become progressively more flexible.
   - **Top 100 samples** from clusters with highest intra-seq. RMSD — test set.
   - **Next 100 samples** from clusters with highest intra-seq. RMSD — validation set.
   - Very large (> 1000 nts) RNAs — training set.
4. If any samples were not assigned clusters, append them to the training set.

Test/validation set: **100 RNAs each**, training set: **~4000 RNAs**.

# Multi-state models slightly improve recovery

## Room for improvement in designing models and evaluation



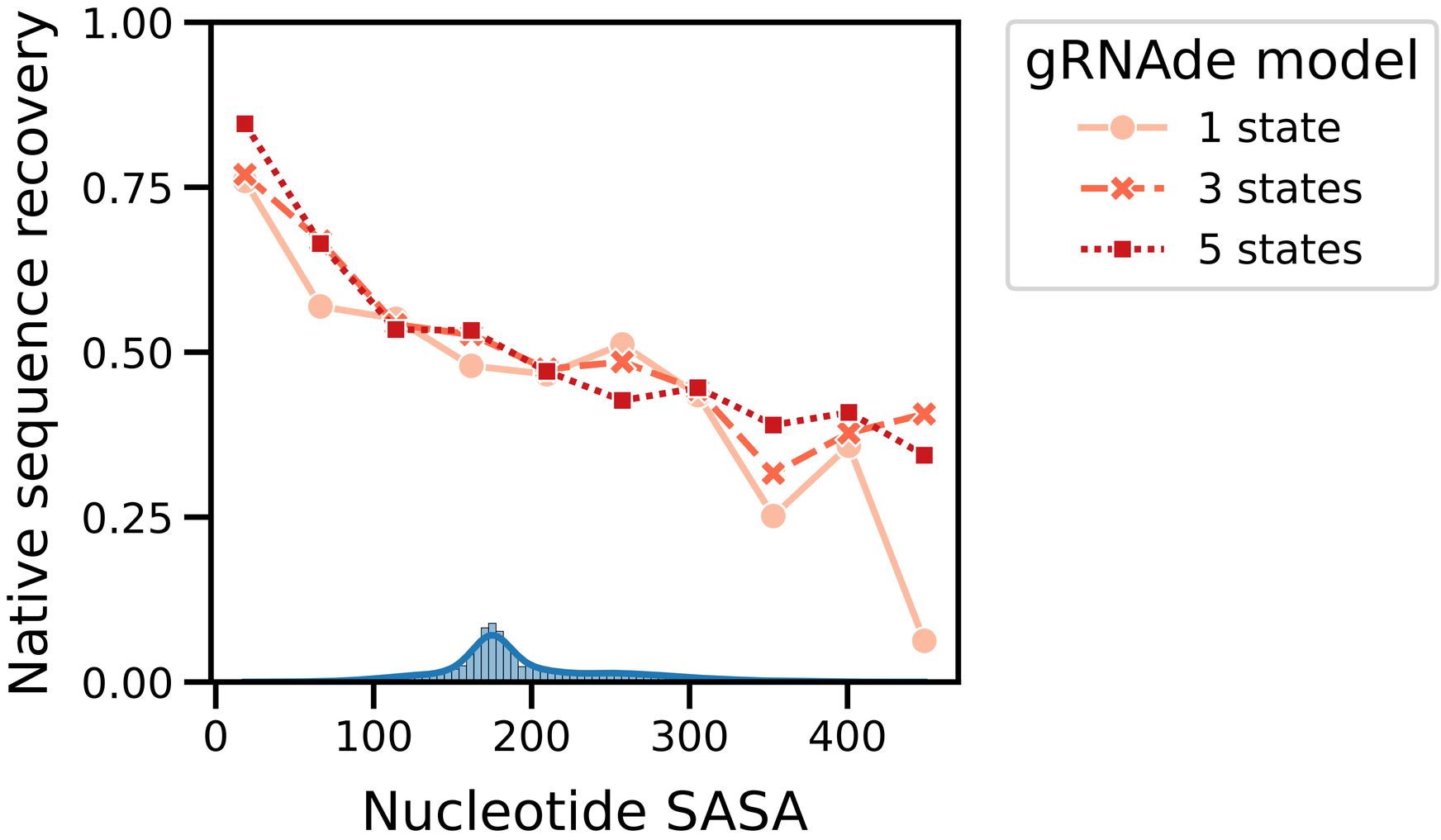

**Hypothesis:**

Multi-state gRNAde shows improved sequence recovery for structurally flexible regions of RNAs.

- Look at (local) per-nucleotide sequence recovery.

