COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

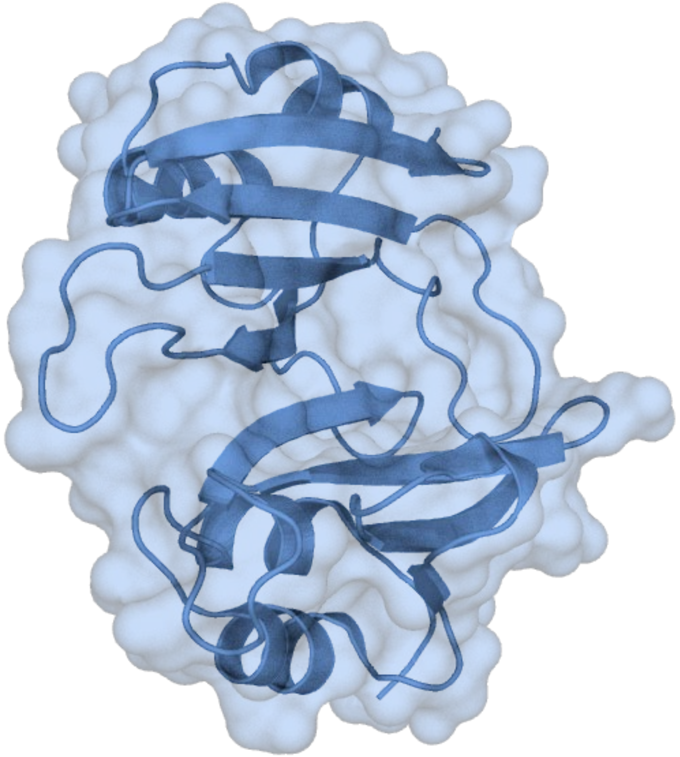
OpenFold

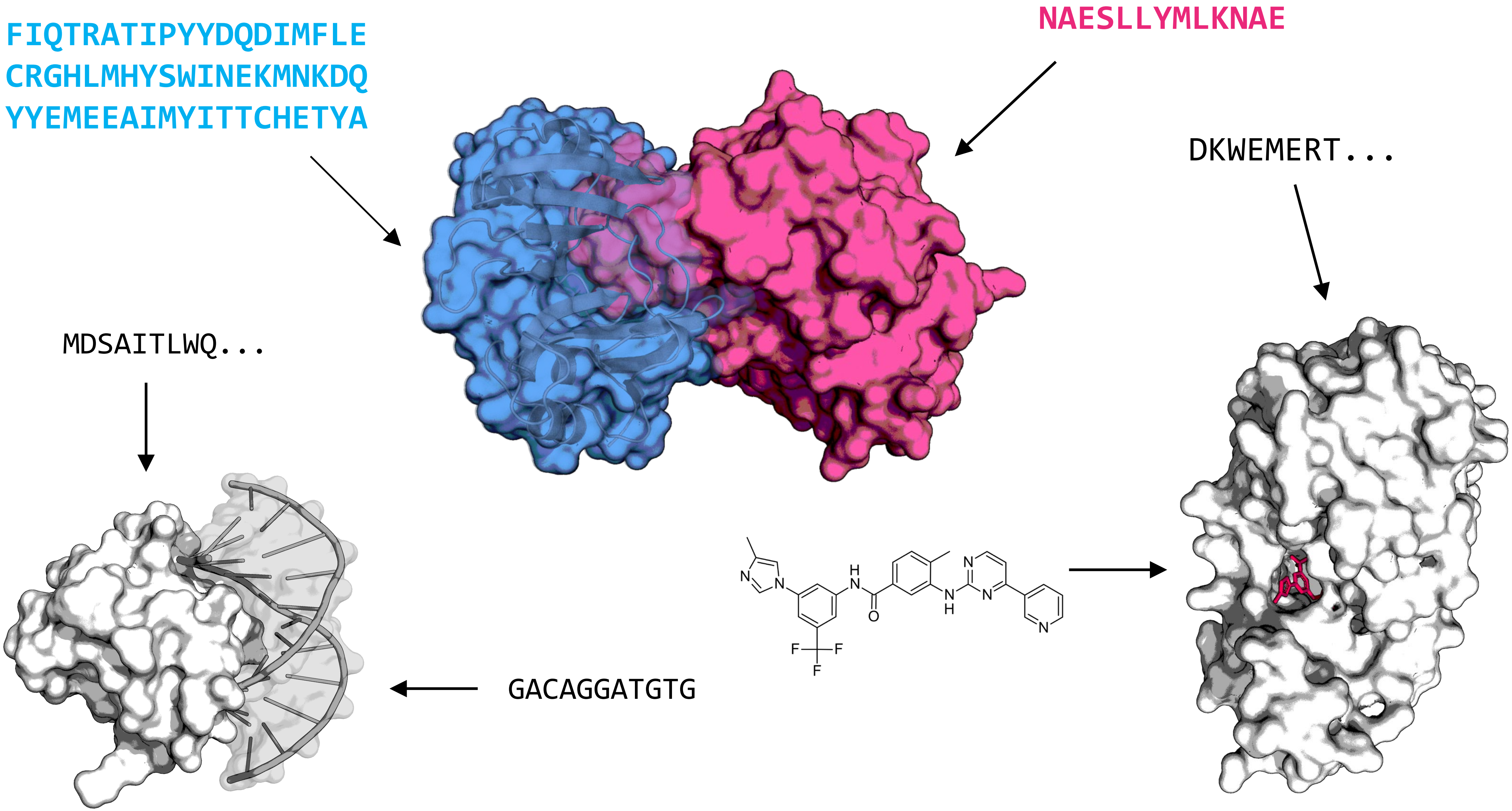
Lessons learned and insights gained from rebuilding and retraining AlphaFold2

Mohammed AlQuraishi

IPAM Learning and Emergence in Molecular Systems Workshop, Jan 23rd, 2023

FIQTRATIPYYDQDIMFLE
CRGHLMHYSWINEKMNKDQ
YYEMEEAIMYITTCHETYA





FIQTRATIPYYDQDIMFLE
CRGHLMHYSWINEKMNKDQ
YYEMEEAIMYITTCHETYA

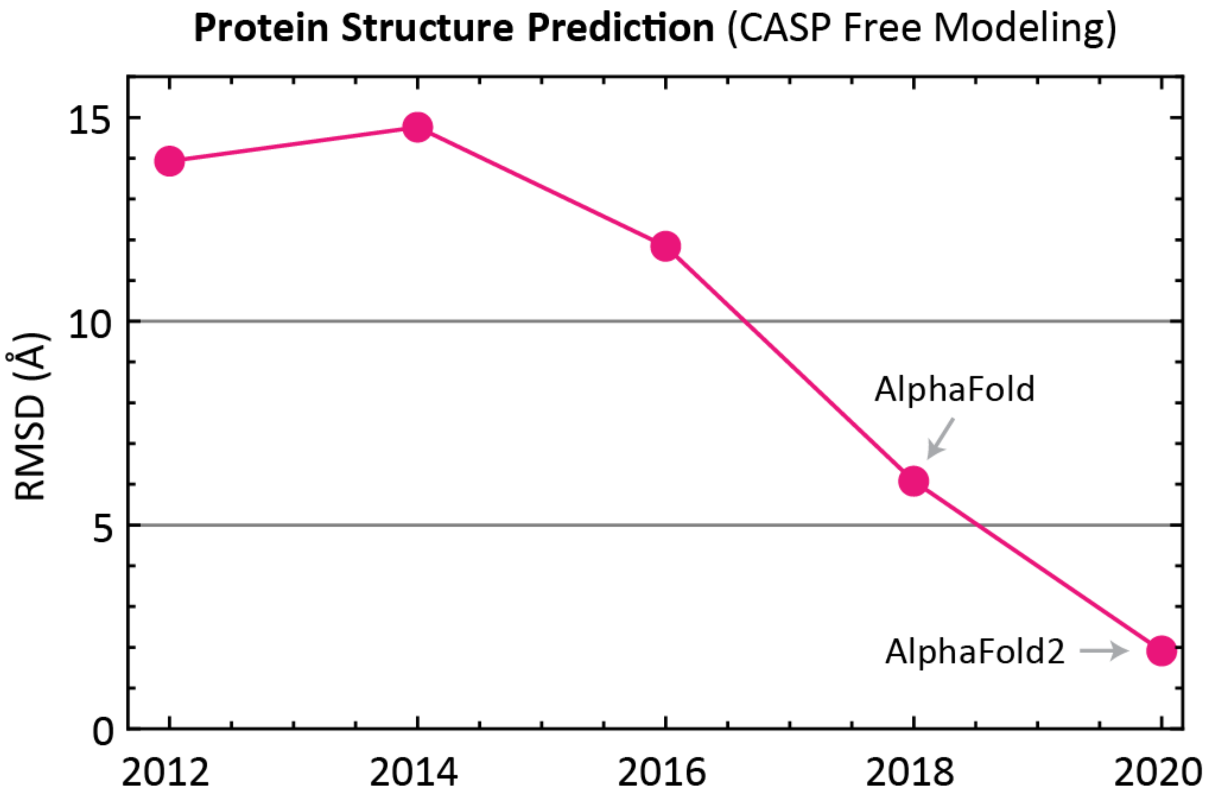
NAESLLYMLKNAE

DKWEMERT...

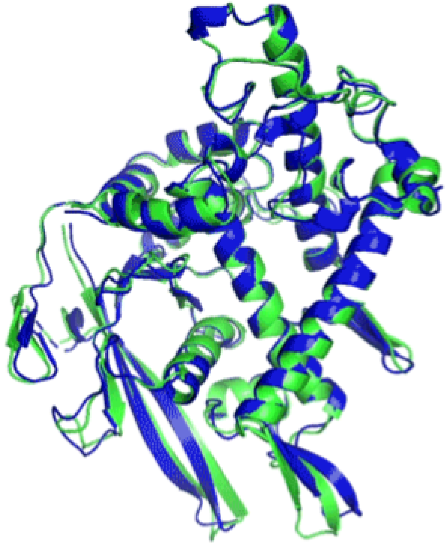
MDSAITLWQ...

GACAGGATGTG

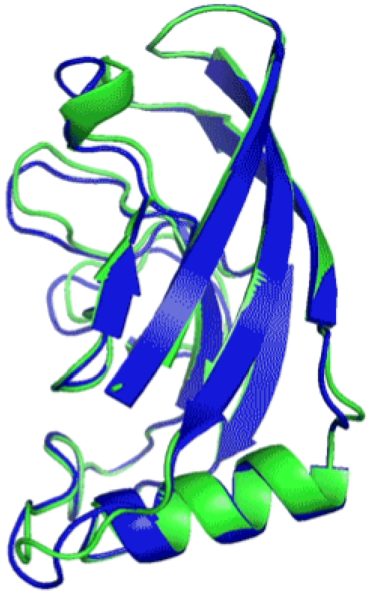
AlphaFold2 Revolution



Protein Structure Prediction (CASP Free Modeling)



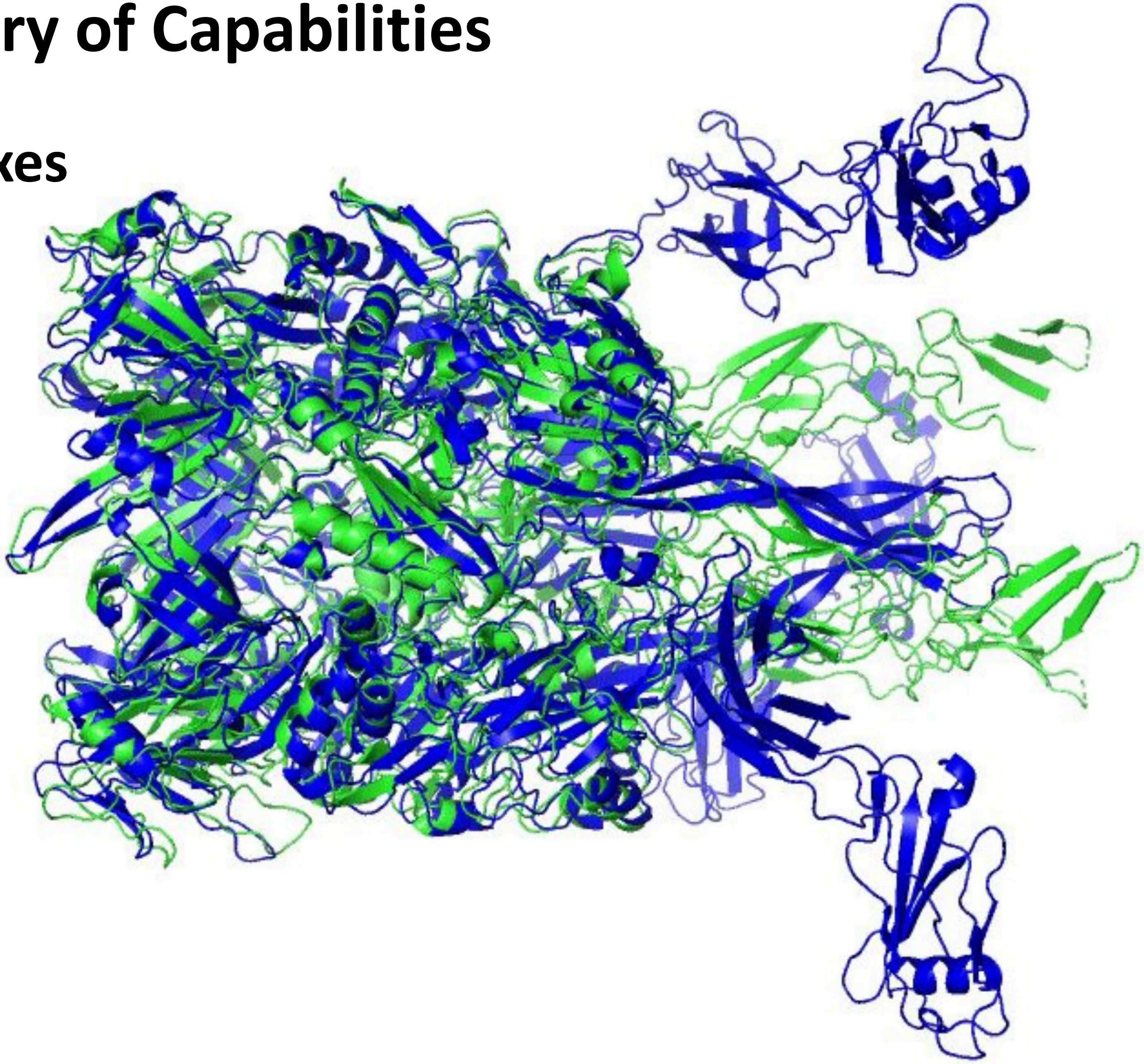
T1037 / 6vr4
90.7 GDT
(RNA polymerase domain)



T1049 / 6y4f
93.3 GDT
(adhesin tip)

Summary of Capabilities

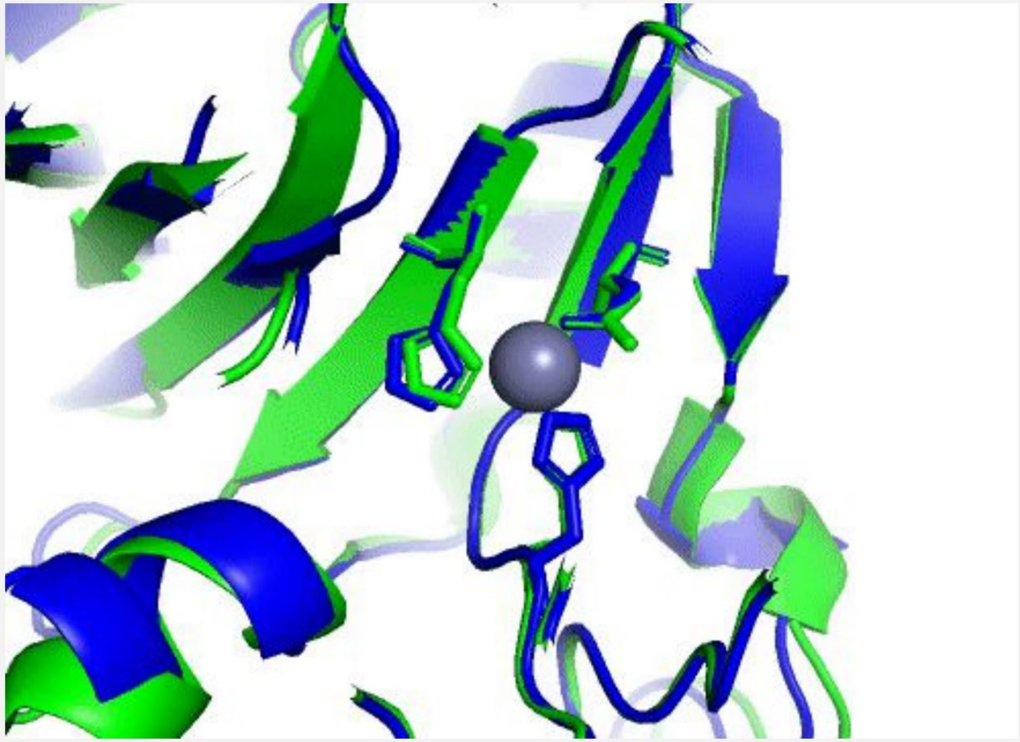
- **Single proteins and complexes**
  - As long as can fit into GPUs
  - Median accuracy ~2Å
- **Struggles a bit with**
  - Multi-domain proteins
- **Struggles a lot with**
  - Mutations
  - Single sequences
- **Can't handle**
  - Ligands, co-factors, et cetera.
  - Modified amino acids
  - Environmental conditions



Summary of Capabilities

- **Single proteins and complexes**
  - As long as can fit into GPUs
  - Median accuracy ~2Å
- **Struggles a bit with**
  - Multi-domain proteins
- **Struggles a lot with**
  - Mutations
  - Single sequences
- **Can’t handle**
  - Ligands, co-factors, et cetera.
  - Modified amino acids
  - Environmental conditions

T1056 (zinc binding)



AlphaFold / Experiment

CASP15

#1 PEZYFoldings AF2-based. Diverse MSAs. Custom, fine-tuned AF2 refinement

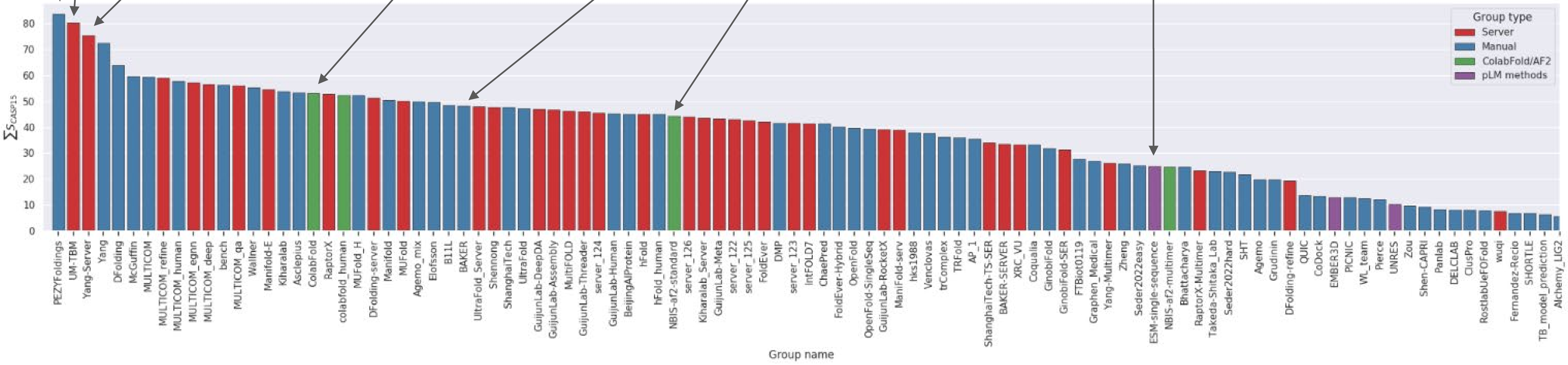
#2 UM-TBM Diverse MSAs. Threading then AF2 predictions guide I-TASSER REMC

#3 Yang-Server Diverse MSAs. AF2 predictions fed to trRosettaX2

ColabFold and NBIS-af2-standard

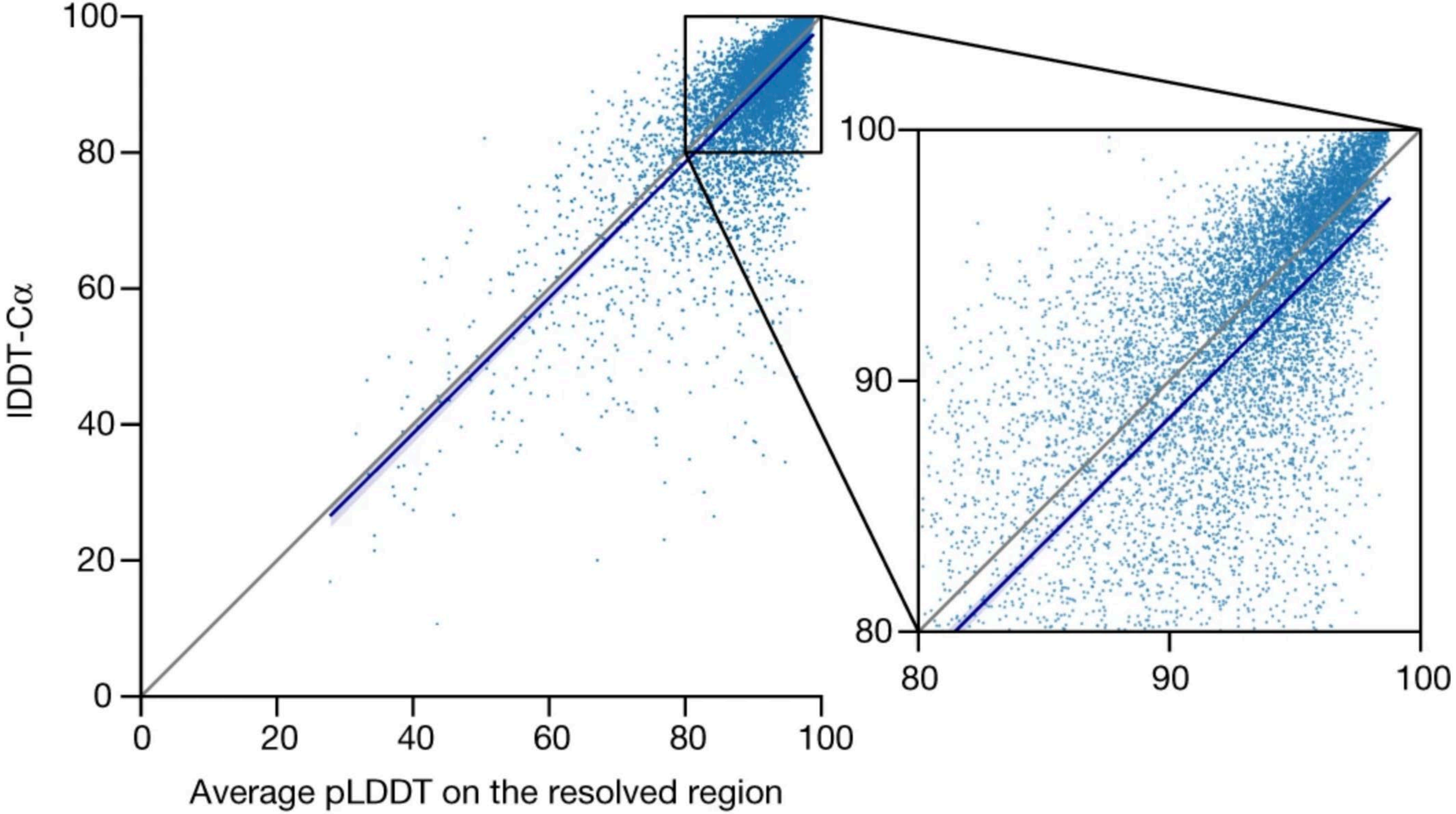
BAKER top non-AF2 method

ESM-singlesequence is the top pure pLM method (Built using OpenFold)



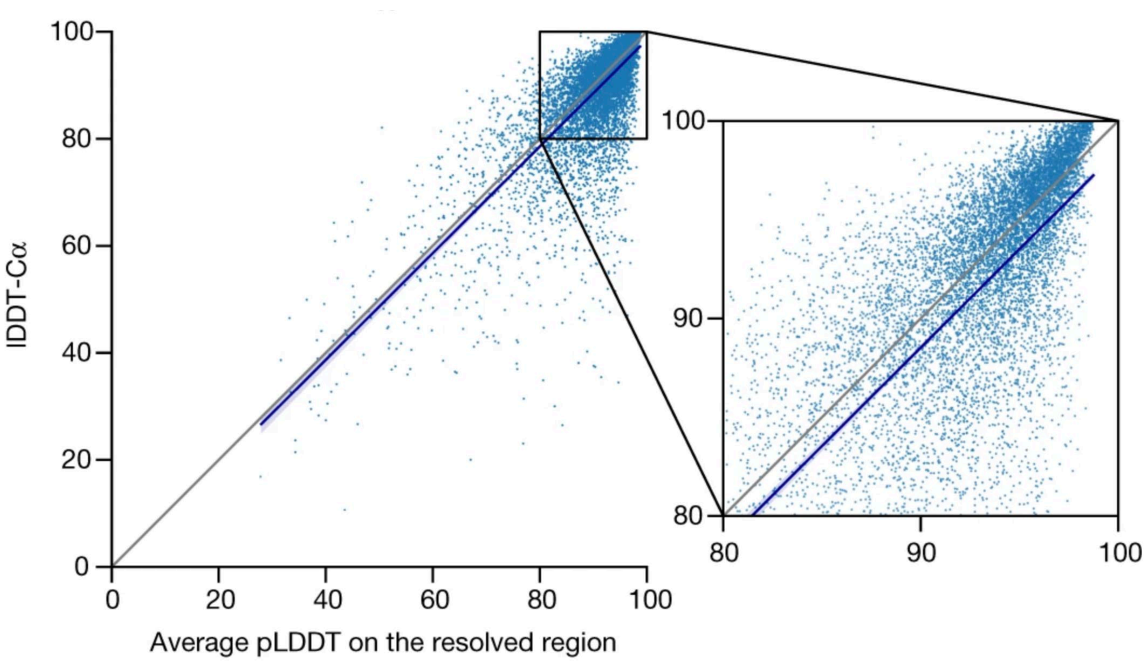
Credit to Dan Rigden

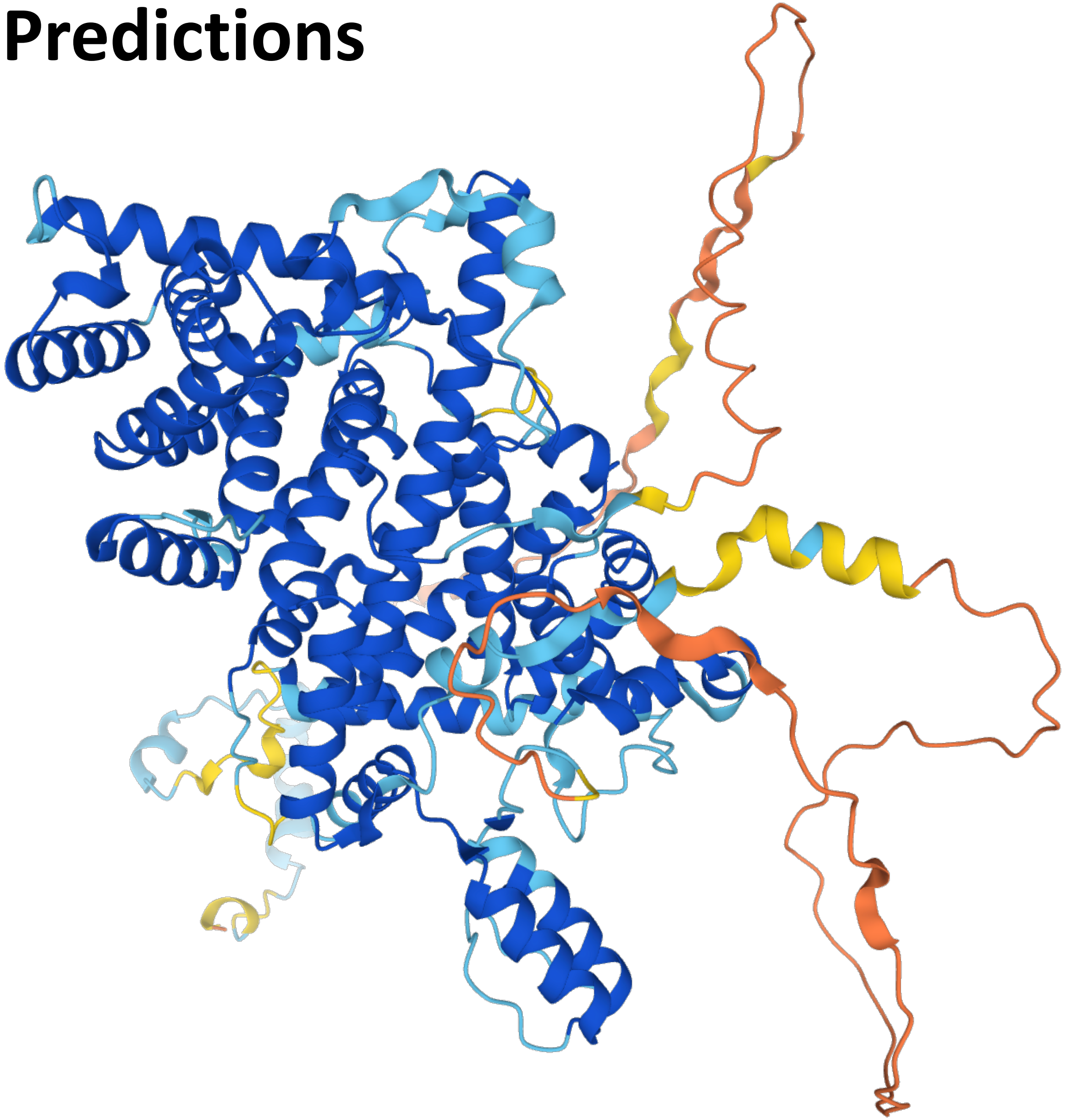
Calibrated Predictions



Jumper et al., *Nature* 2021

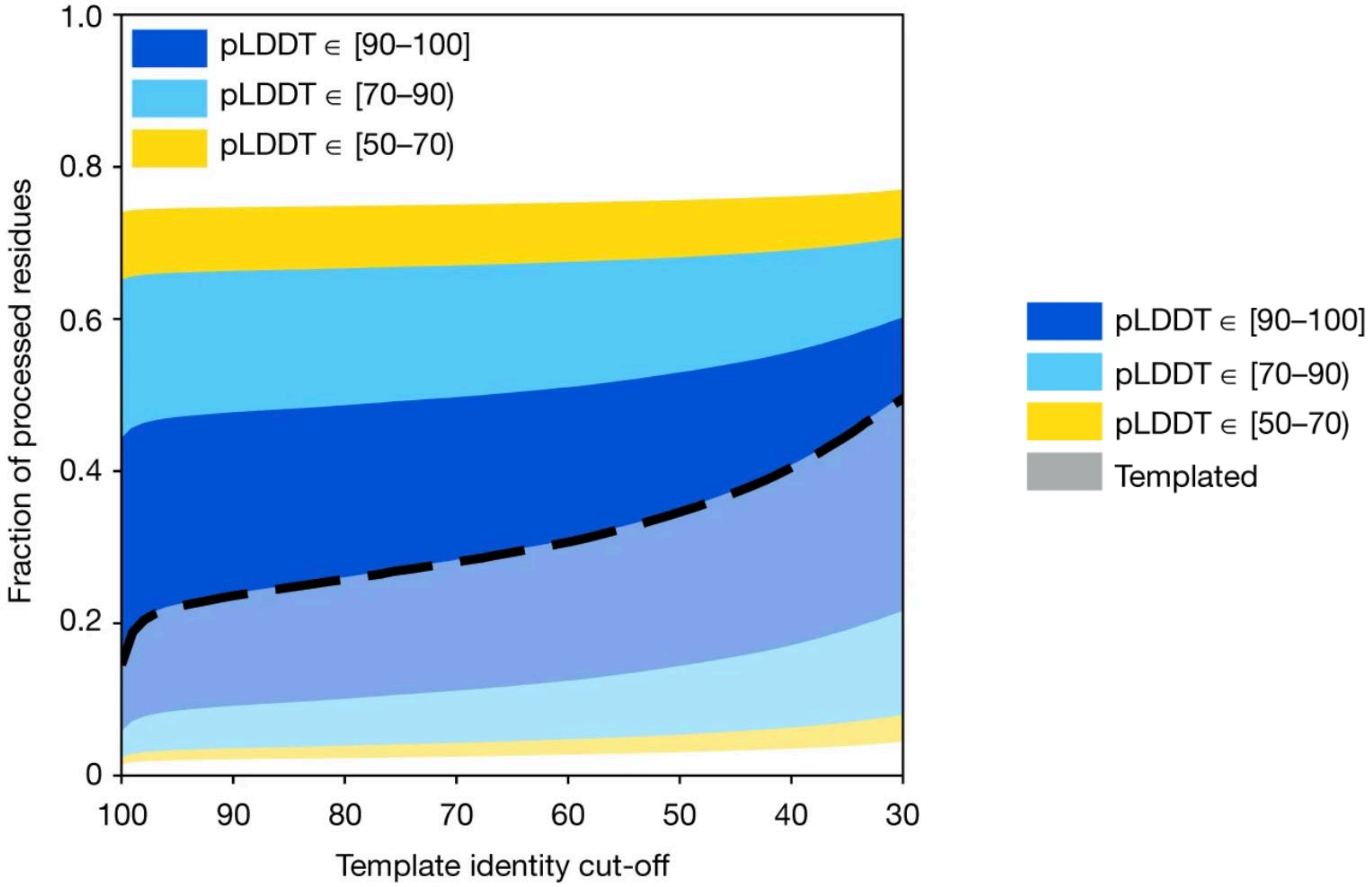
Calibrated Predictions





Jumper et al., *Nature* 2021

Human Proteome Coverage



Tunyasuvunakool et al., *Nature* 2021

OpenFold

Reproducing AlphaFold2 (and beyond)

Why?

Four initial motivations:

1. Full scale retraining (for new applications)
2. Modular components (in PyTorch)
3. Knowledge acquisition / reproduce DeepMind’s results
4. License for commercial use

Why?

Four initial motivations:

1. Full scale retraining (for new applications)

2. Modular components (in PyTorch)

3. Knowledge acquisition / reproduce DeepMind’s results

~~4. License for commercial use~~

Why?

Four initial motivations:

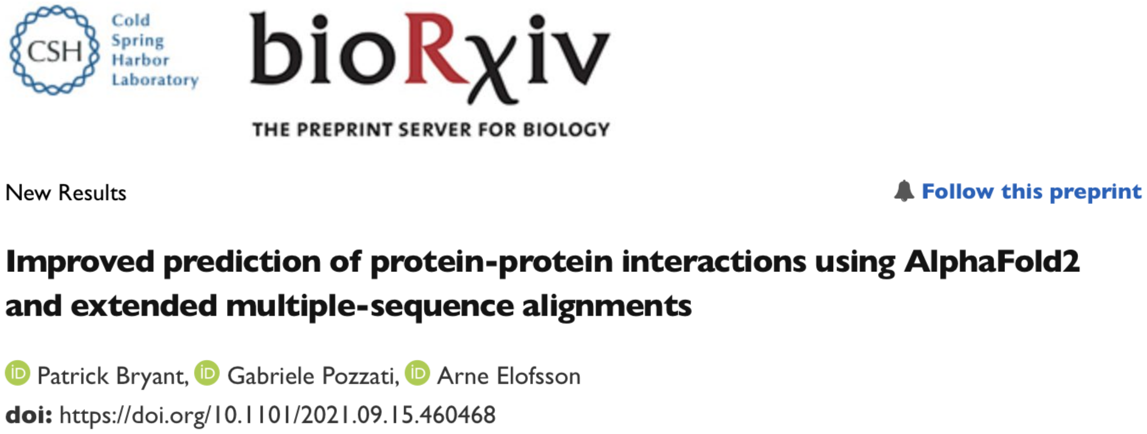
1. Full scale retraining (for new applications)

2. Modular components (in PyTorch)

3. Knowledge acquisition / reproduce DeepMind’s results

~~4. License for commercial use~~

Complexes, complexes, complexes…

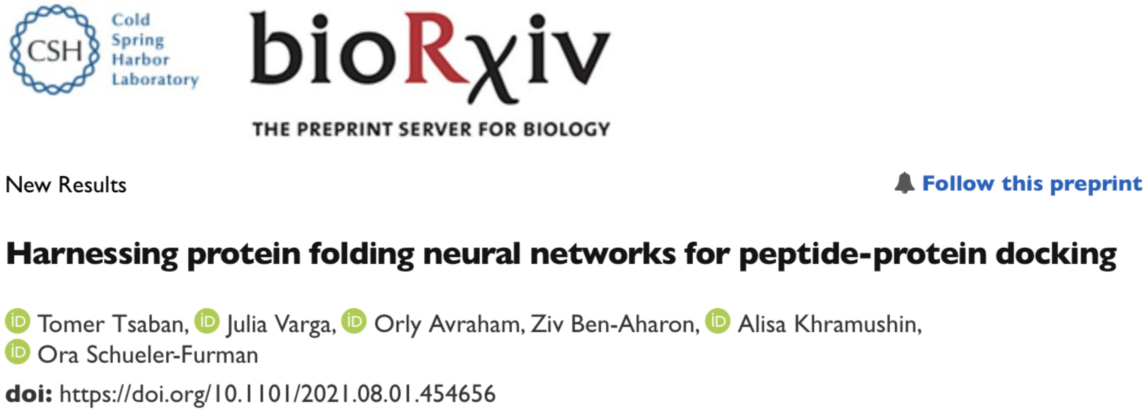

CSH Cold Spring Harbor Laboratory
bioRxiv
THE PREPRINT SERVER FOR BIOLOGY
Follow this preprint

New Results

Improved prediction of protein-protein interactions using AlphaFold2 and extended multiple-sequence alignments

Patrick Bryant, Gabriele Pozzati, Arne Elofsson

doi: https://doi.org/10.1101/2021.09.15.460468

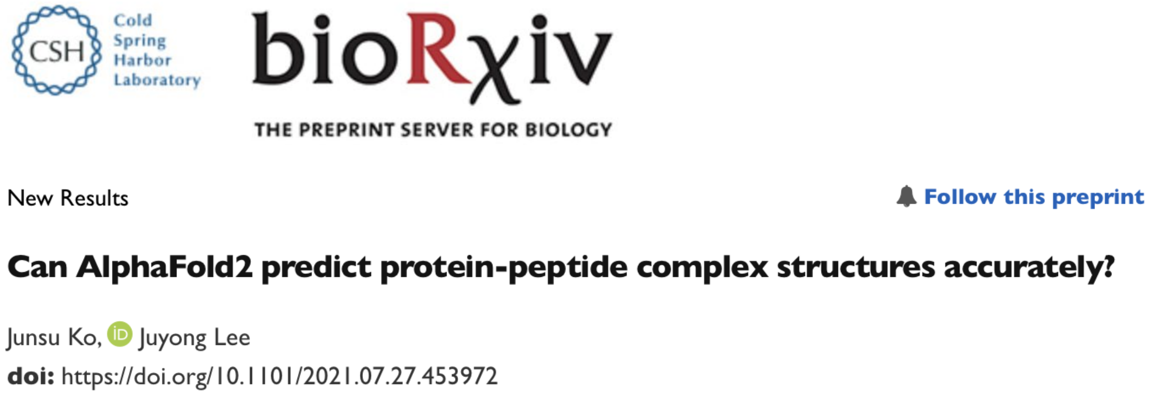

CSH Cold Spring Harbor Laboratory
bioRxiv
THE PREPRINT SERVER FOR BIOLOGY
Follow this preprint

New Results

Harnessing protein folding neural networks for peptide-protein docking

Tomer Tsaban, Julia Varga, Orly Avraham, Ziv Ben-Aharon, Alisa Khramushin, Ora Schueler-Furman

doi: https://doi.org/10.1101/2021.08.01.454656


CSH Cold Spring Harbor Laboratory
bioRxiv
THE PREPRINT SERVER FOR BIOLOGY
Follow this preprint

New Results

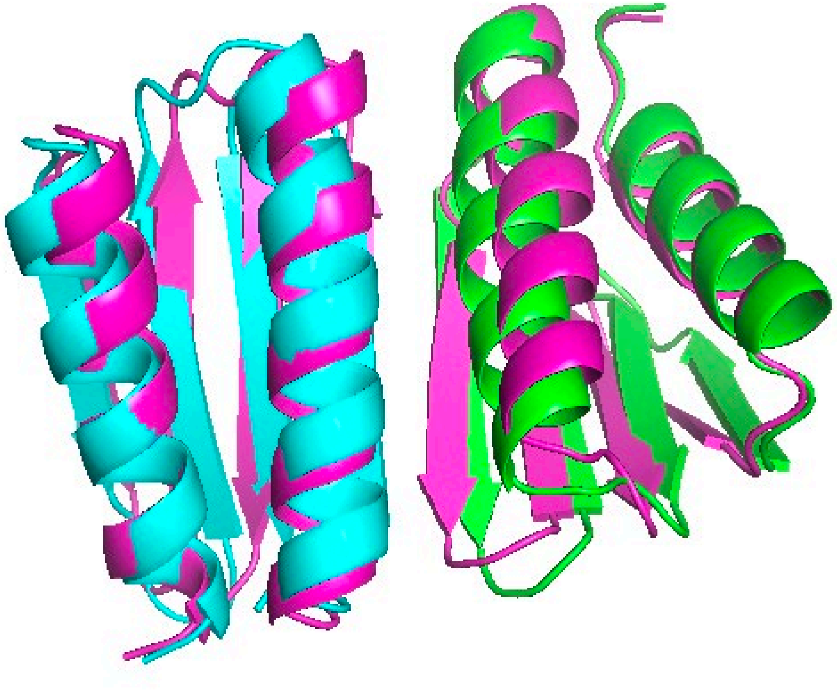
Can AlphaFold2 predict protein-peptide complex structures accurately?