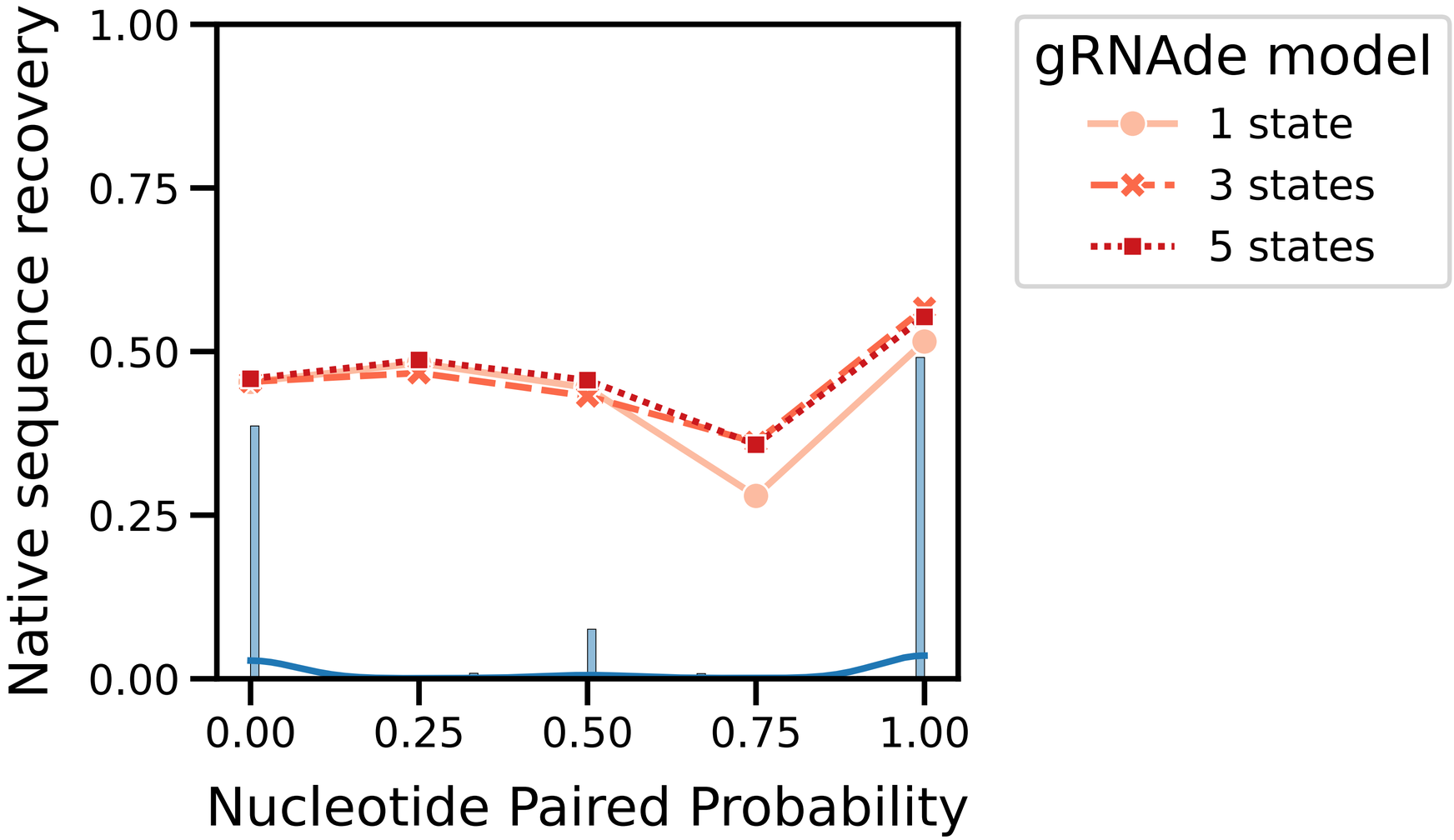
# Surface vs. core nucleotides

## Multi-state models show improved recovery on surface



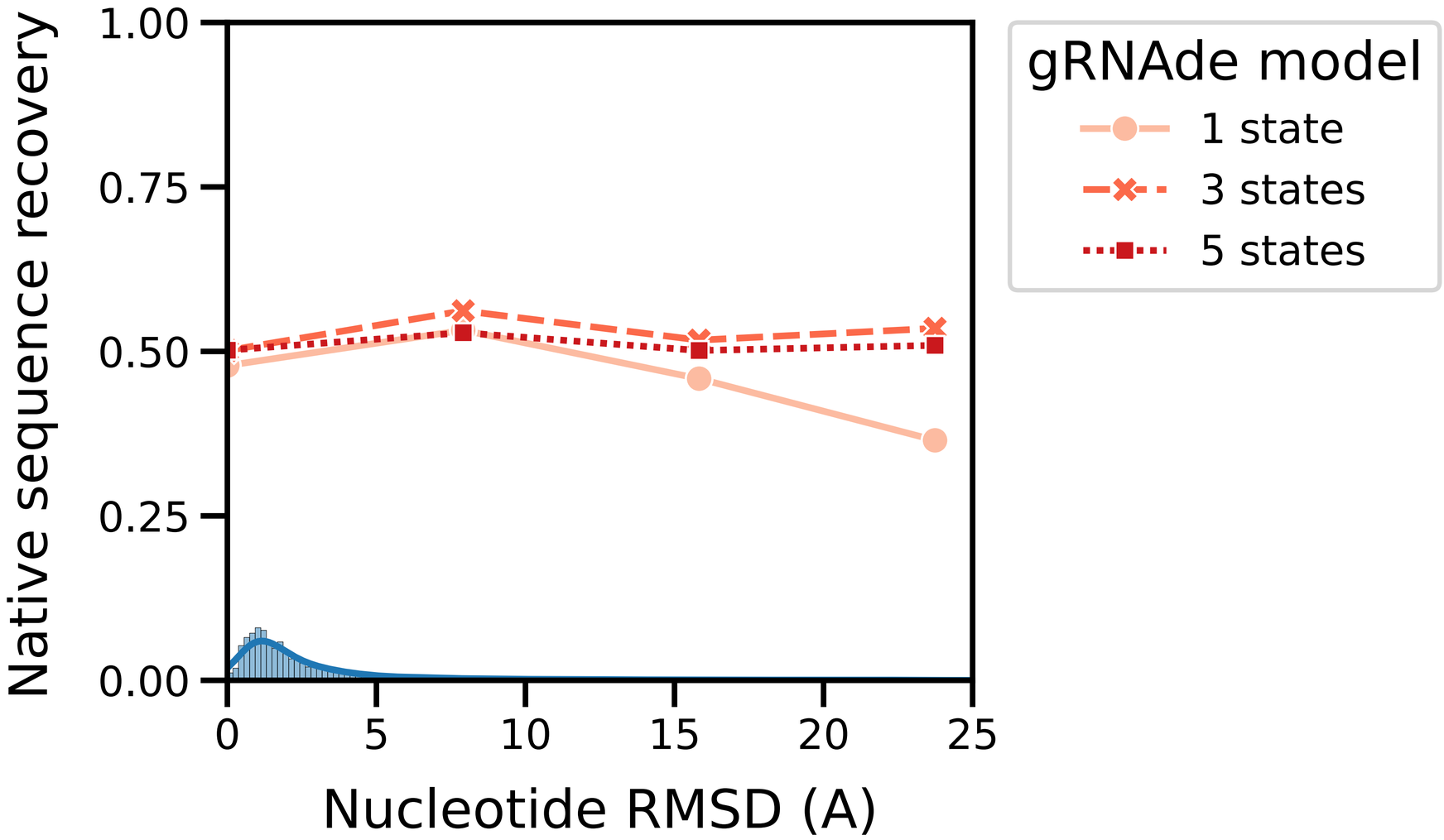
# Paired vs. unpaired nucleotides

## Multi-state models recover ambiguous positions better



# Structurally flexible nucleotides

## Multi-state models show improved recovery in variable regions



# Limitations & Future Work

# Things we are thinking about

## Applications and wet lab validation

- RNA polymerase ribozyme quasispecies.
- Riboswitches and transient gene expression.
- Want to help people actually use this — Please reach out!

## Limitations of current models

- Support for multiple chains and accounting for interactions with ligands.
- Improved architectures and benchmarking of multi-state design.

## Resources

- Open-source code and checkpoints: github.com/chaitjo/geometric-rna-design
- Tutorial available + forthcoming book chapter in *Methods in Molecular Biology*.