Junsu Ko, Juyong Lee

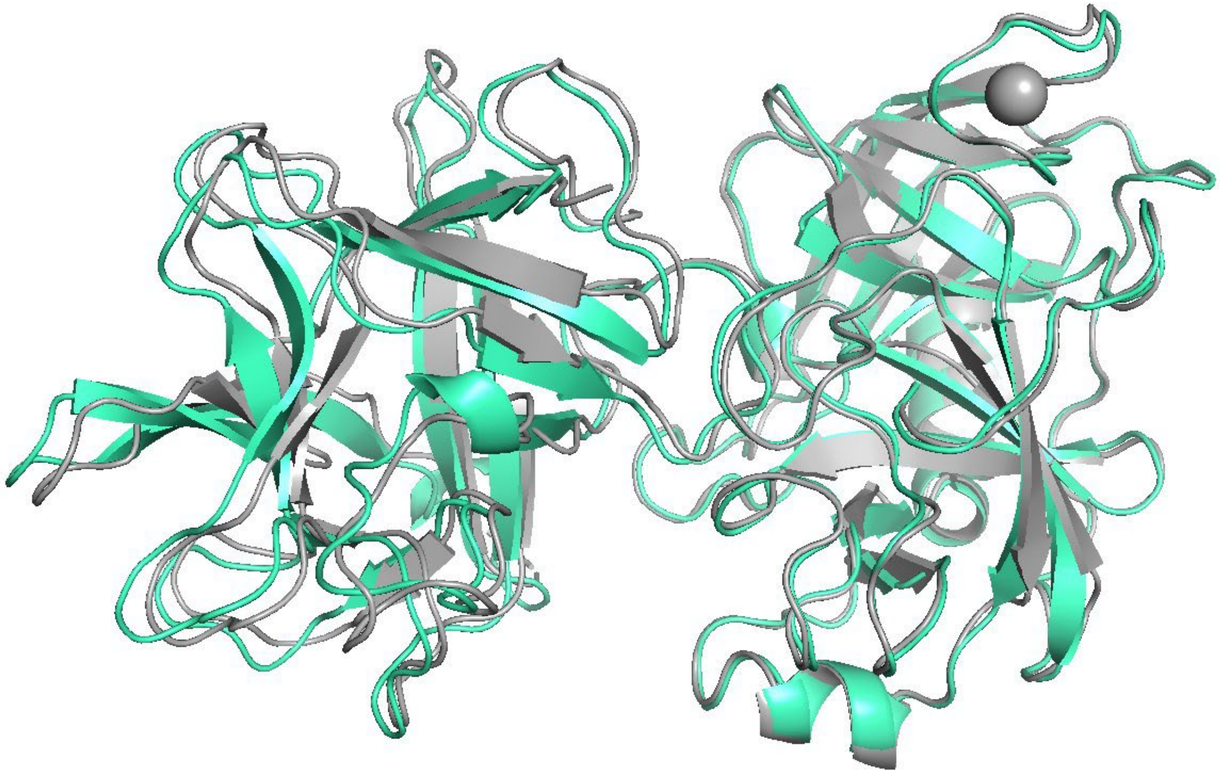
doi: https://doi.org/10.1101/2021.07.27.453972

Complexes, complexes, complexes…

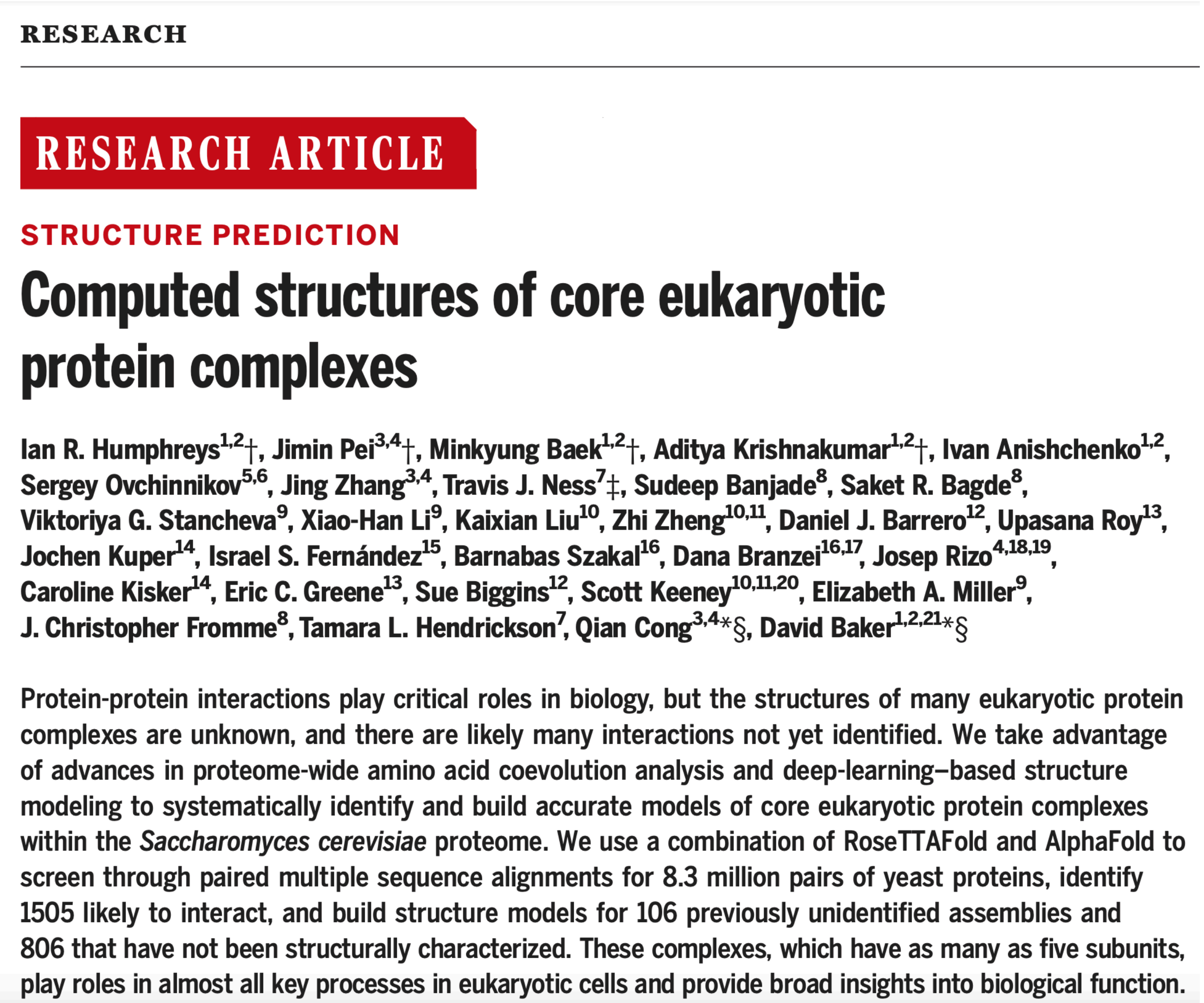
Basic principle: feed AF2 a concatenated sequence (AF2 unchanged)



Credit: Minkyung Baek



Credit: Dzmitry Padhorny



RESEARCH

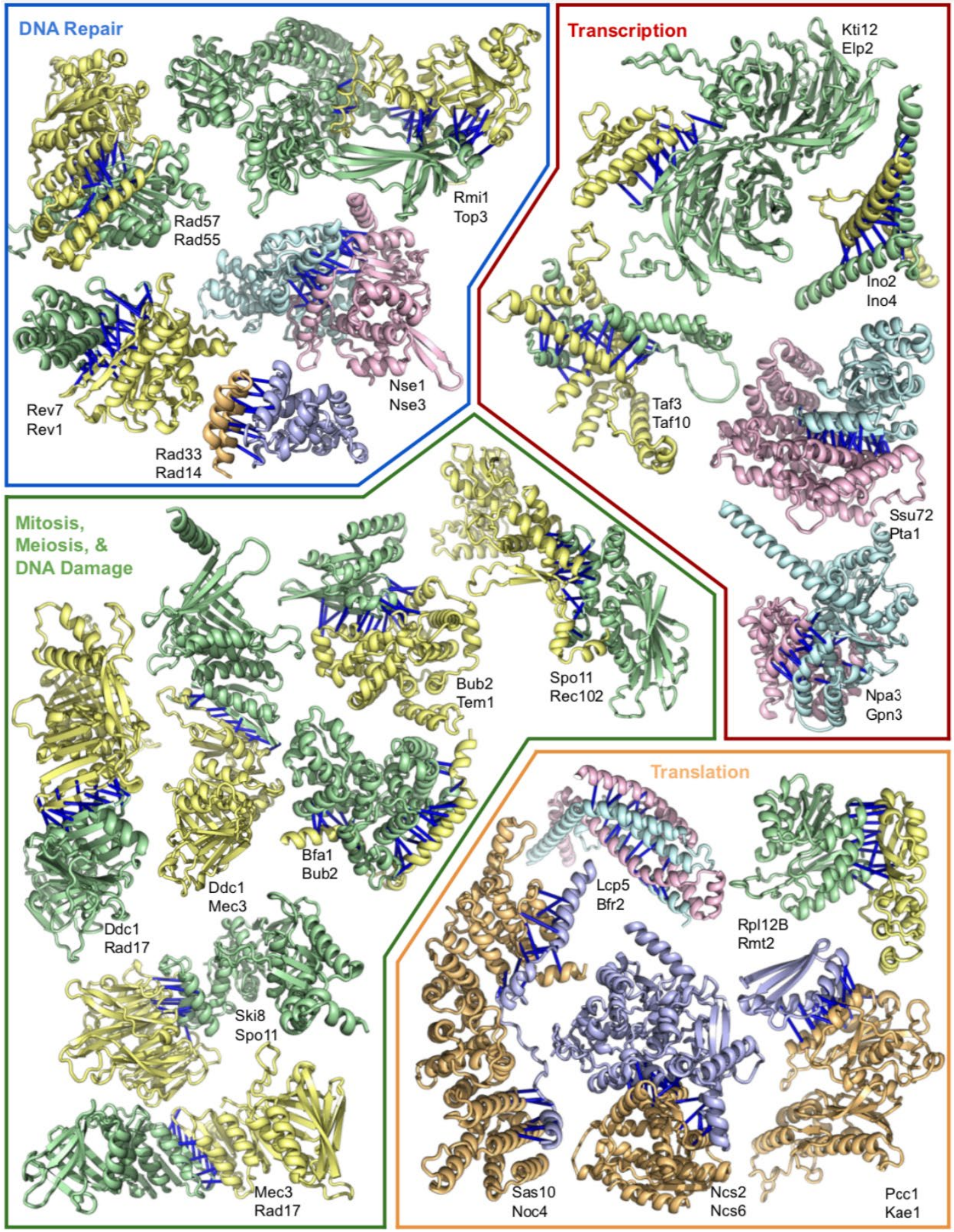
RESEARCH ARTICLE

STRUCTURE PREDICTION

Computed structures of core eukaryotic protein complexes

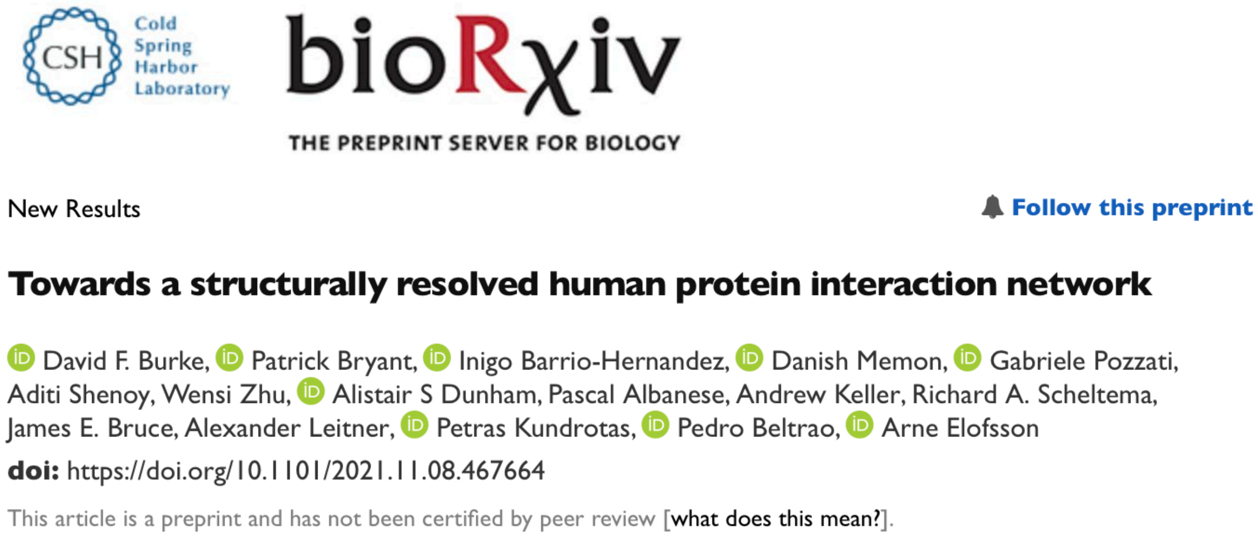
Ian R. Humphreys[1,2]†, Jimin Pei[3,4]†, Minkyung Baek[1,2]†, Aditya Krishnakumar[1,2]†, Ivan Anishchenko[1,2], Sergey Ovchinnikov[5,6], Jing Zhang[3,4], Travis J. Ness[7]‡, Sudeep Banjade[8], Saket R. Bagde[8], Viktoriya G. Stancheva[9], Xiao-Han Li[9], Kaixian Liu[10], Zhi Zheng[10,11], Daniel J. Barrero[12], Upasana Roy[13], Jochen Kuper[14], Israel S. Fernández[15], Barnabas Szakal[16], Dana Branzei[16,17], Josep Rizo[4,18,19], Caroline Kisker[14], Eric C. Greene[13], Sue Biggins[12], Scott Keeney[10,11,20], Elizabeth A. Miller[9], J. Christopher Fromme[8], Tamara L. Hendrickson[7], Qian Cong[3,4]*§, David Baker[1,2,21]*§

Protein-protein interactions play critical roles in biology, but the structures of many eukaryotic protein complexes are unknown, and there are likely many interactions not yet identified. We take advantage of advances in proteome-wide amino acid coevolution analysis and deep-learning–based structure modeling to systematically identify and build accurate models of core eukaryotic protein complexes within the *Saccharomyces cerevisiae* proteome. We use a combination of RoseTTAFold and AlphaFold to screen through paired multiple sequence alignments for 8.3 million pairs of yeast proteins, identify 1505 likely to interact, and build structure models for 106 previously unidentified assemblies and 806 that have not been structurally characterized. These complexes, which have as many as five subunits, play roles in almost all key processes in eukaryotic cells and provide broad insights into biological function.



Humphreys et al., *Science* 2021

Complexes, complexes, complexes…



Cold Spring Harbor Laboratory

bioRχiv

THE PREPRINT SERVER FOR BIOLOGY

New Results

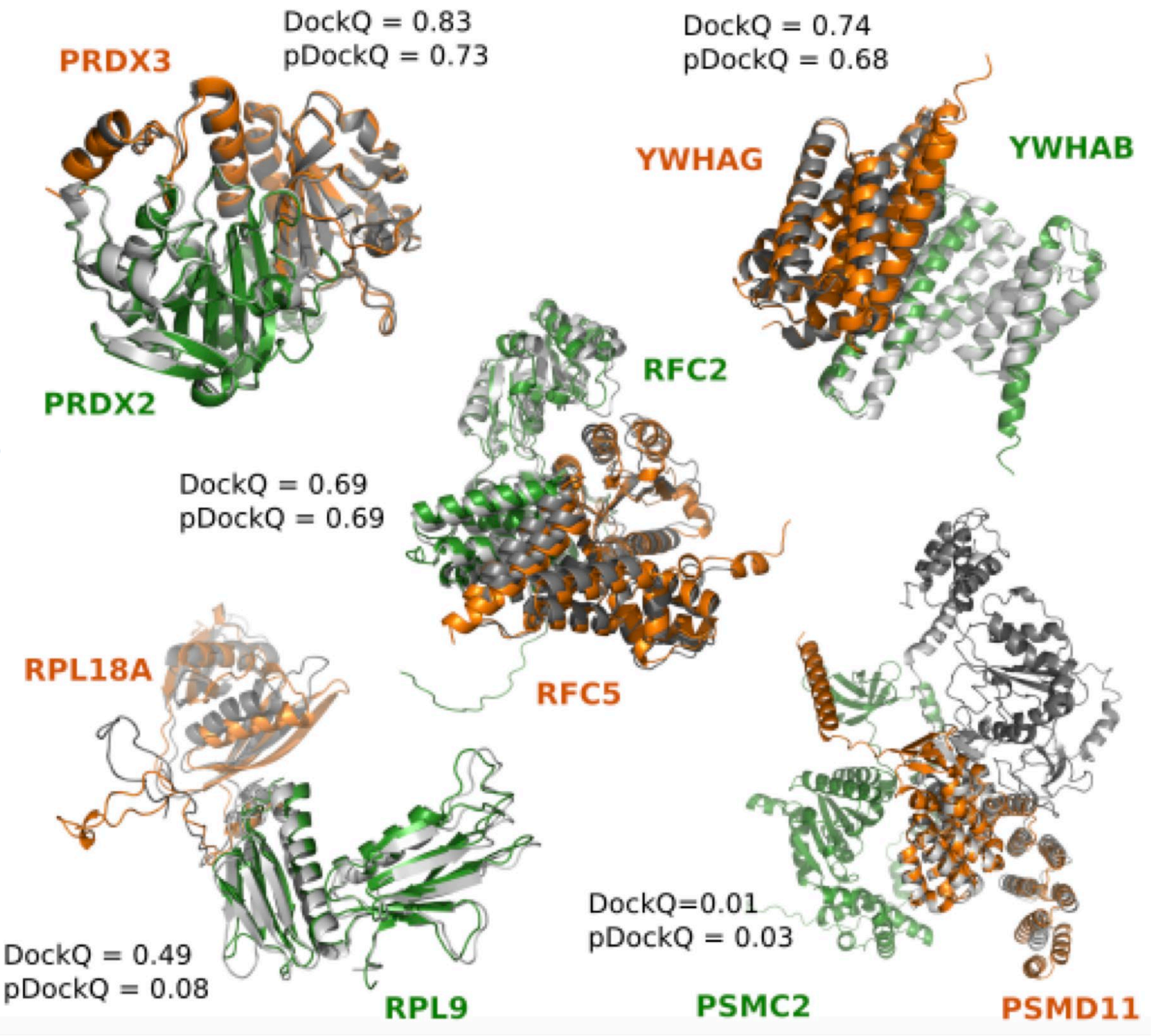
Follow this preprint

Towards a structurally resolved human protein interaction network

David F. Burke, Patrick Bryant, Inigo Barrio-Hernandez, Danish Memon, Gabriele Pozzati, Aditi Shenoy, Wensi Zhu, Alistair S Dunham, Pascal Albanese, Andrew Keller, Richard A. Scheltema, James E. Bruce, Alexander Leitner, Petras Kundrotas, Pedro Beltrao, Arne Elofsson

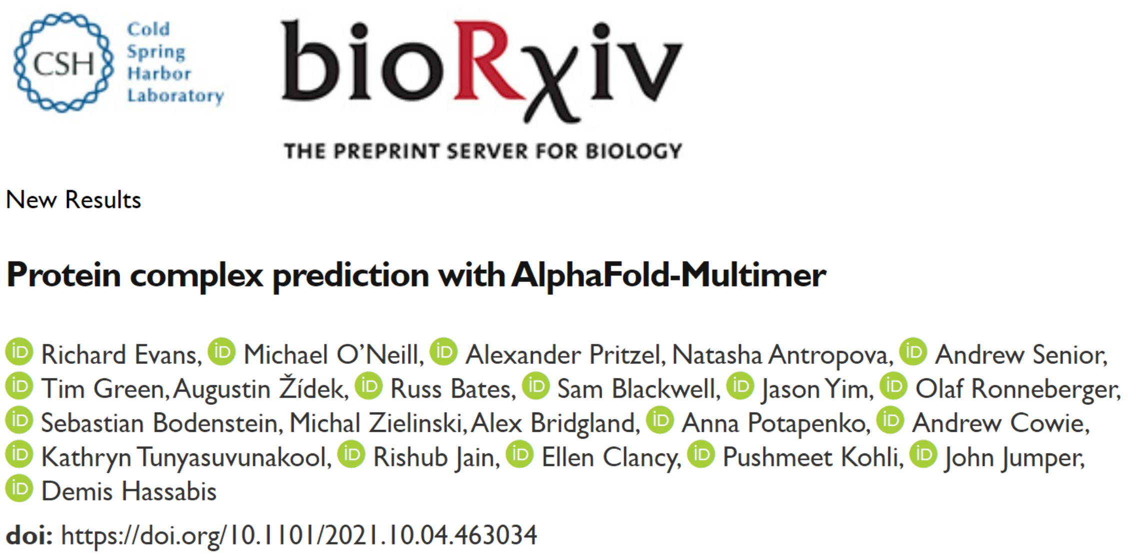
doi: https://doi.org/10.1101/2021.11.08.467664

This article is a preprint and has not been certified by peer review [what does this mean?].



Burke et al., *bioRxiv* 2021

AlphaFold2-Multimer



Cold Spring Harbor Laboratory

bioRxiv

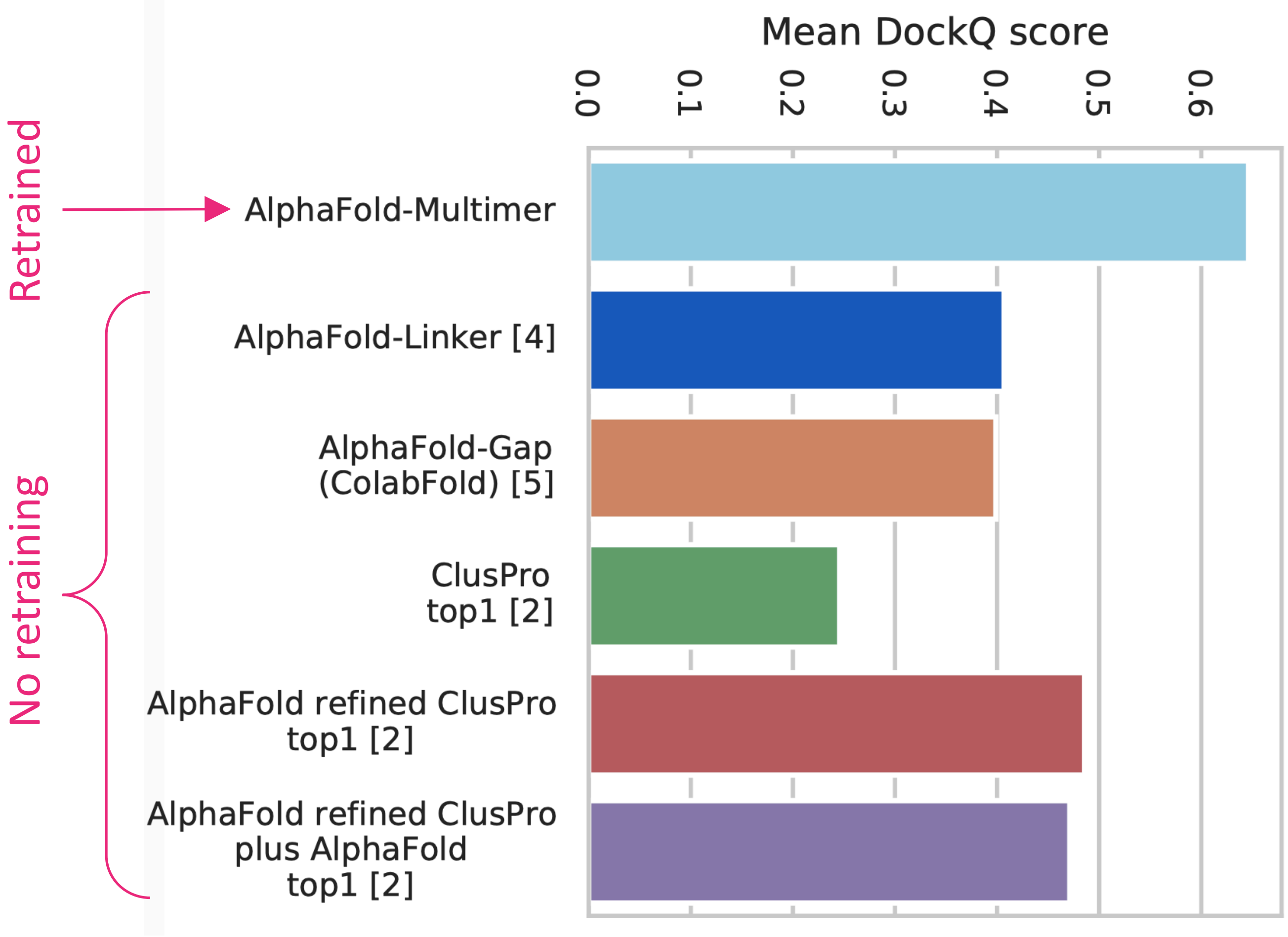
THE PREPRINT SERVER FOR BIOLOGY

New Results

Protein complex prediction with AlphaFold-Multimer

Richard Evans, Michael O'Neill, Alexander Pritzel, Natasha Antropova, Andrew Senior, Tim Green, Augustin Žídek, Russ Bates, Sam Blackwell, Jason Yim, Olaf Ronneberger, Sebastian Bodenstein, Michal Zielinski, Alex Bridgland, Anna Potapenko, Andrew Cowie, Kathryn Tunyasuvunakool, Rishub Jain, Ellen Clancy, Pushmeet Kohli, John Jumper, Demis Hassabis

doi: https://doi.org/10.1101/2021.10.04.463034



Evans et al., *bioRxiv* 2021

Why?

Four initial motivations:

1. Full scale retraining (for new applications)

2. Modular components (in PyTorch)

3. Knowledge acquisition / reproduce DeepMind’s results

~~4. License for commercial use~~

Why?

Four initial motivations:

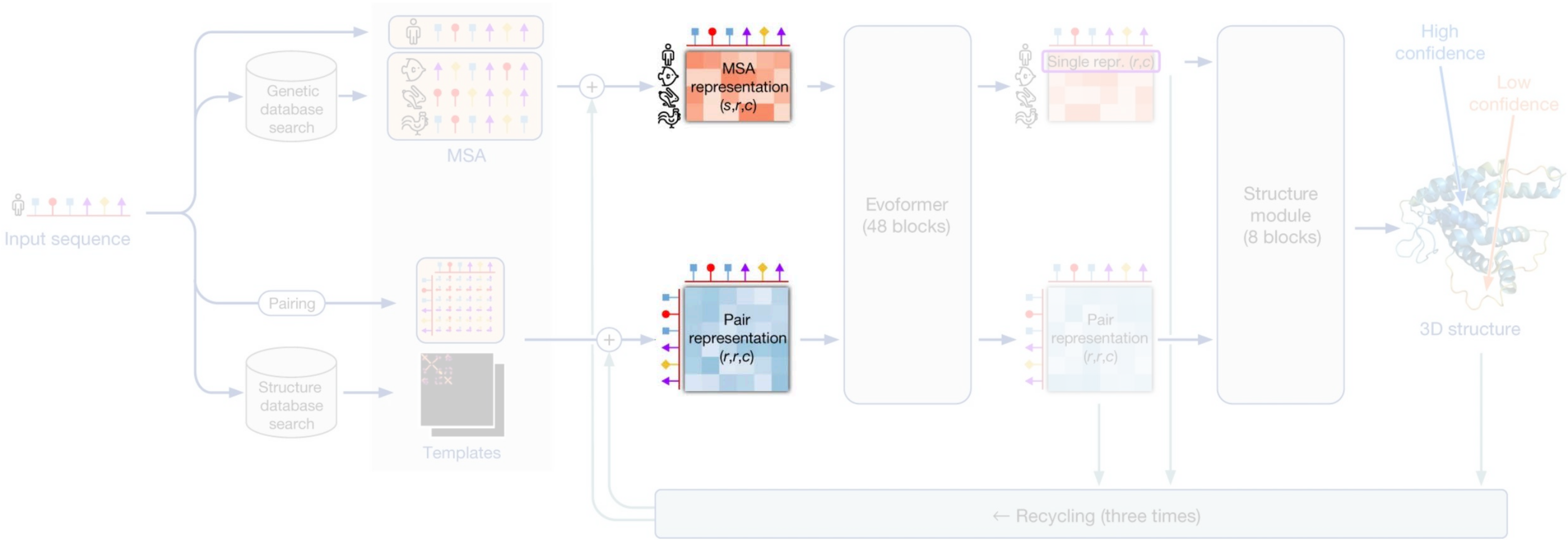
1. Full scale retraining (for new applications)

2. Modular components (in PyTorch)

3. Knowledge acquisition / reproduce DeepMind’s results

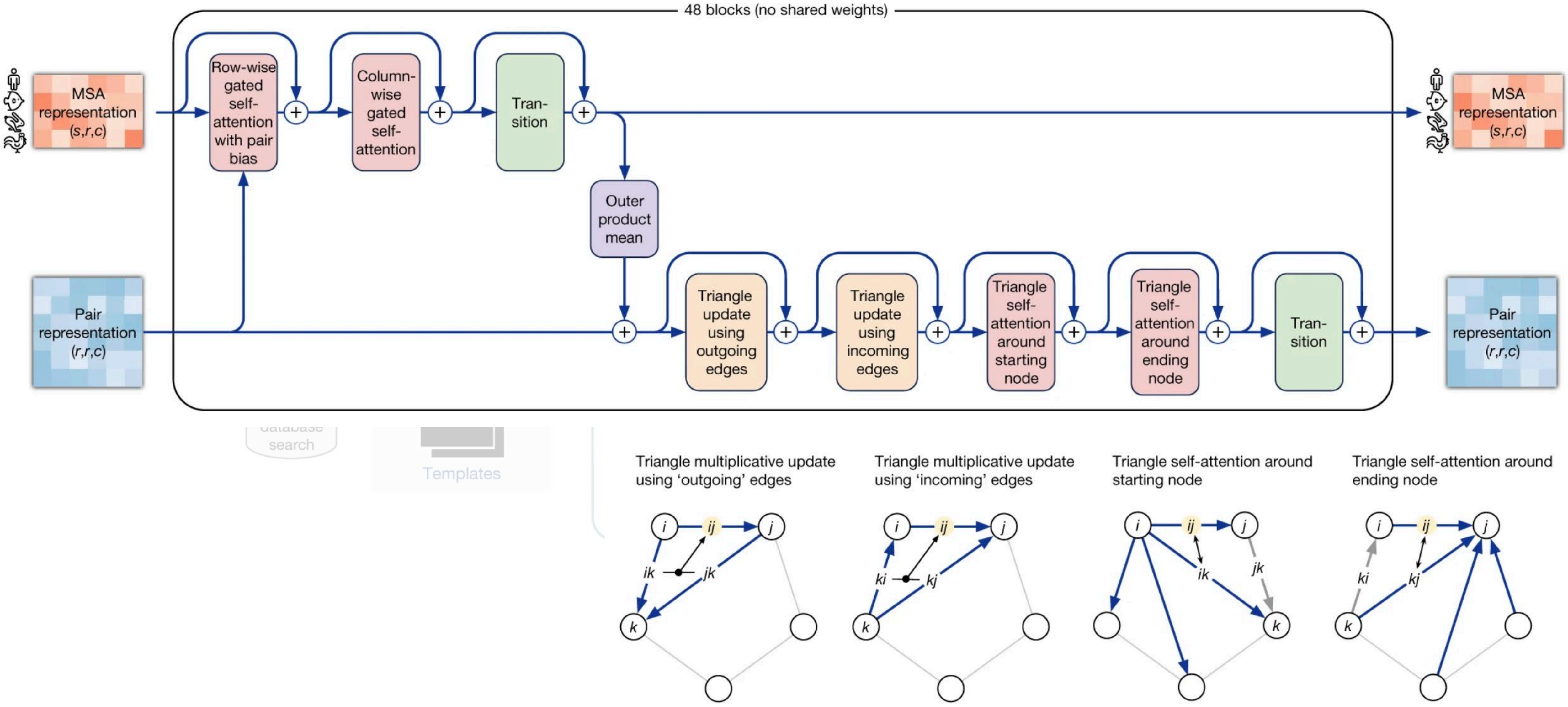
~~4. License for commercial use~~

AlphaFold2 Architecture


Input sequence
Genetic database search
MSA
Pairing
Structure database search
Templates
MSA representation (s,r,c)
Pair representation (r,r,c)
Evoformer (48 blocks)
Single repr. (r,c)
Pair representation (r,r,c)
Structure module (8 blocks)
High confidence
Low confidence
3D structure
← Recycling (three times)

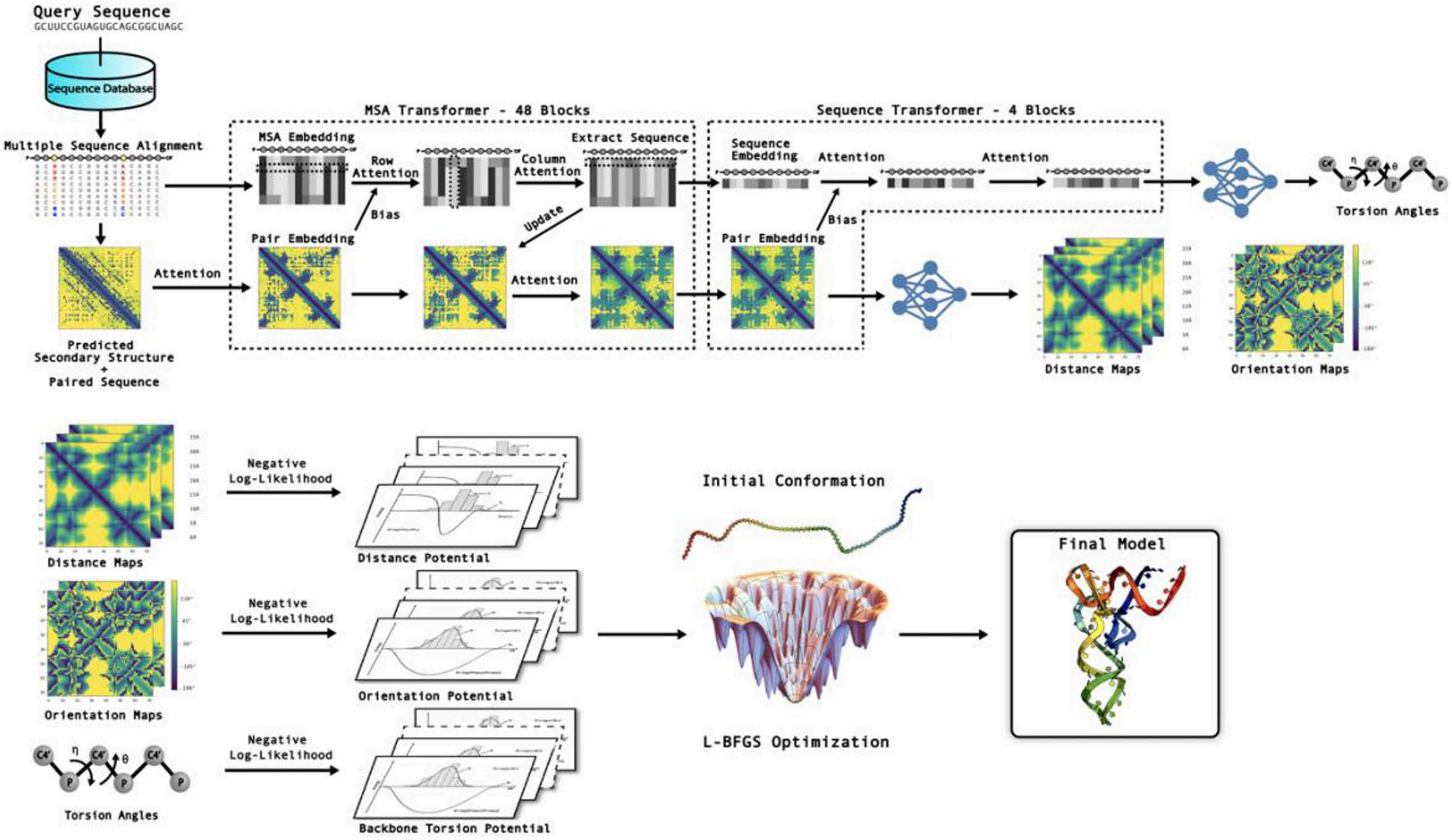
Jumper et al., *Nature* 2021

AlphaFold2 Architecture


48 blocks (no shared weights)
MSA representation (s,r,c)
Row-wise gated self-attention with pair bias
Column-wise gated self-attention
Tran-sition
Outer product mean
MSA representation (s,r,c)
Pair representation (r,r,c)
Triangle update using outgoing edges
Triangle update using incoming edges
Triangle self-attention around starting node
Triangle self-attention around ending node
Tran-sition
Pair representation (r,r,c)
search
Templates
Triangle multiplicative update using 'outgoing' edges
Triangle multiplicative update using 'incoming' edges
Triangle self-attention around starting node
Triangle self-attention around ending node
i
ij
j
ik
jk
k
ki
kj

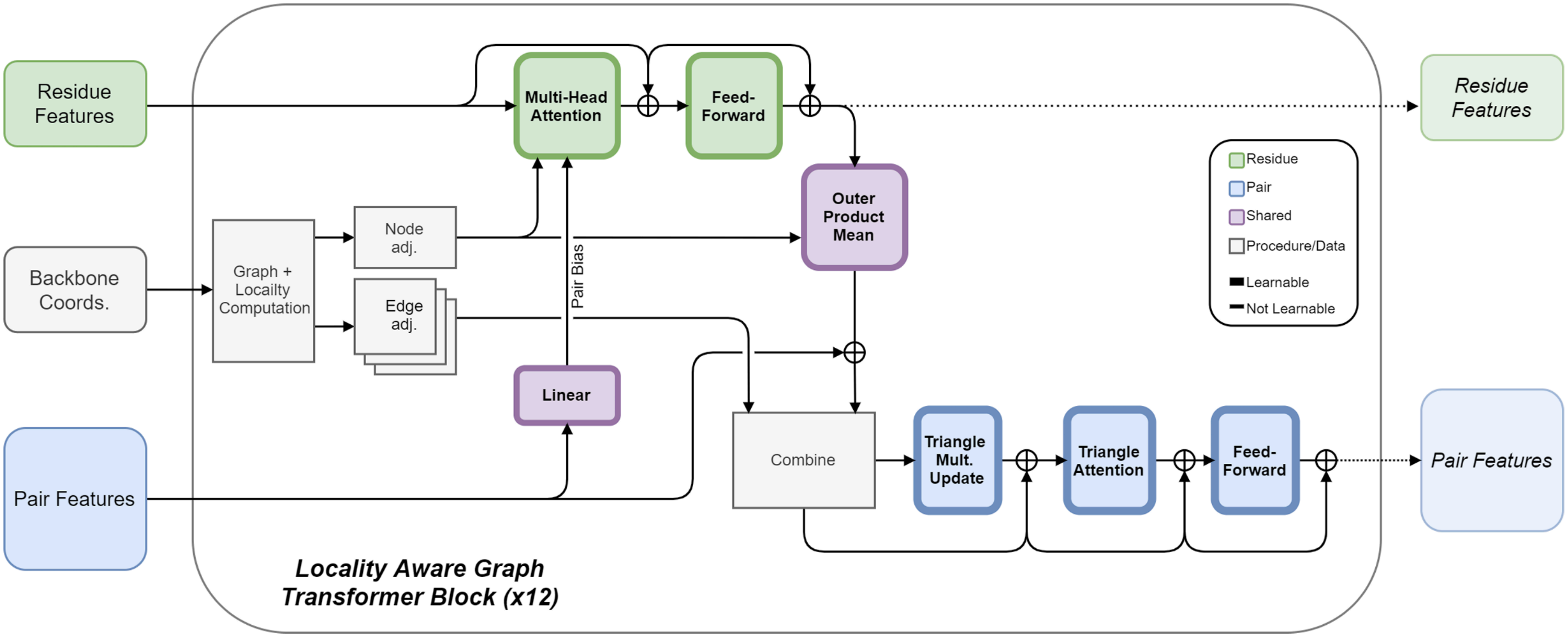
Jumper et al., *Nature* 2021

Reuse of AF2 Components (RNA Structure Prediction)