# Thank you for listening! Questions?

**Email:** chaitanya.joshi@cl.cam.ac.uk, **Website:** chaitjo.com

Thank you to:

Pietro Liò, Arian Jamasb, Ramon Viñas, Charles Harris, Simon Mathis, Alex Morehead, Rishabh Anand, and my labmates at Cambridge

Roger Foo (NUS, Singapore)

Phil Holliger (MRC LMB)

Alex Borodavka (Cambridge Biochem.)

Janusz Bujnicki (IIMCB, Warsaw)

Mihir Metkar (Moderna)

Rhiju Das (Stanford)

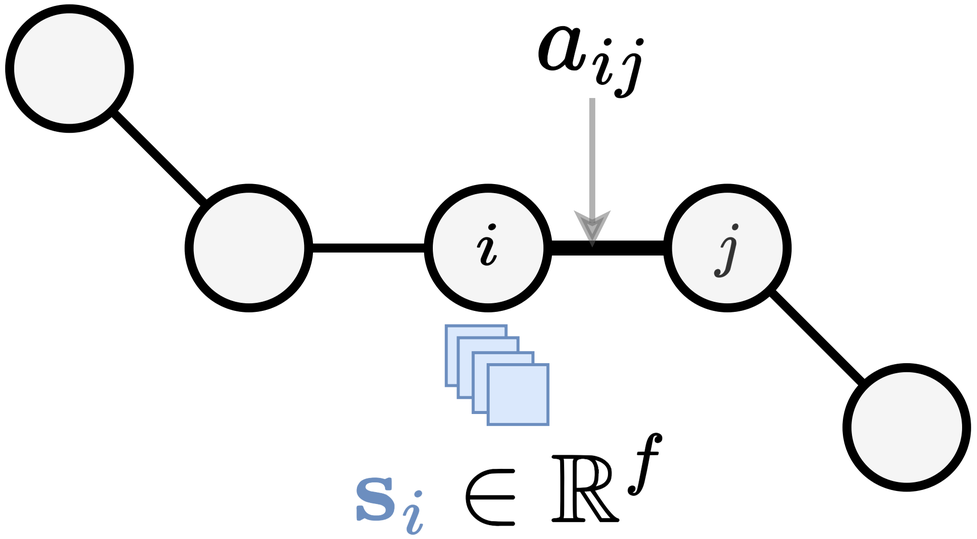
# Primer on Geometric Graph Neural Networks

**A Hitchhiker's Guide to Geometric GNNs for 3D Atomic Systems.** Alexandre Duval*, Simon V. Mathis*, Chaitanya K. Joshi*, Victor Schmidt*, Santiago Miret, Fragkiskos D. Malliaros, Taco Cohen, Pietro Liò, Yoshua Bengio, Michael Bronstein.

**On the Expressive Power of Geometric Graph Neural Networks.** Chaitanya K. Joshi*, Cristian Bodnar*, Simon V. Mathis, Taco Cohen, and Pietro Liò. ICML 2023.

# Normal graphs

## A graph is a set of nodes connected by edges




E.g. atom type

$$\mathcal{G} = (\boldsymbol{A}, \boldsymbol{S})$$

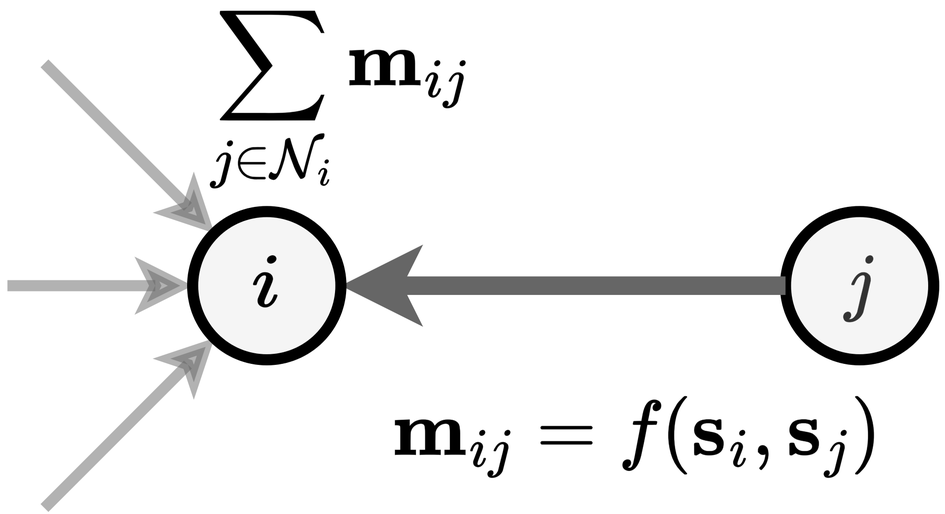
Scalar features $\in \mathbb{R}^{n \times f}$

$n \times n$ adjacency matrix

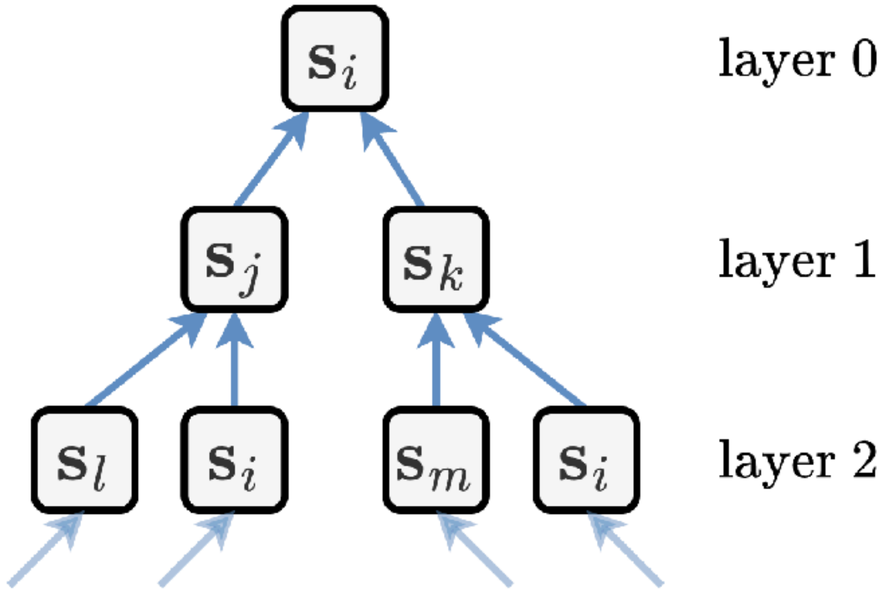
**Note:** $f$ is the dimension or number of scalar feature channels.