Robin Pearcea, Gilbert S. Omenna, Yang Zhang, bioRxiv 2022

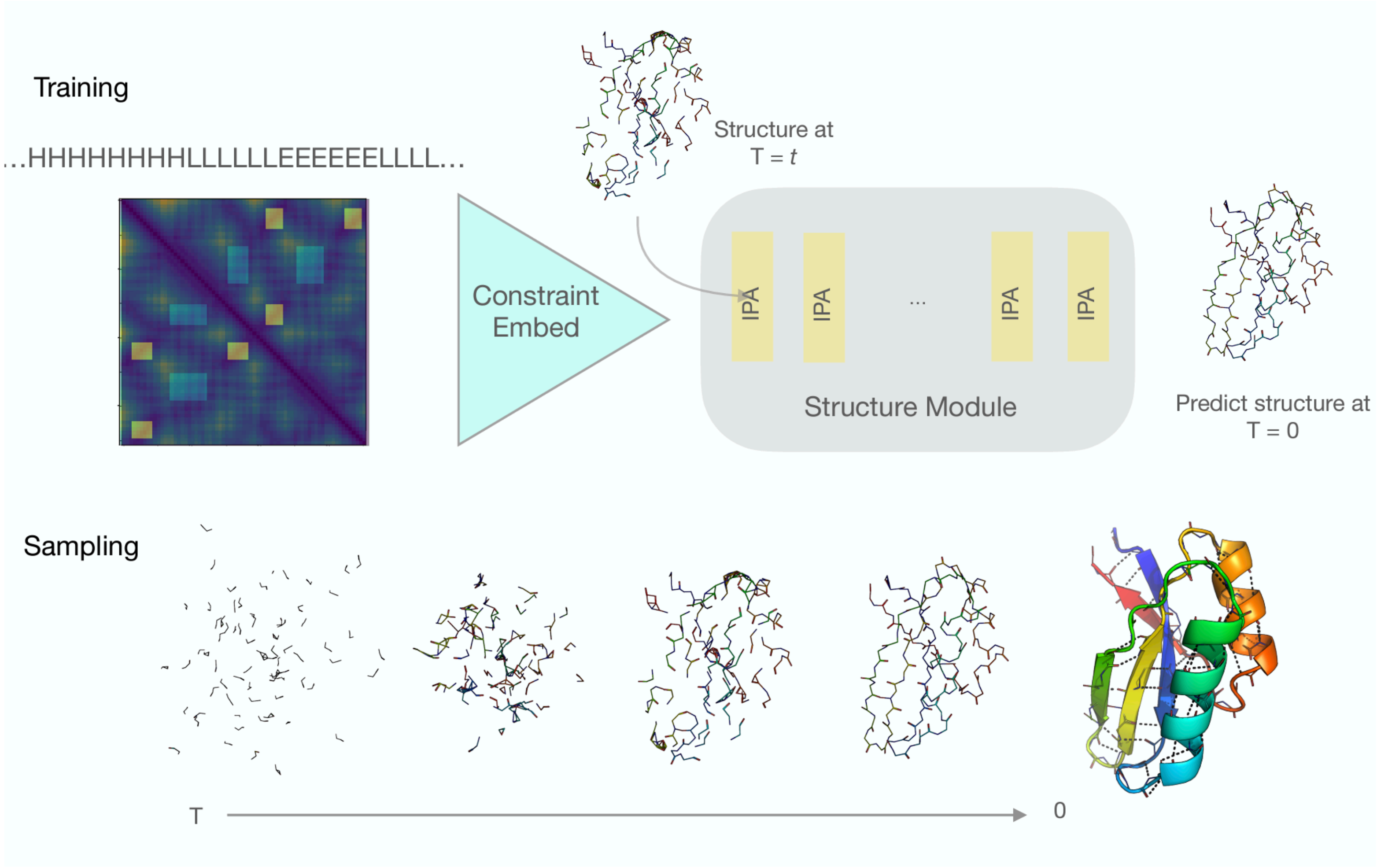
Reuse of AF2 Components (Inverse Folding, Refinement)



Matt McPartlon, Jinbo Xu, bioRxiv 2022

Matt McPartlon, Ben Lai, Jinbo Xu, bioRxiv 2022

Reuse of AF2 Components (Structure Generation)



Namrata Anand, Tudor Achim, arXiv 2022

Why?

Four initial motivations:

1. Full scale retraining (for new applications)
2. Modular components (in PyTorch)
3. Knowledge acquisition / reproduce DeepMind's results
4. ~~License for commercial use~~

Why?

Four initial motivations:

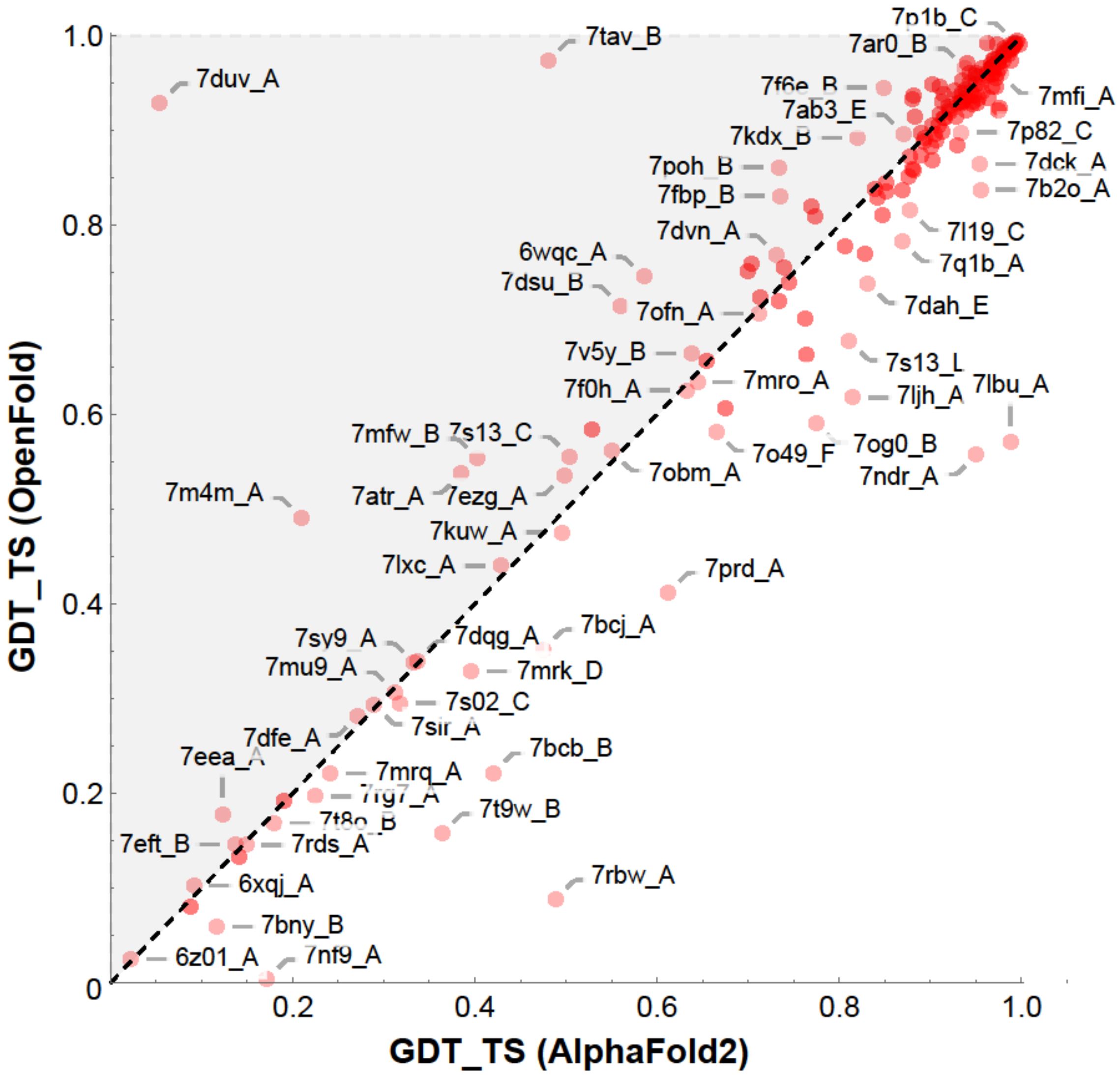
1. Full scale retraining (for new applications)

2. Modular components (in PyTorch)

3. Knowledge acquisition / reproduce DeepMind’s results

~~4. License for commercial use~~

How well does it work?



On held-out validation set (CAMEO)

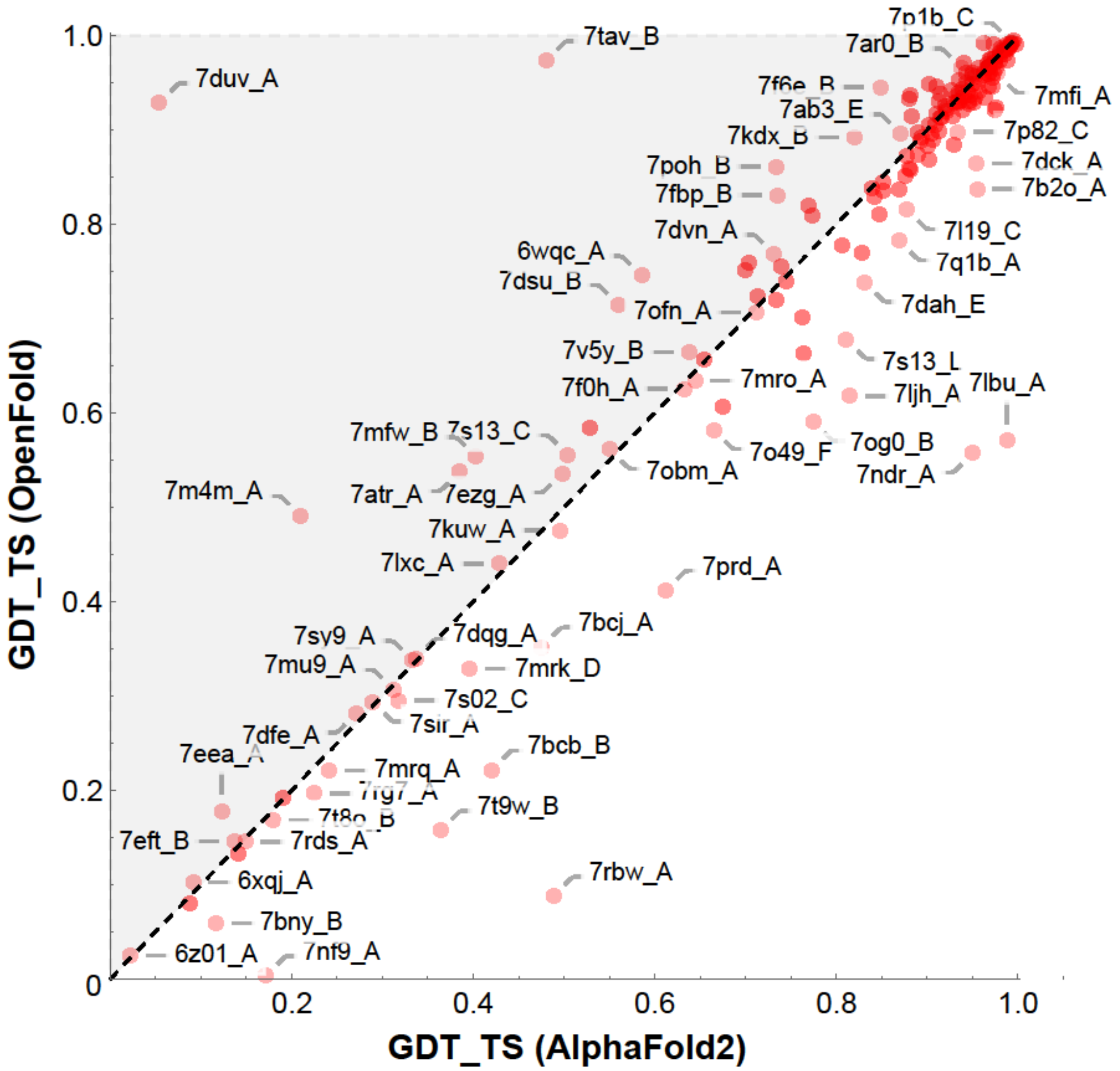
AlphaFold2 Mean GDT_TS: **77.8**

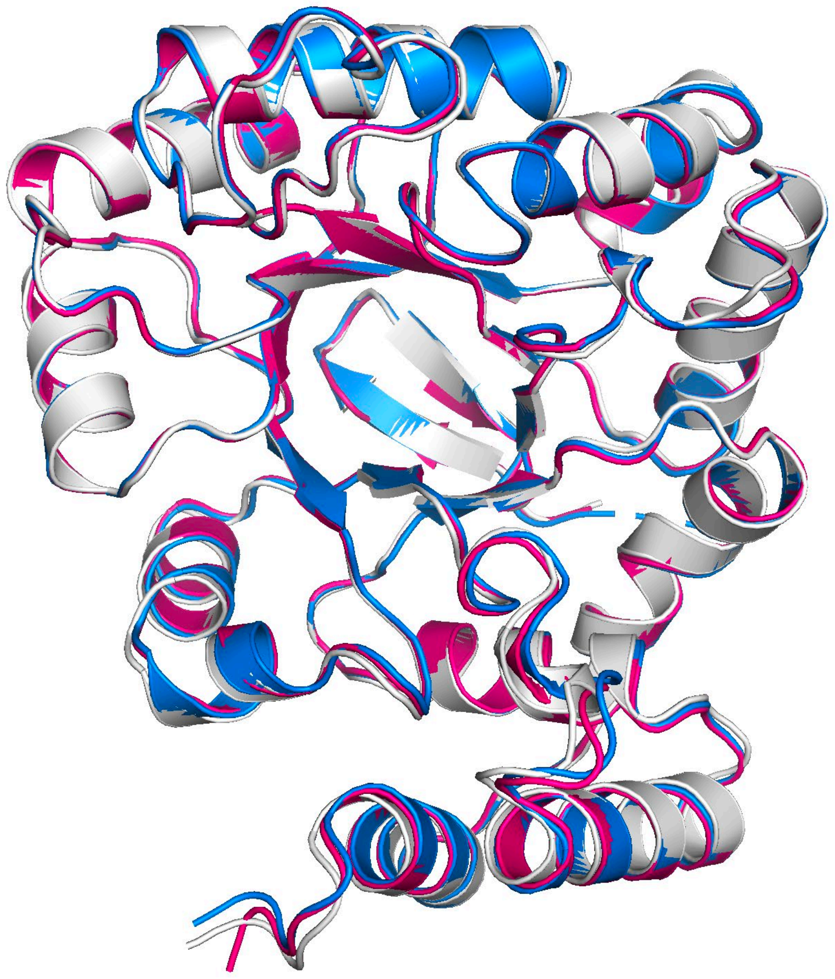
AlphaFold2 Mean RMSD: 2.25Å

OpenFold Mean GDT_TS: 77.3

OpenFold Mean RMSD: **2.22Å**

How well does it work?





How well does it work? (inference characteristics)

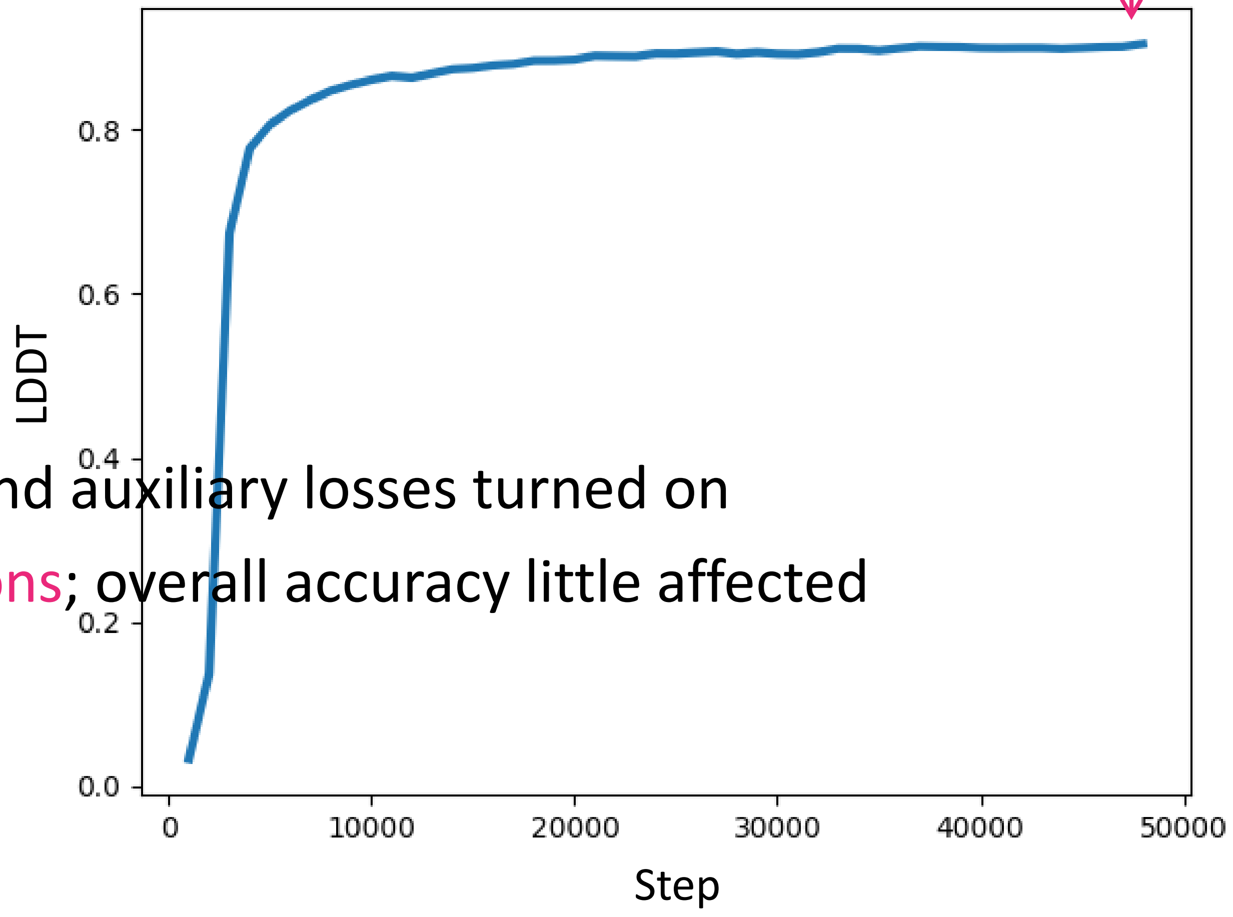
1. Faster inference than AF2 (up to 3X faster on proteins < 1,000 residues)
2. Low-memory attention (Rabe and Staats 2021)
   - Inference on longer chains than AF2 (4,000+ residues on mortal GPUs)
   - Both monomer and multimer modes (large complexes)
   - Applicable to published AF2 weights
3. Trade speed for memory, inference for longer sequences / complexes
4. Cost is code complexity, *e.g.,*:
   - Original triangle multiplicate update was ~10 lines of code
   - Optimized version is now nearly 400 lines of code

How well does it work? (training characteristics)

1. bfloat16 precision training on A100 GPUs (AF2 trained on TPUs)
2. In-progress float16 precision training (would enable V100 GPUs)
3. Distributed training via DeepSpeed and PyTorch Lightning
4. Custom memory-efficient CUDA kernels for attention
5. Large amounts of precomputed MSAs for self-distillation (>AF2’s)