# Normal Graph Neural Networks

## Message passing updates node features using local aggregation



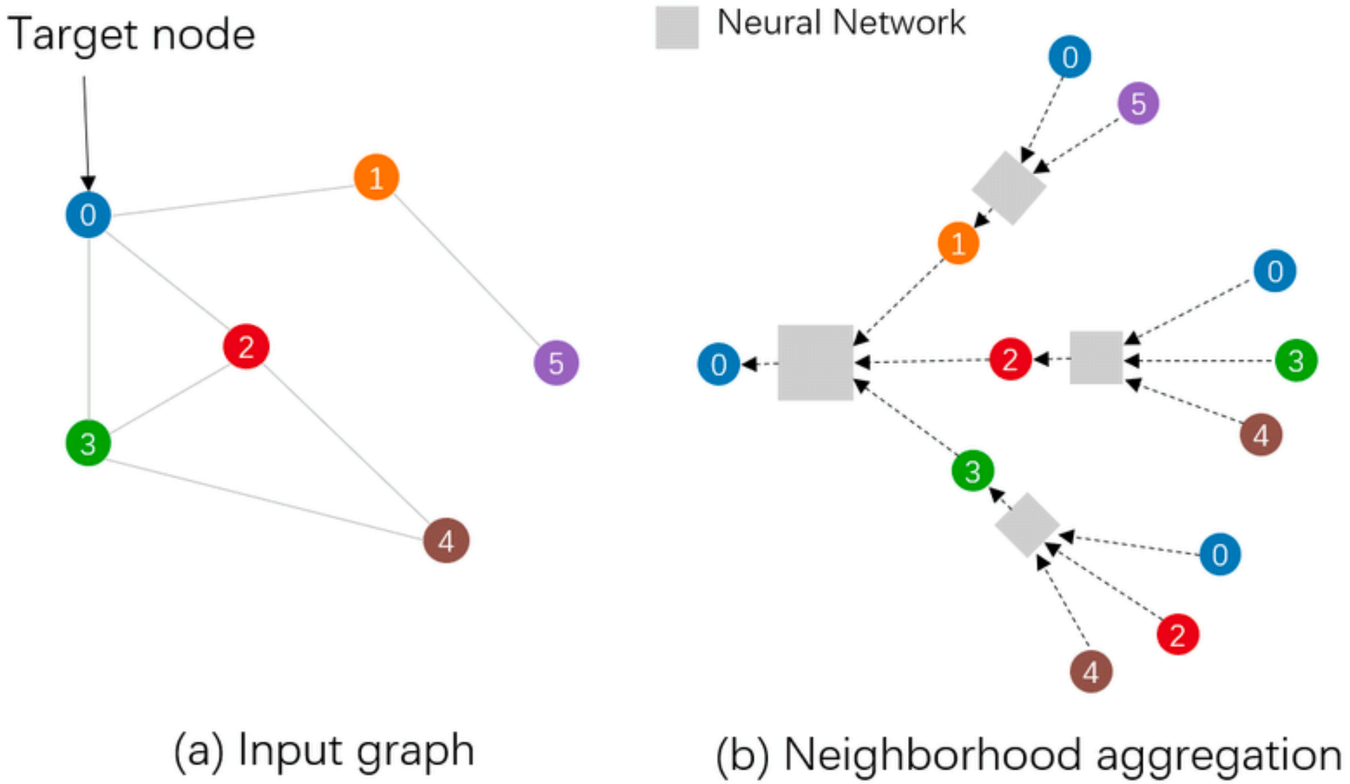

$$m_i^{(t)} := \text{AGG}\left(\{\!\{(s_i^{(t)}, s_j^{(t)}) \mid j \in \mathcal{N}_i\}\!\}\right),$$
$$s_i^{(t+1)} := \text{UPD}\left(s_i^{(t)}, m_i^{(t)}\right),$$




**Computation tree:**
Message passing gathers & propagates features beyond local neighbourhoods.

# Normal Graph Neural Networks

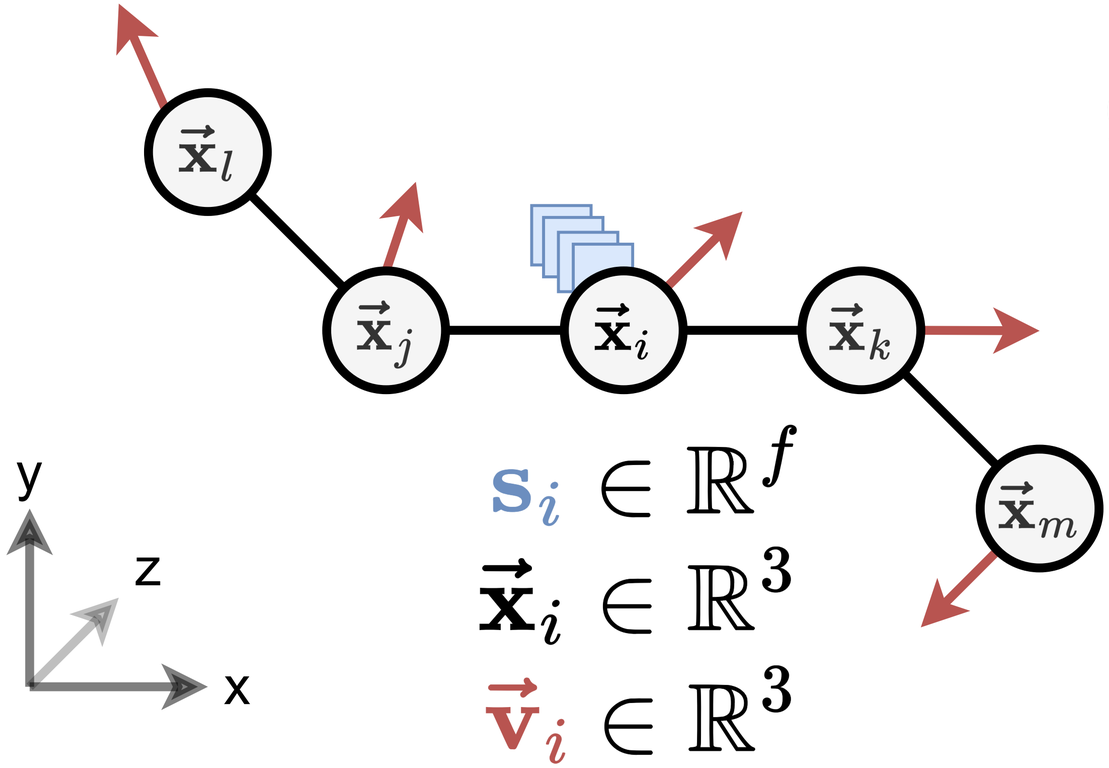
## Learn how to propagate information along the graph



# Geometric graphs

Each node is:

- **embedded in Euclidean space** *e.g.* atoms in 3D
- **decorated with geometric attributes** *s.a.* velocity




$$\mathcal{G} = (\boldsymbol{A}, \boldsymbol{S}, \vec{\boldsymbol{X}}, \vec{\boldsymbol{V}})$$

Geometric features*
$\in \mathbb{R}^{n \times 3}$

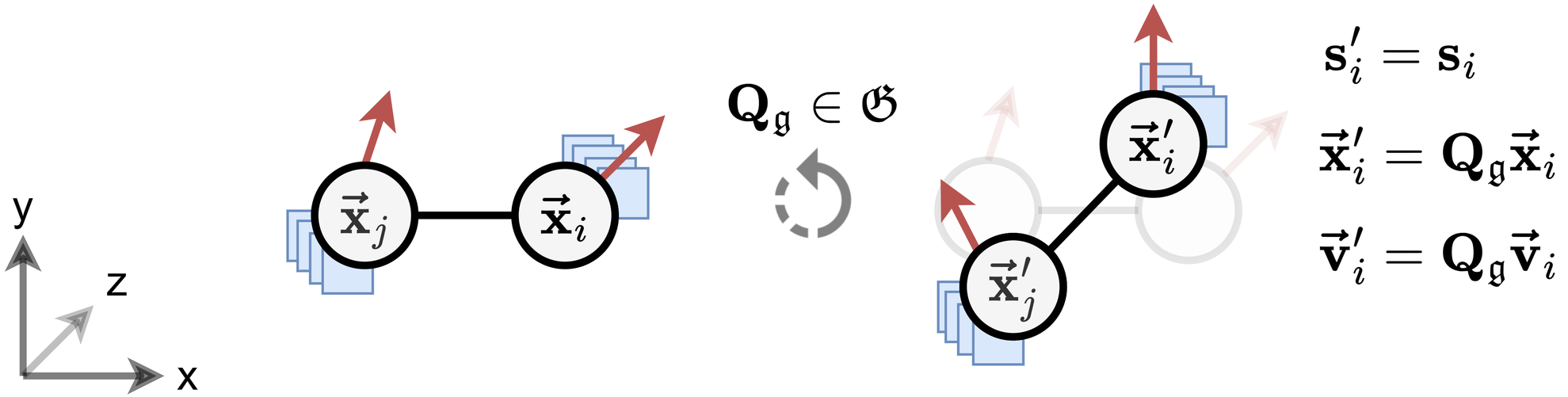
Coordinates
$\in \mathbb{R}^{n \times 3}$

* We work with a single vector feature per node, but our setup generalises to multiple vector features and higher-order tensors.

# Physical symmetries

**Geometric attributes transform with Euclidean transformations of the system**

**Rotations** & **Reflections** $Q_{\mathfrak{g}} \in \mathfrak{G}$ act on only vectors $\vec{V}$ and coordinates $\vec{X}$:



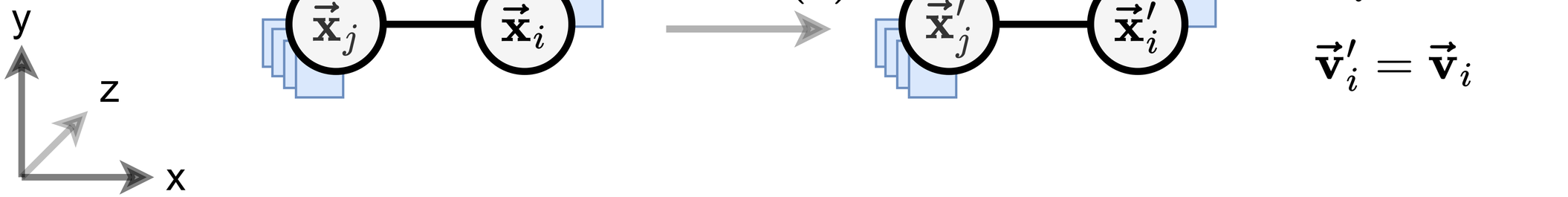

Scalar features remain unchanged $\rightarrow$ **invariant**.

* We use $\mathfrak{G}$ to denote rotations SO(d) or rotations and reflections O(d)

# Physical symmetries

**Geometric attributes transform with Euclidean transformations of the system**

**Translations** $\vec{t} \in T(d)$ act on only the coordinates $\vec{X}$:




$$\mathbf{s}'_i = \mathbf{s}_i$$

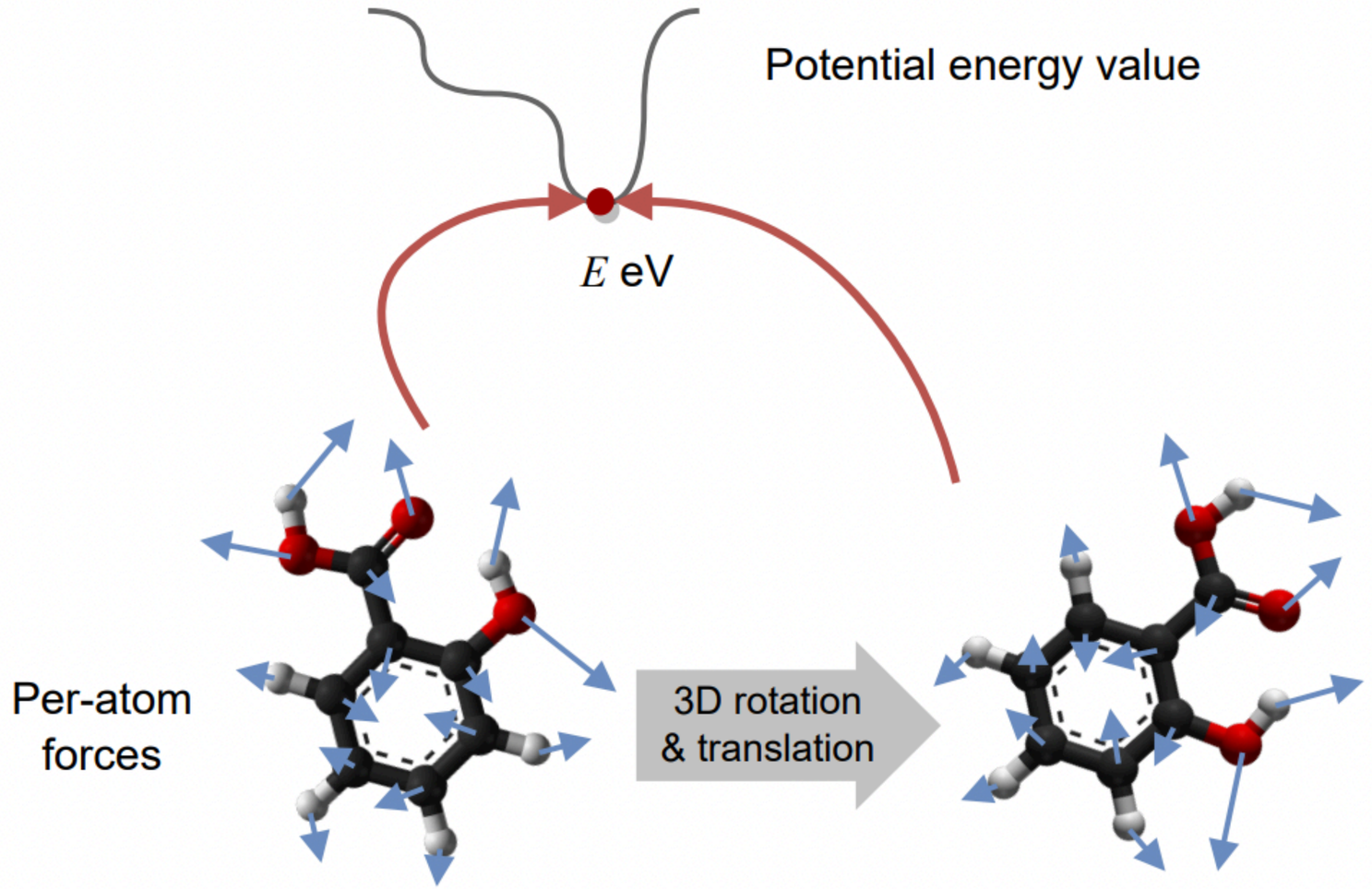
$$\vec{\mathbf{x}}'_i = \vec{\mathbf{x}}_i + \vec{\mathbf{t}}$$