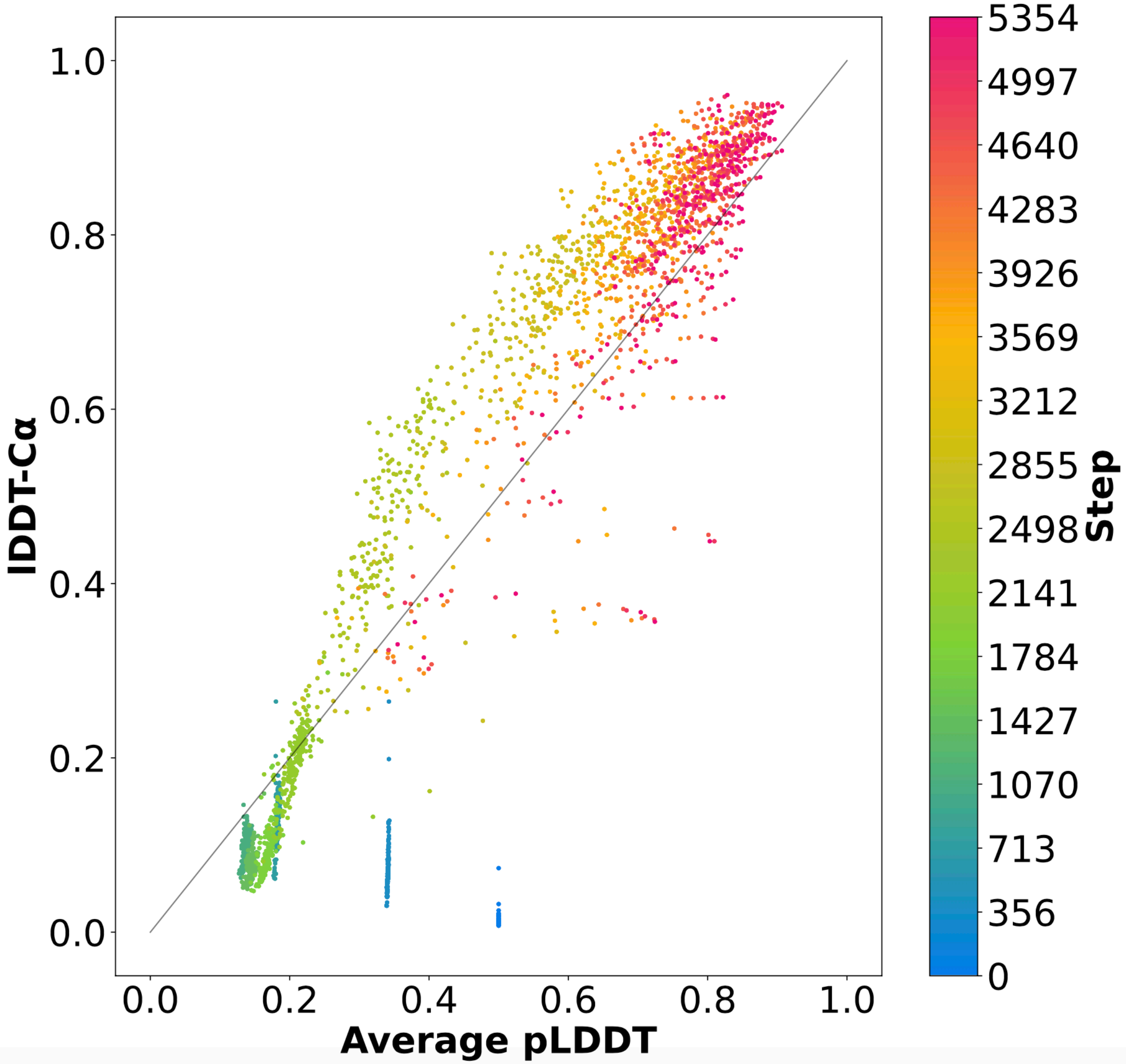
How does it learn? (convergence)

1. Model exhibits fast convergence
    - ≈90% of final IDDT in 2-3 days (44 A100 GPUs)
    - Total training time ≈80 days (inclusive of self-distillation)
2. Fine-tuning stage
    - Crops increased to 384 residues and auxiliary losses turned on
    - Primarily resolves physical violations; overall accuracy little affected

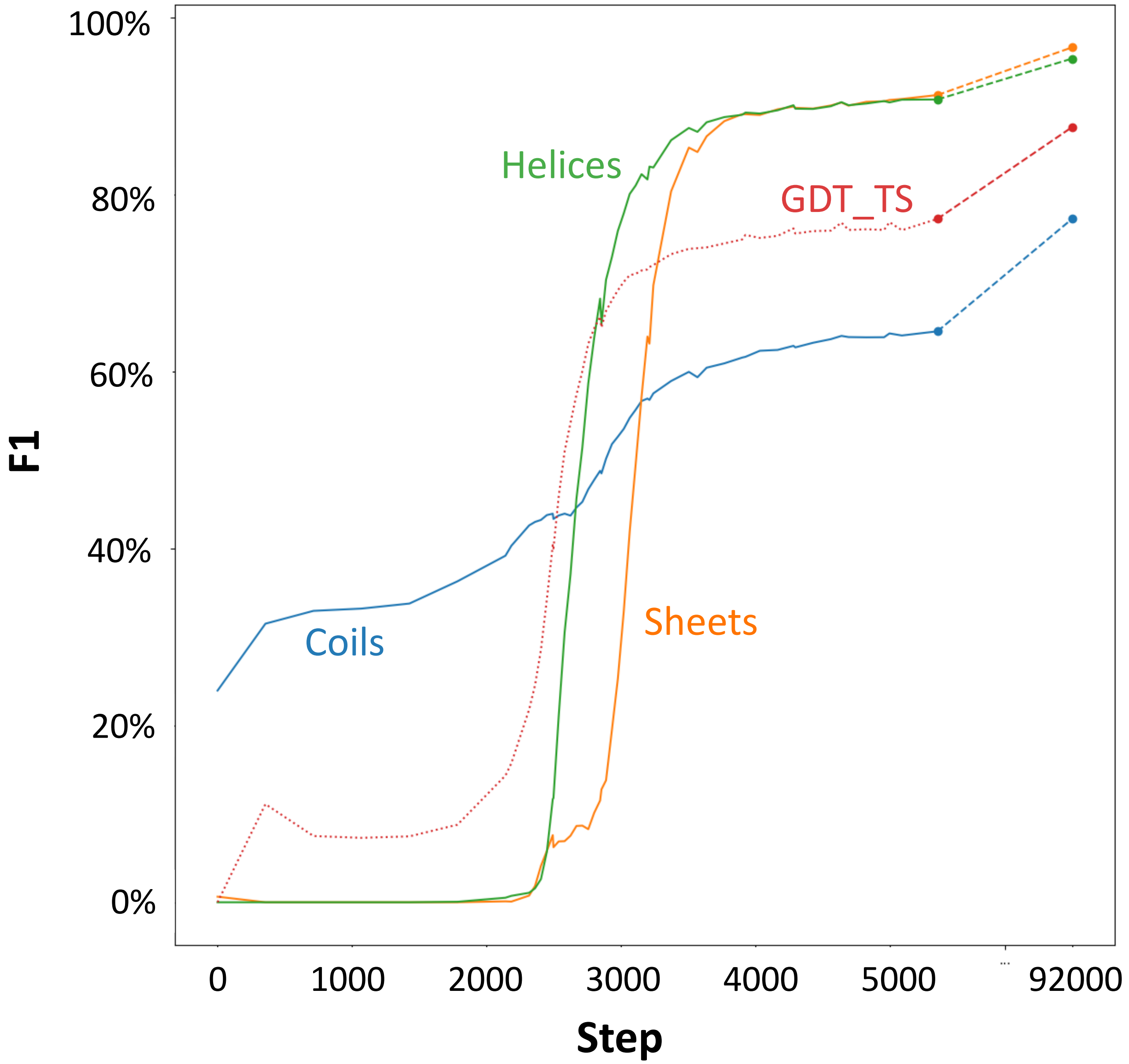


IDDT = 0.9

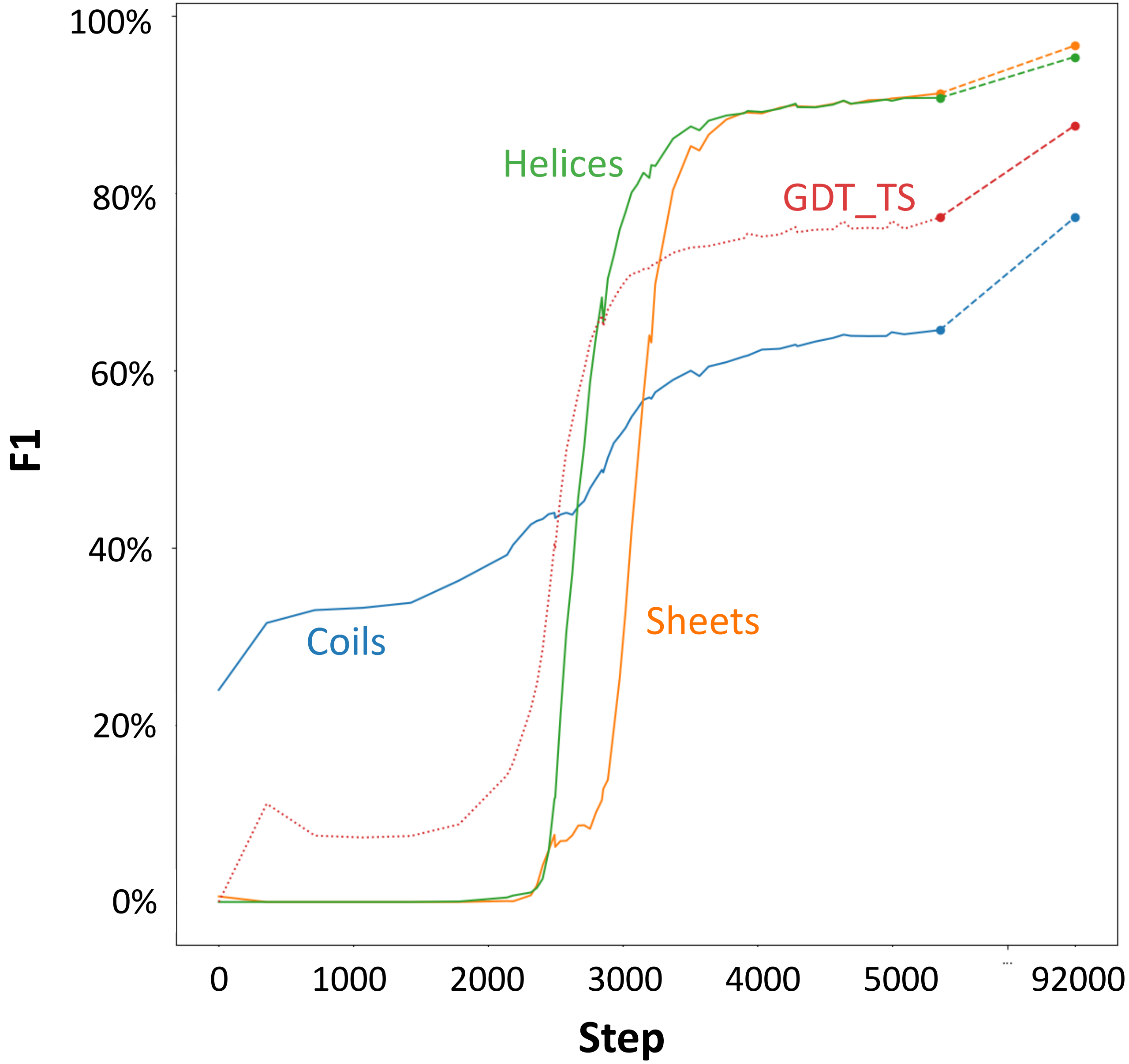
How does it learn? (self-assessment)

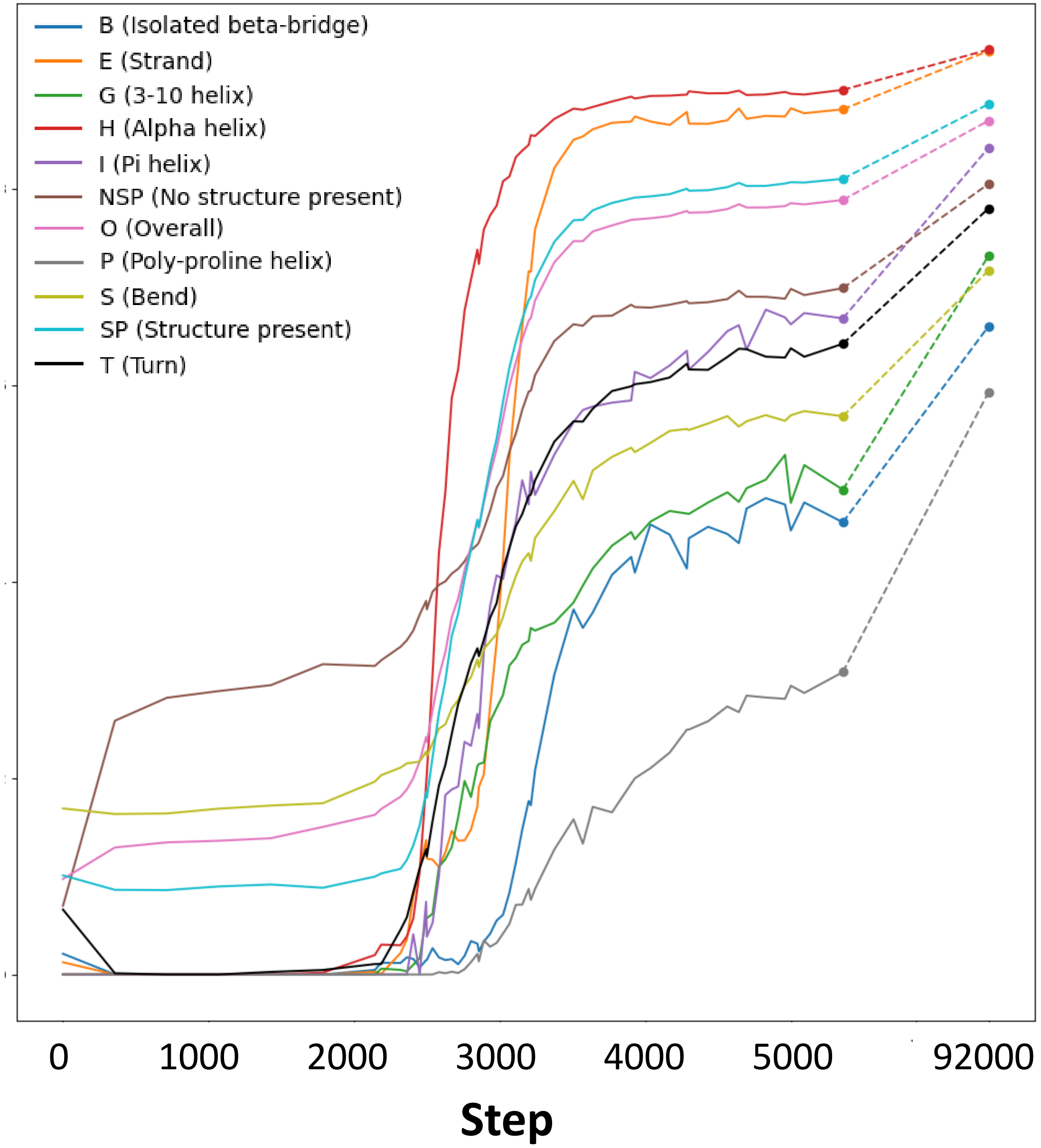


How does it learn? (secondary structure acquisition)

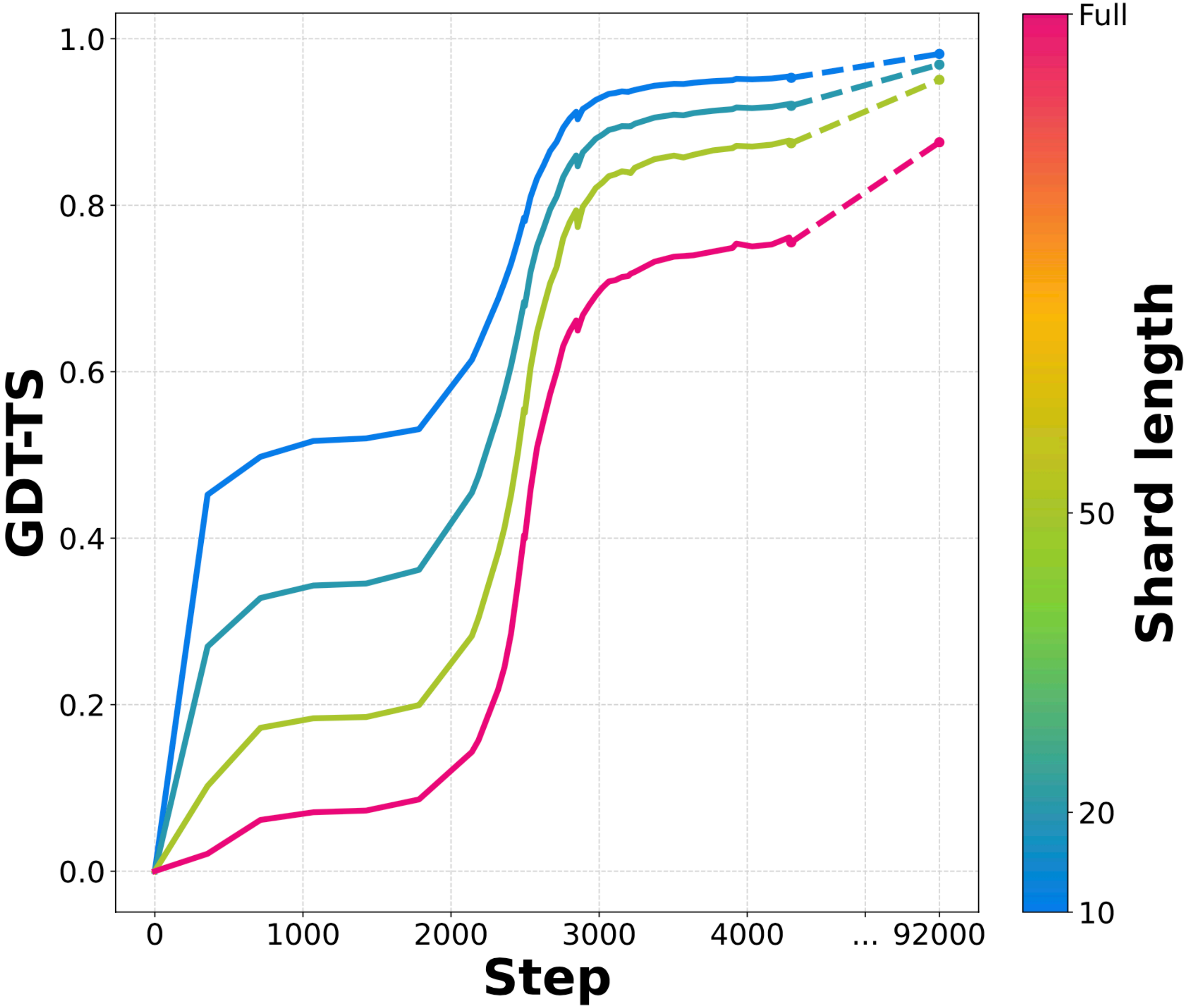


How does it learn? (secondary structure acquisition)

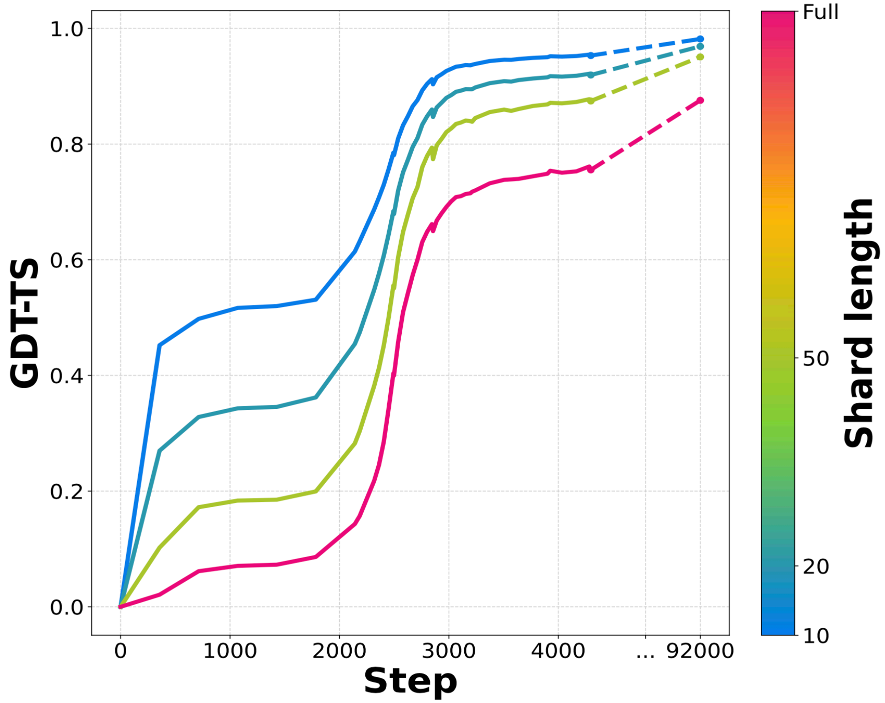


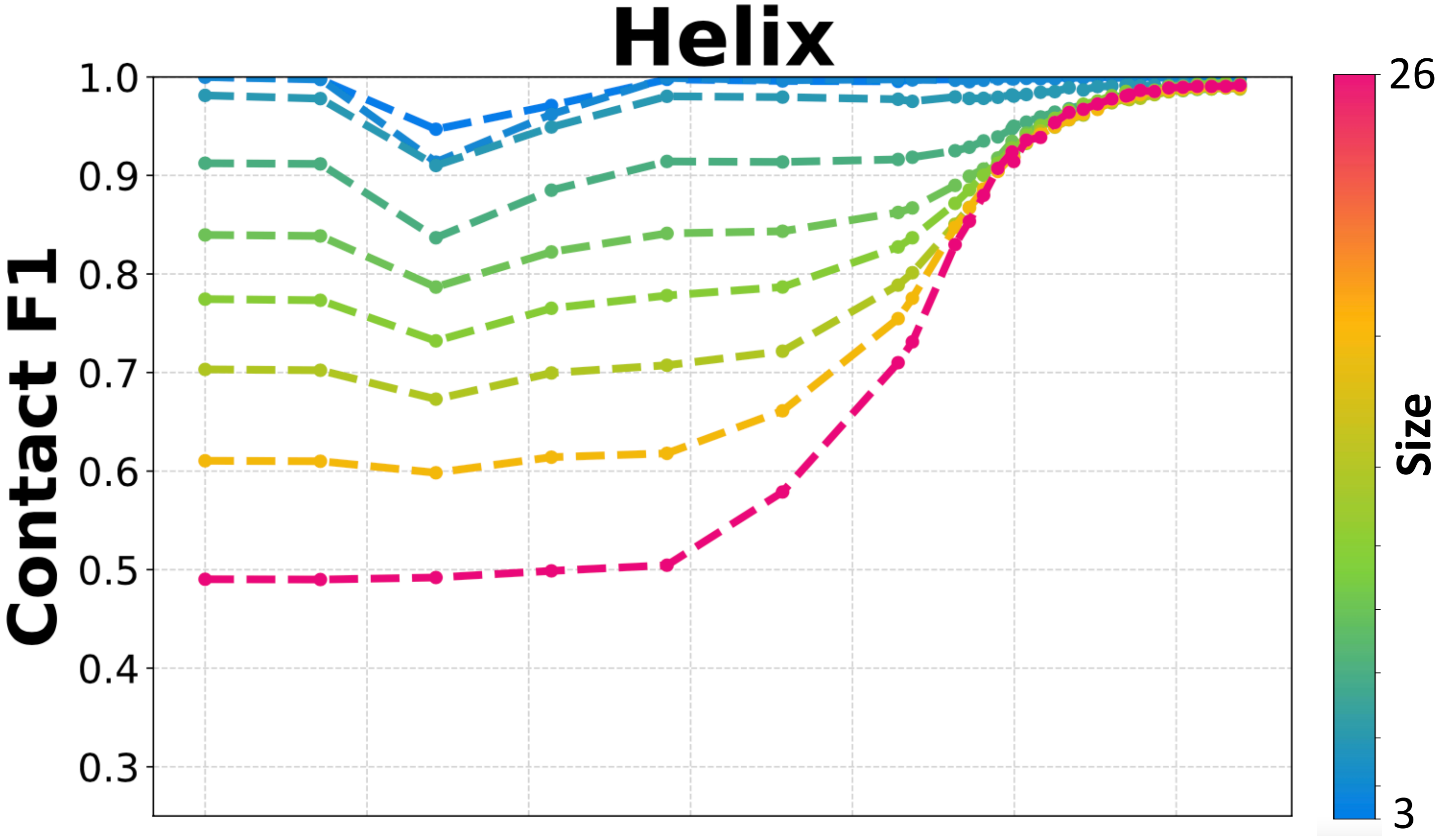


How does it learn? (multiple scales)

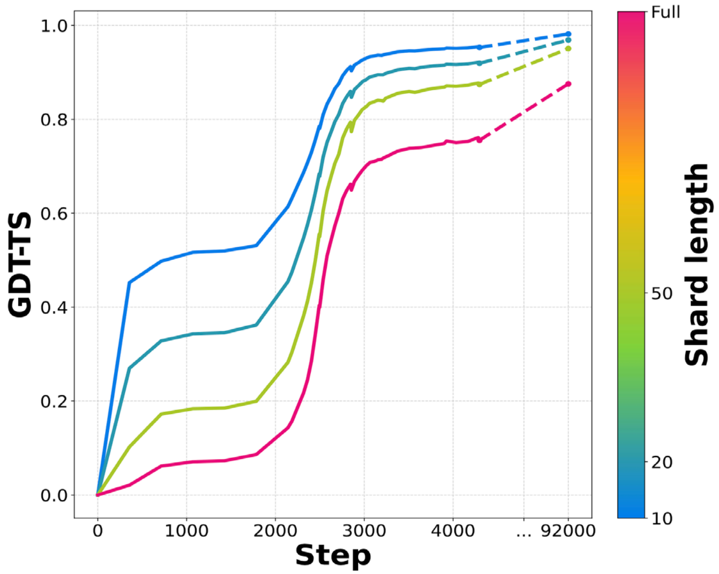


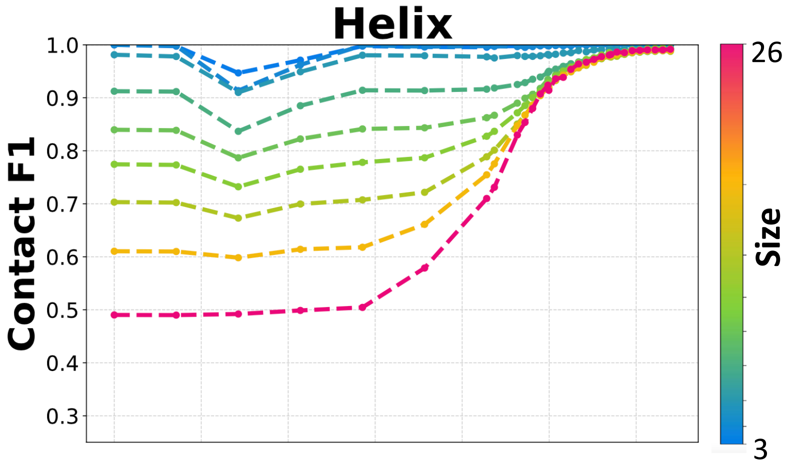
How does it learn? (multiple scales)

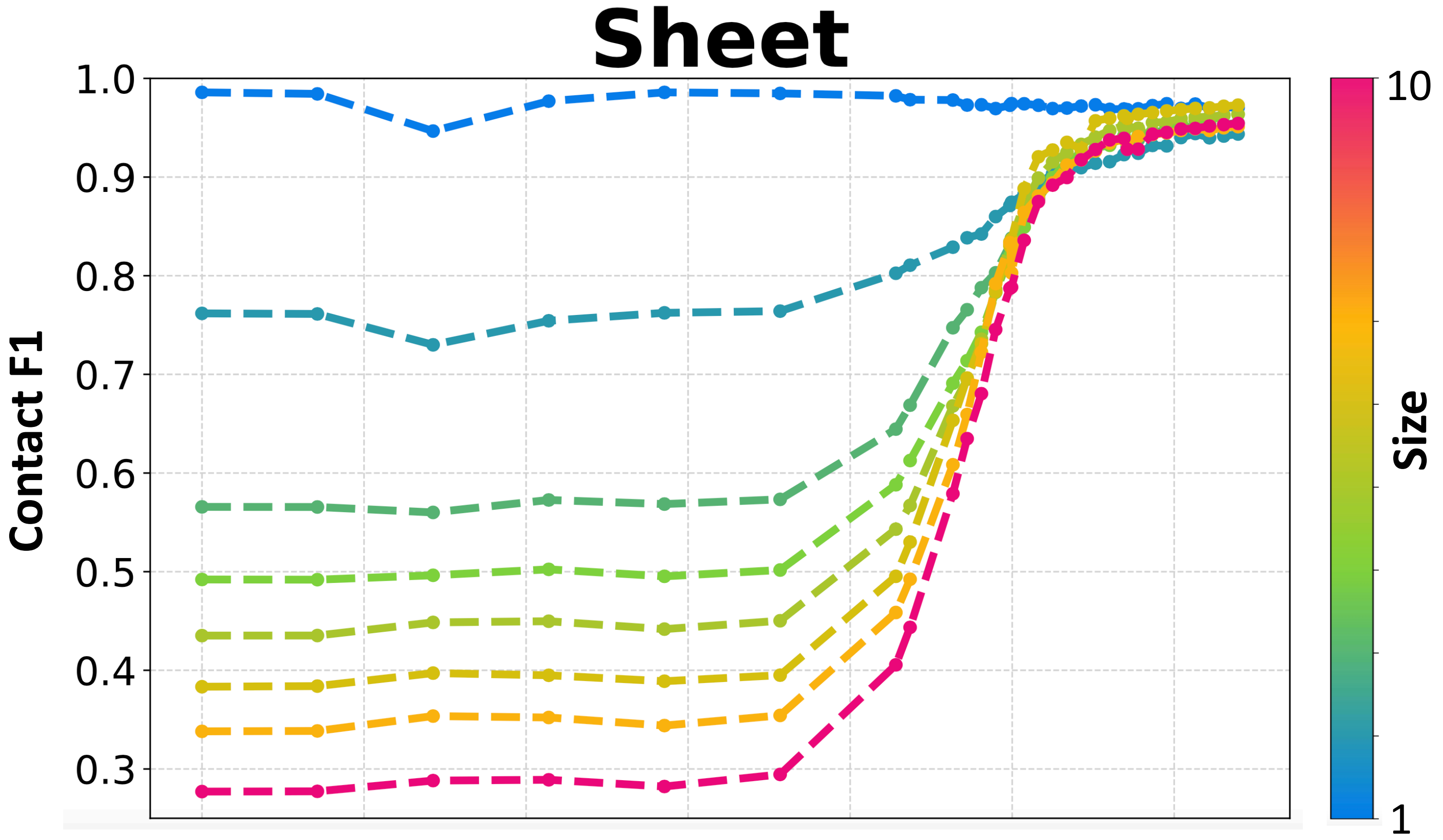




How does it learn? (multiple scales)



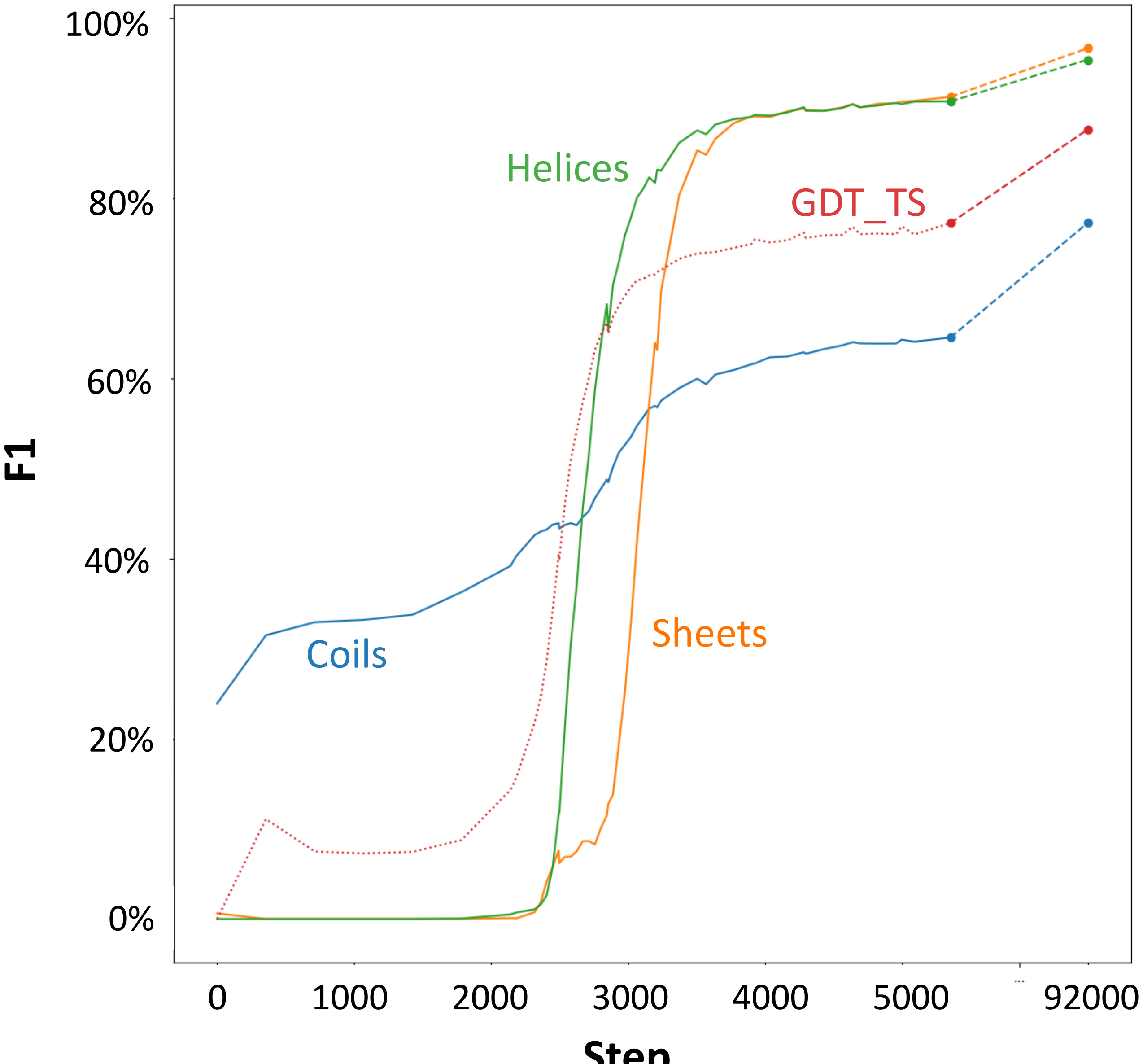




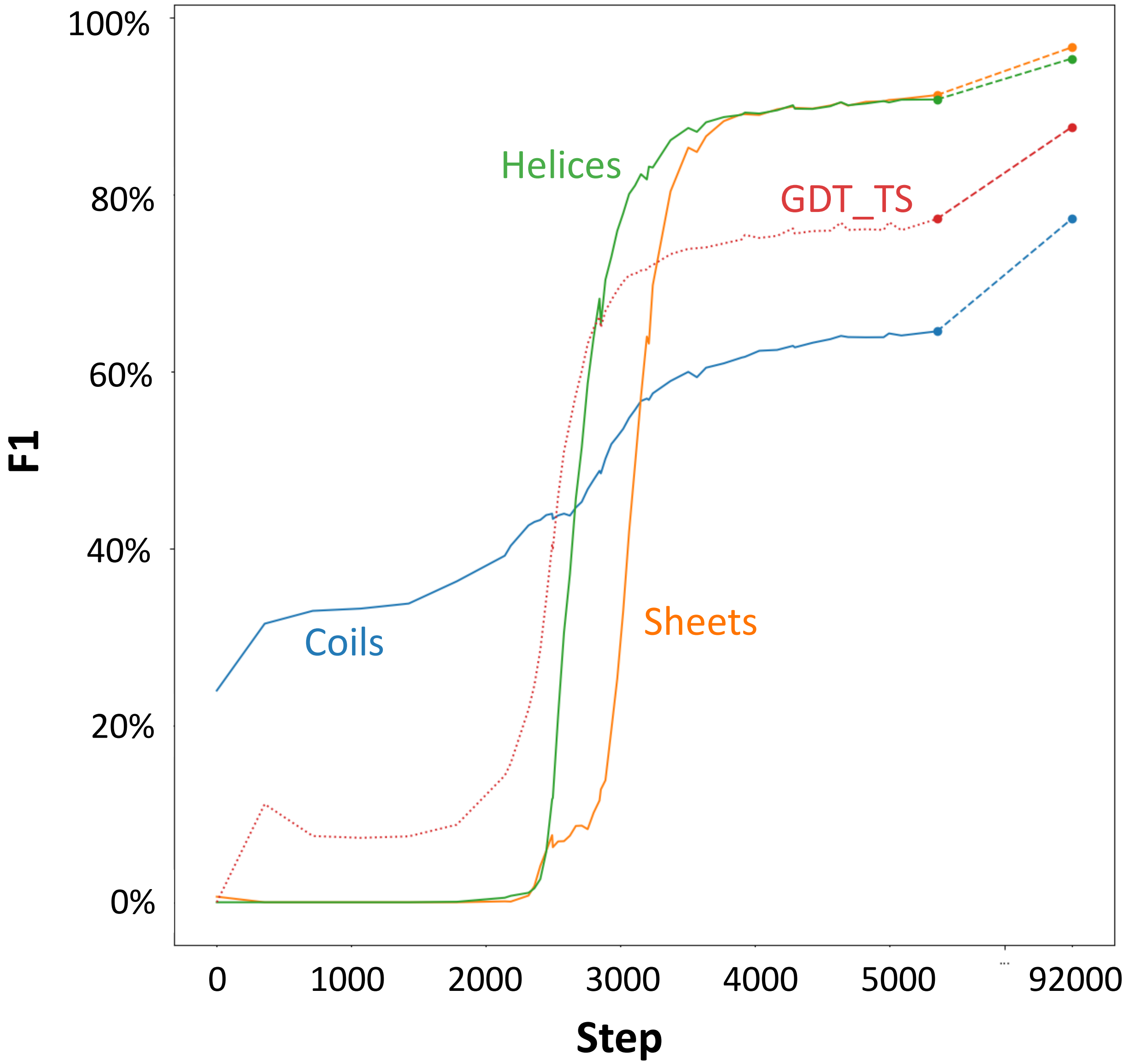
How does it learn? (multiple dimensions)

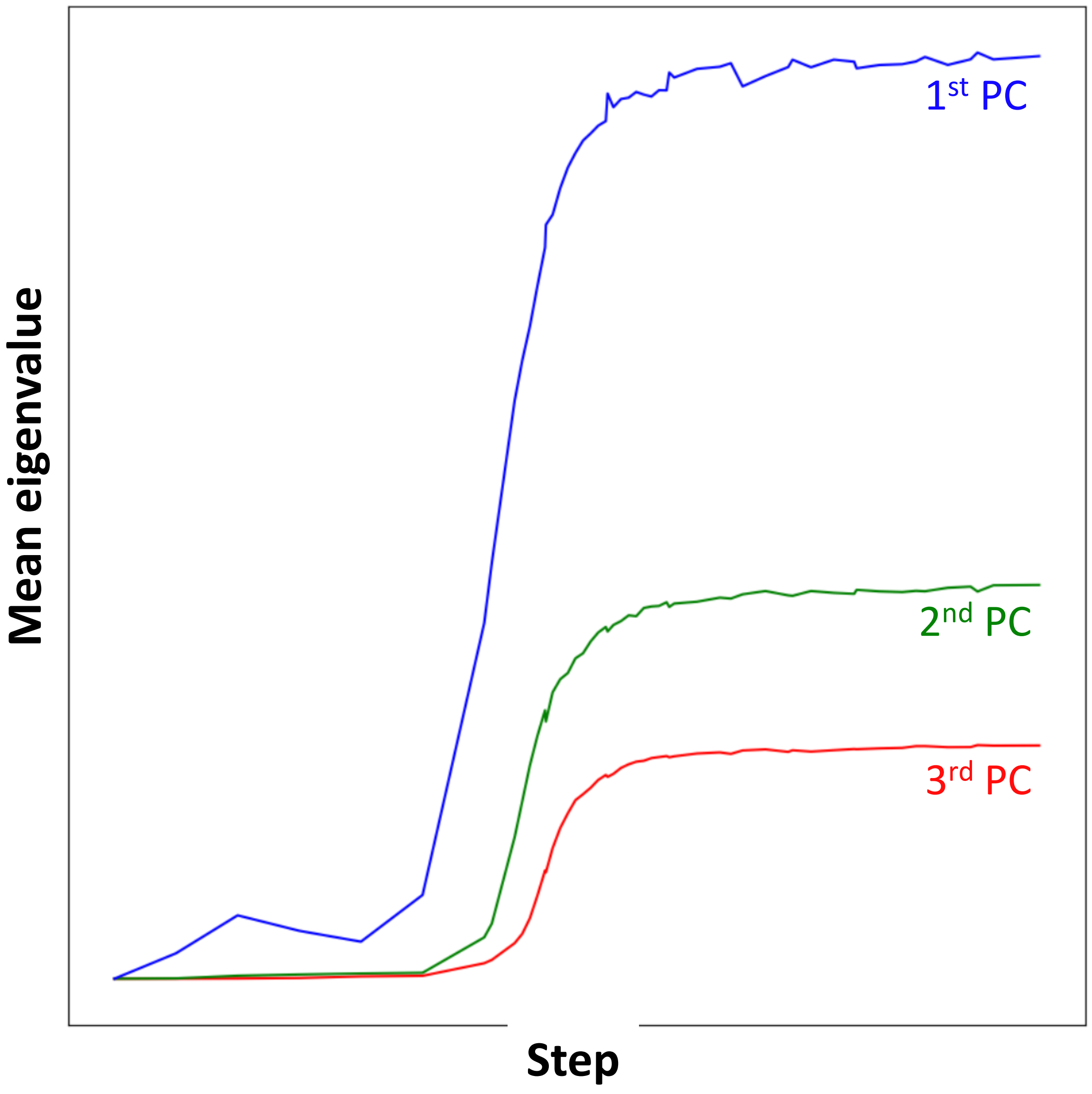
How does it learn? (multiple dimensions)