$$\vec{\mathbf{v}}'_i = \vec{\mathbf{v}}_i$$

Scalar and vector features remain unchanged $\rightarrow$ **invariant**.

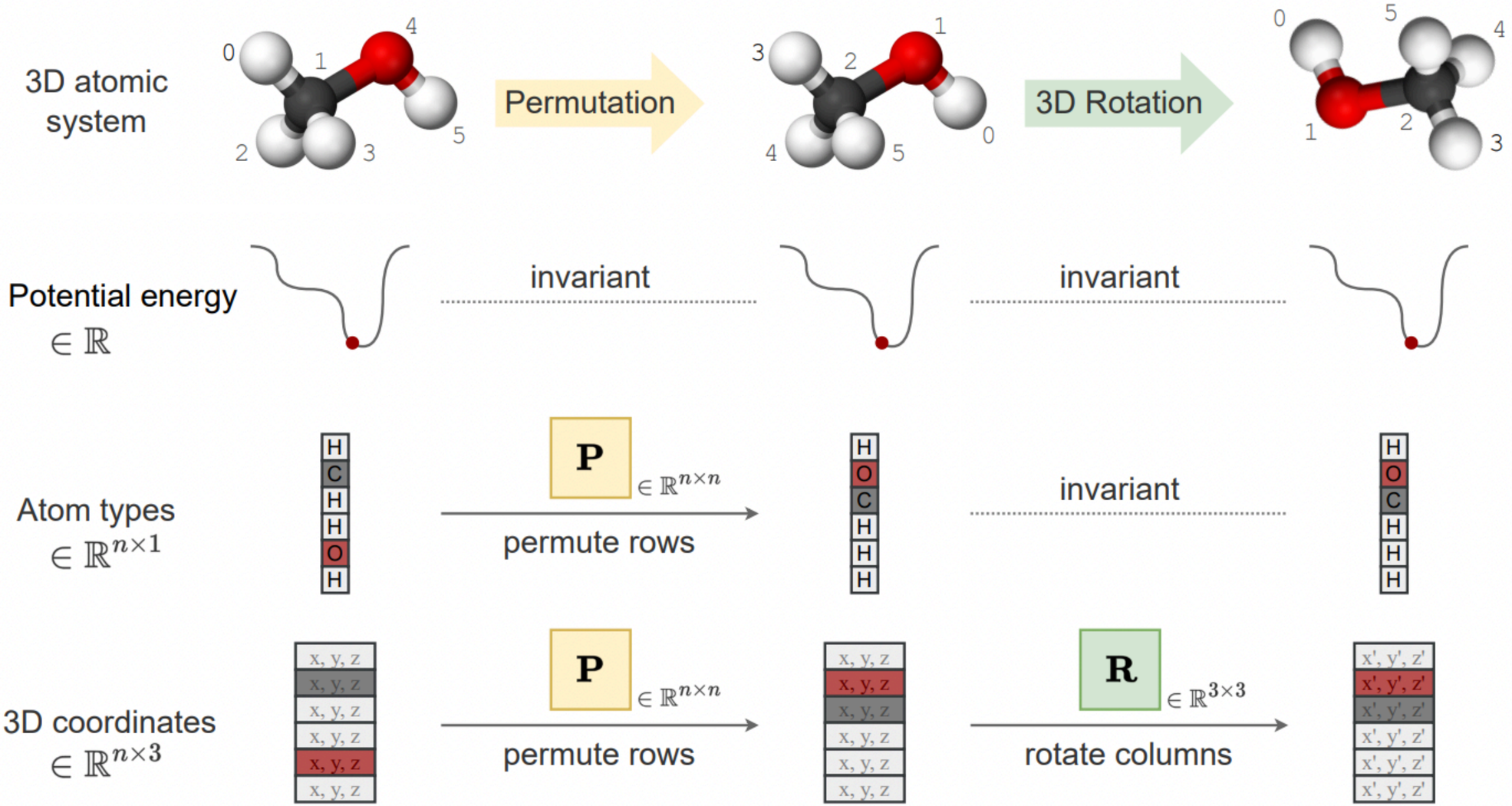
# How to build physics into GNNs?

## Geometric GNNs should account for physical symmetries



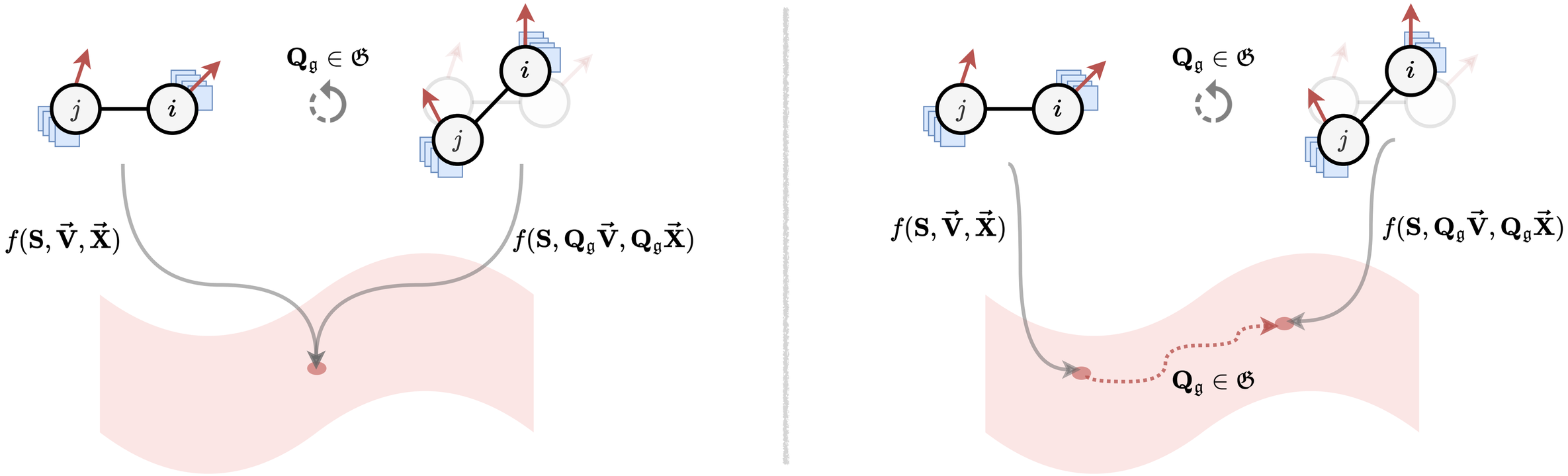
# Why build physics into GNNs?

## Geometric GNNs should account for physical symmetries



# Building blocks of Geometric GNNs

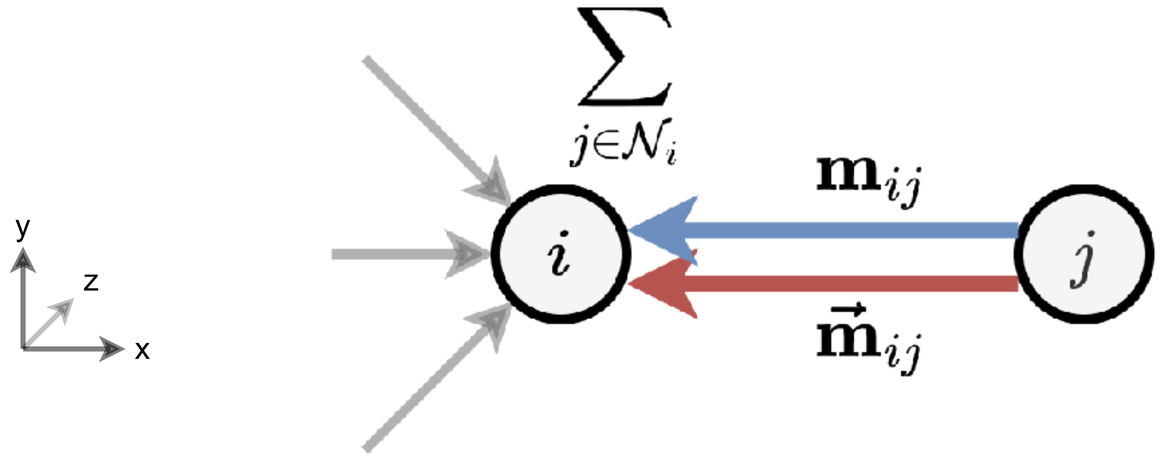
- **Scalar features** must be updated in an invariant manner.
- **Vector features** must be updated in an equivariant manner.




Invariant functions vs. Equivariant functions

# Geometric message passing

- **update scalar and** (optionally) **vector features**
- **aggregate and update functions which retain transformation semantics**




$$\boldsymbol{m}_i^{(t)}, \vec{\boldsymbol{m}}_i^{(t)} := \textsc{Agg}\left(\{\!\{(\boldsymbol{s}_i^{(t)}, \boldsymbol{s}_j^{(t)}, \vec{\boldsymbol{v}}_i^{(t)}, \vec{\boldsymbol{v}}_j^{(t)}, \vec{\boldsymbol{x}}_{ij}) \mid j \in \mathcal{N}_i\}\!\}\right) \quad \text{(Aggregate)}$$

$$\boldsymbol{s}_i^{(t+1)}, \vec{\boldsymbol{v}}_i^{(t+1)} := \textsc{Upd}\left((\boldsymbol{s}_i^{(t)}, \vec{\boldsymbol{v}}_i^{(t)}),\ (\boldsymbol{m}_i^{(t)}, \vec{\boldsymbol{m}}_i^{(t)})\right) \quad \text{(Update)}$$