How does it learn? (multiple dimensions)

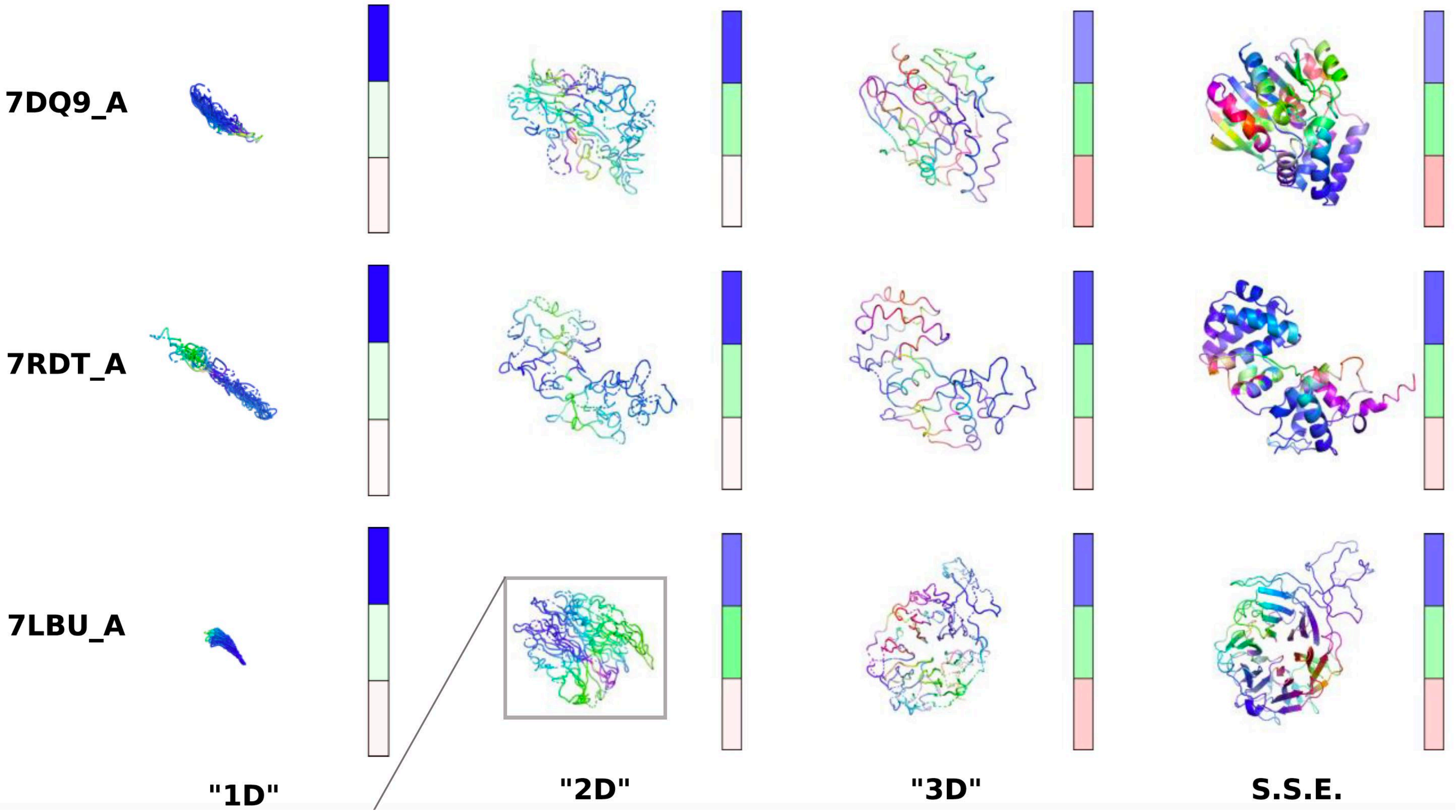


How does it learn? (multiple dimensions)

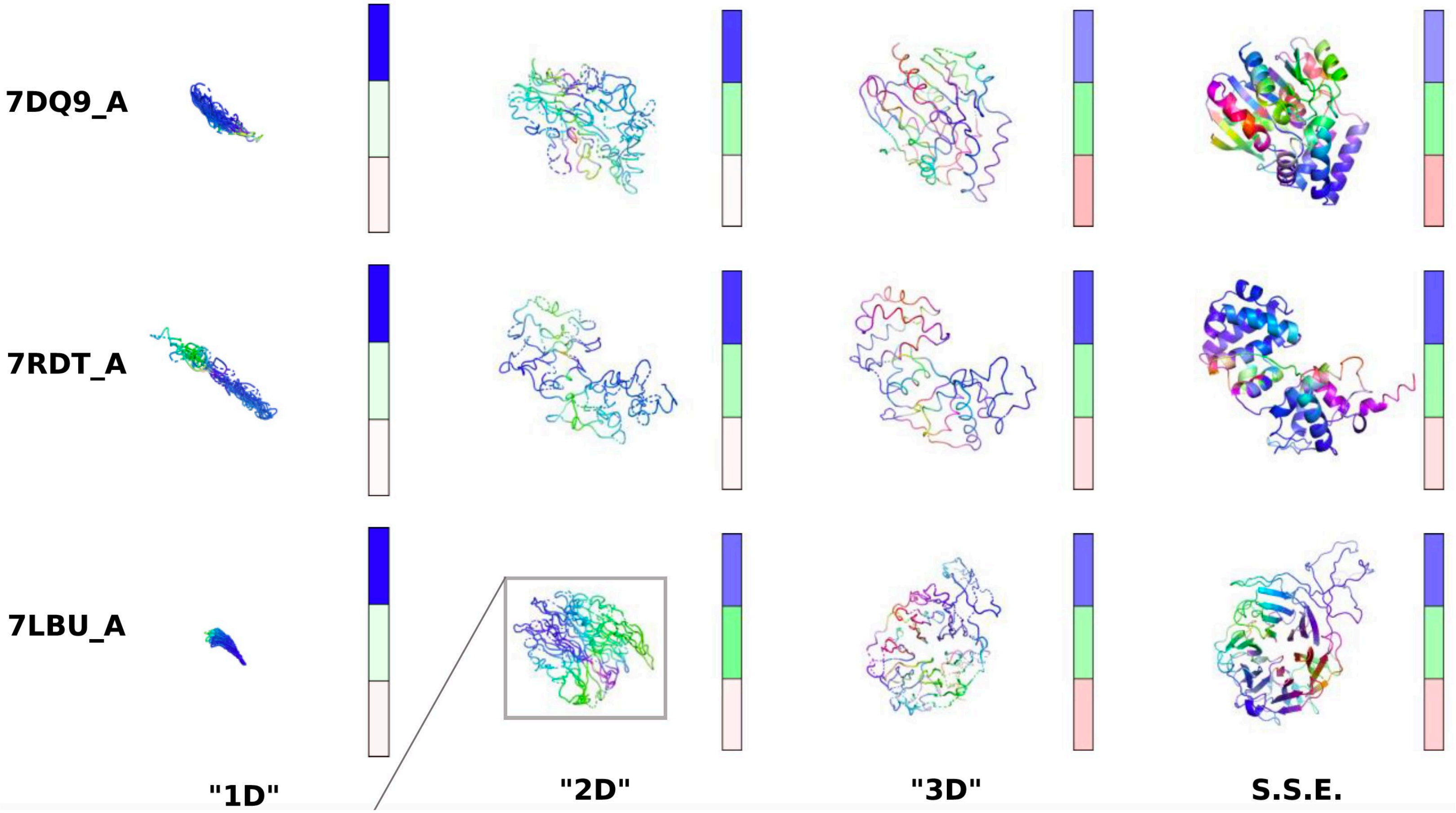




How does it learn? (multiple dimensions)



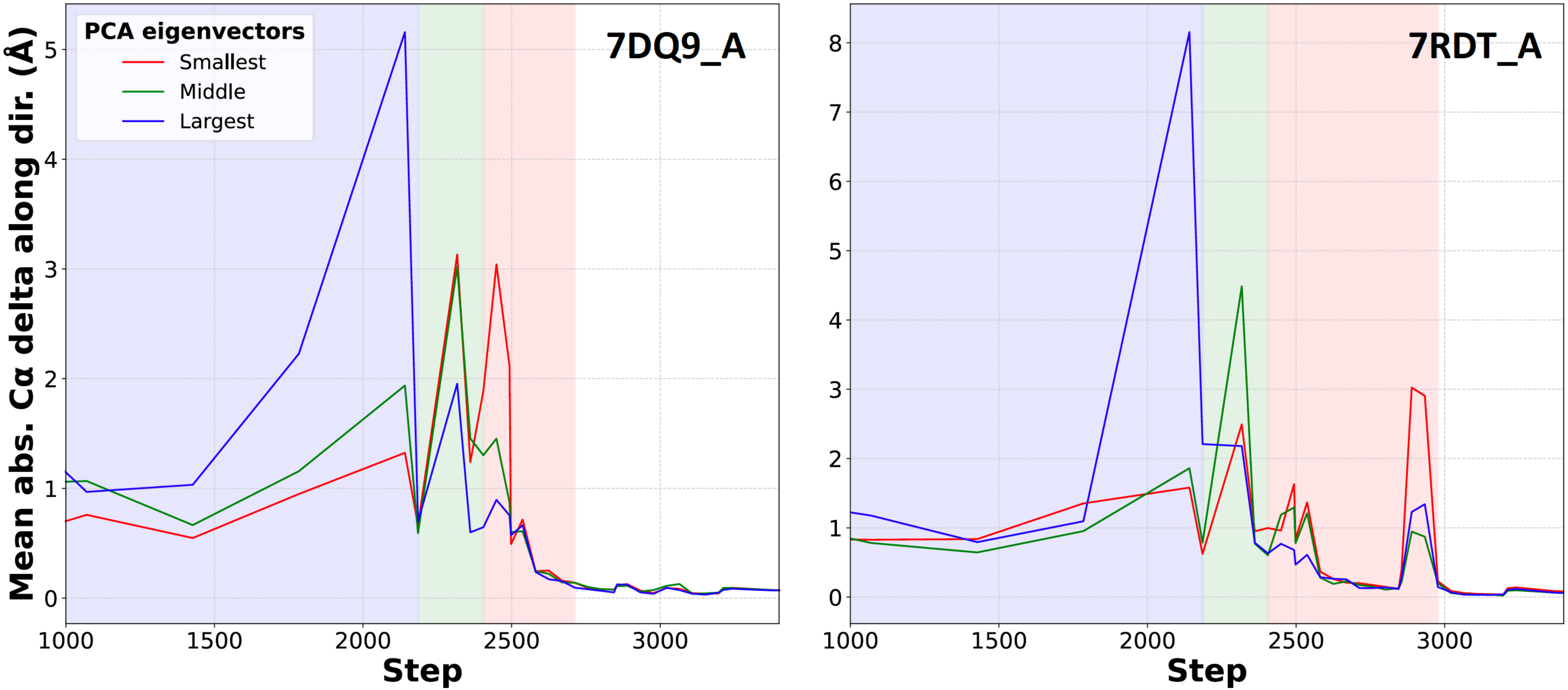
How does it learn? (multiple dimensions)



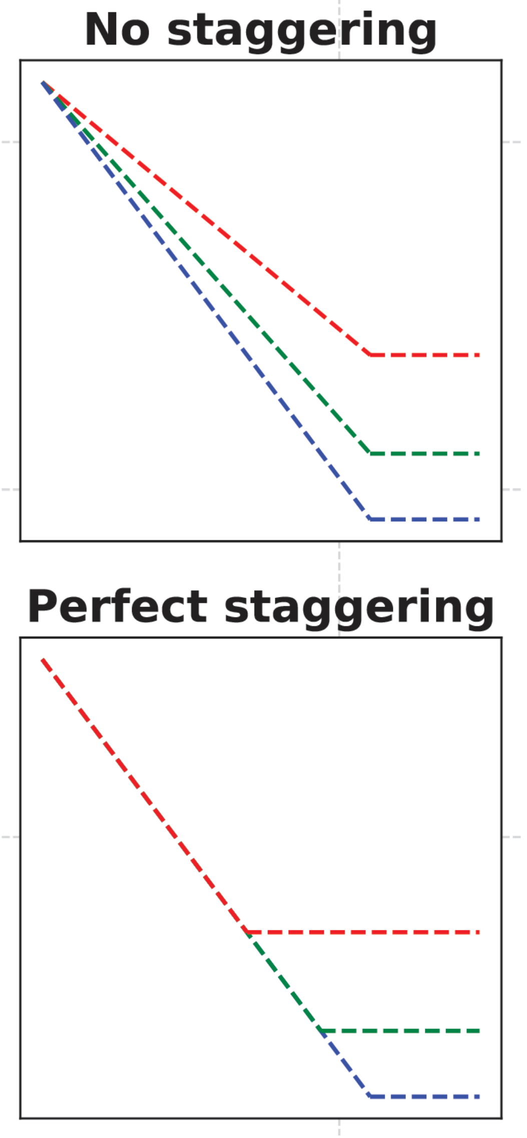
Is it *just* about dimensions?

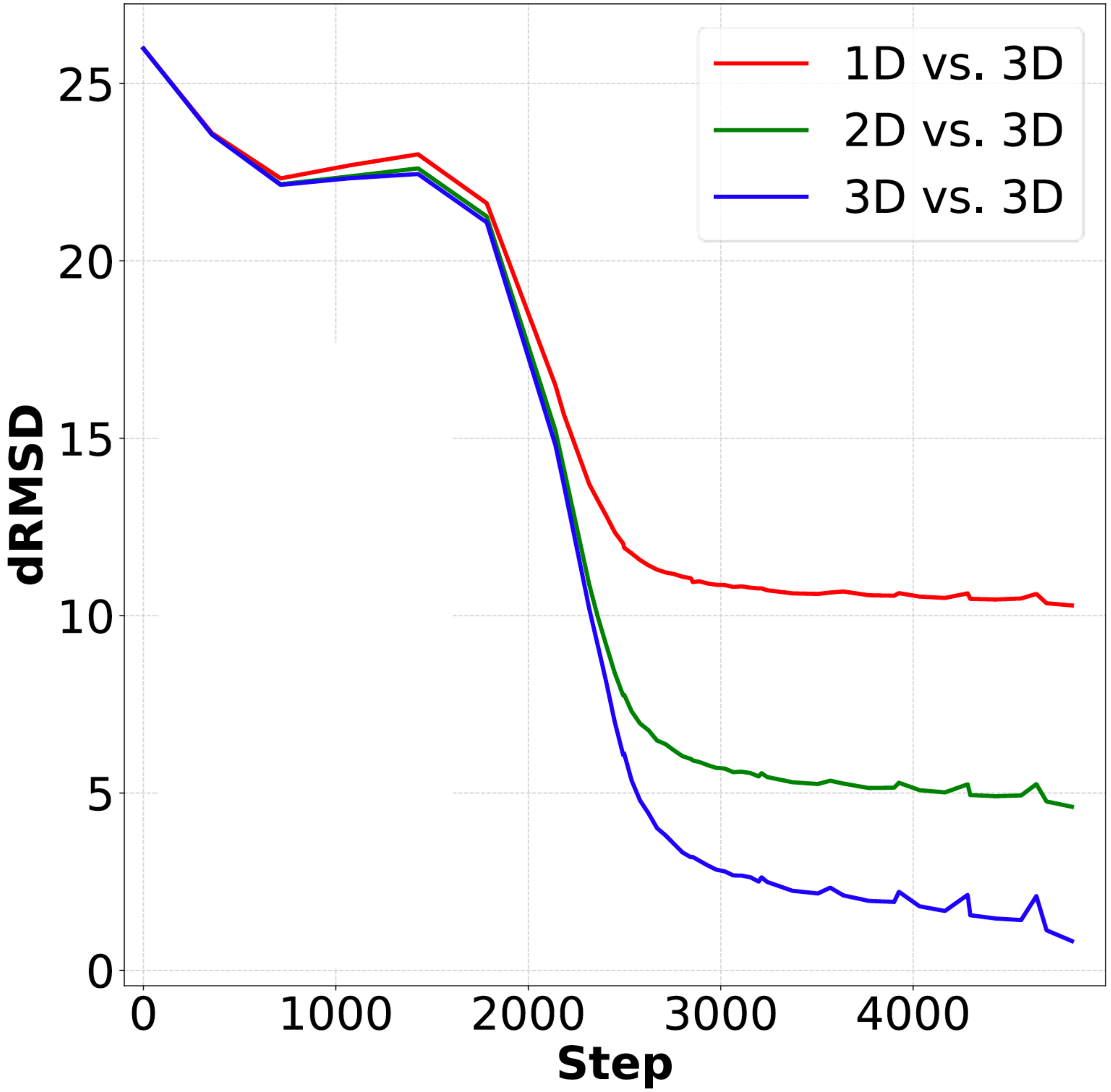
Can a stronger statement be made?

How does it learn? (staggered PCA projections!)

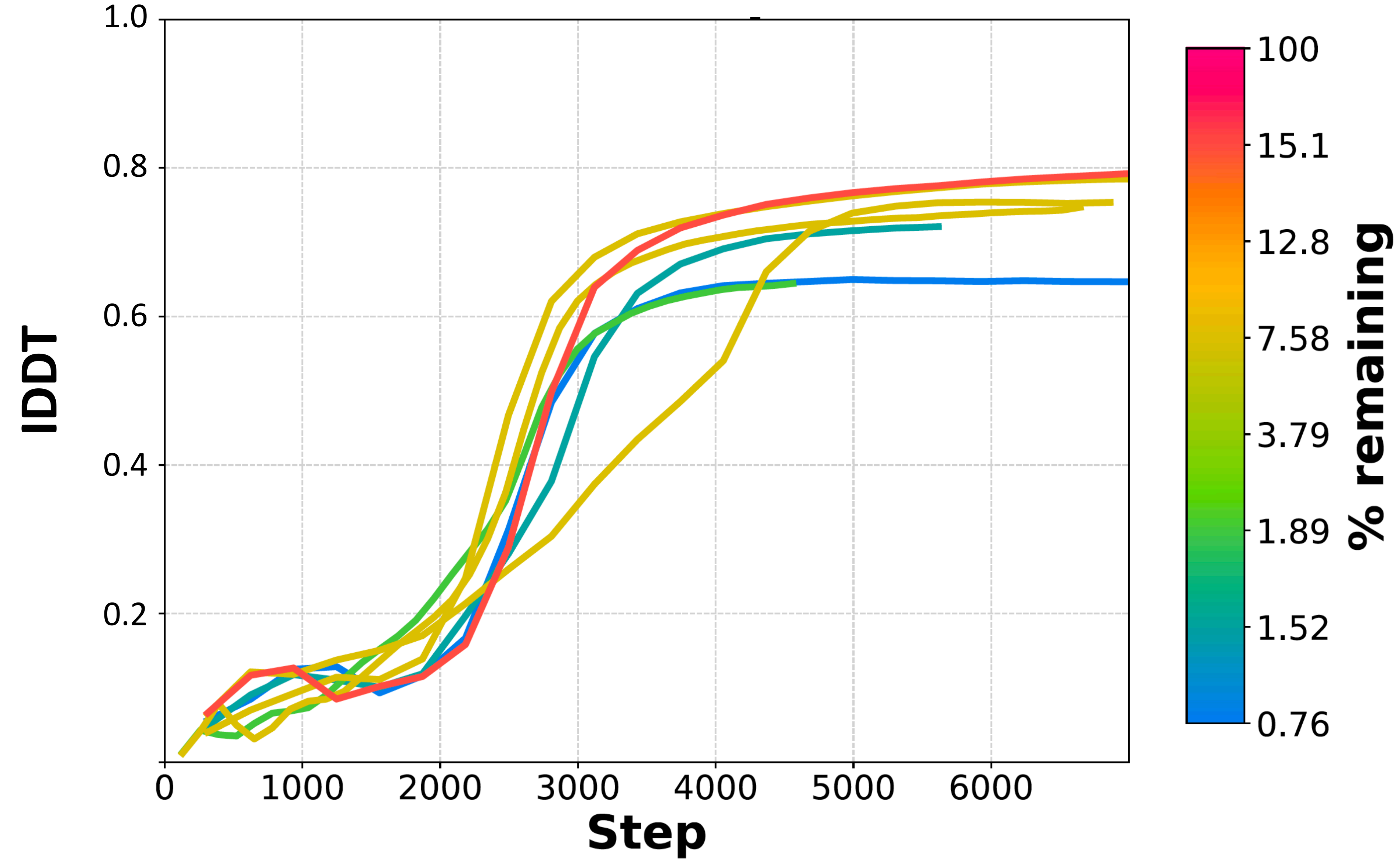


How does it learn? (staggered PCA projections!)

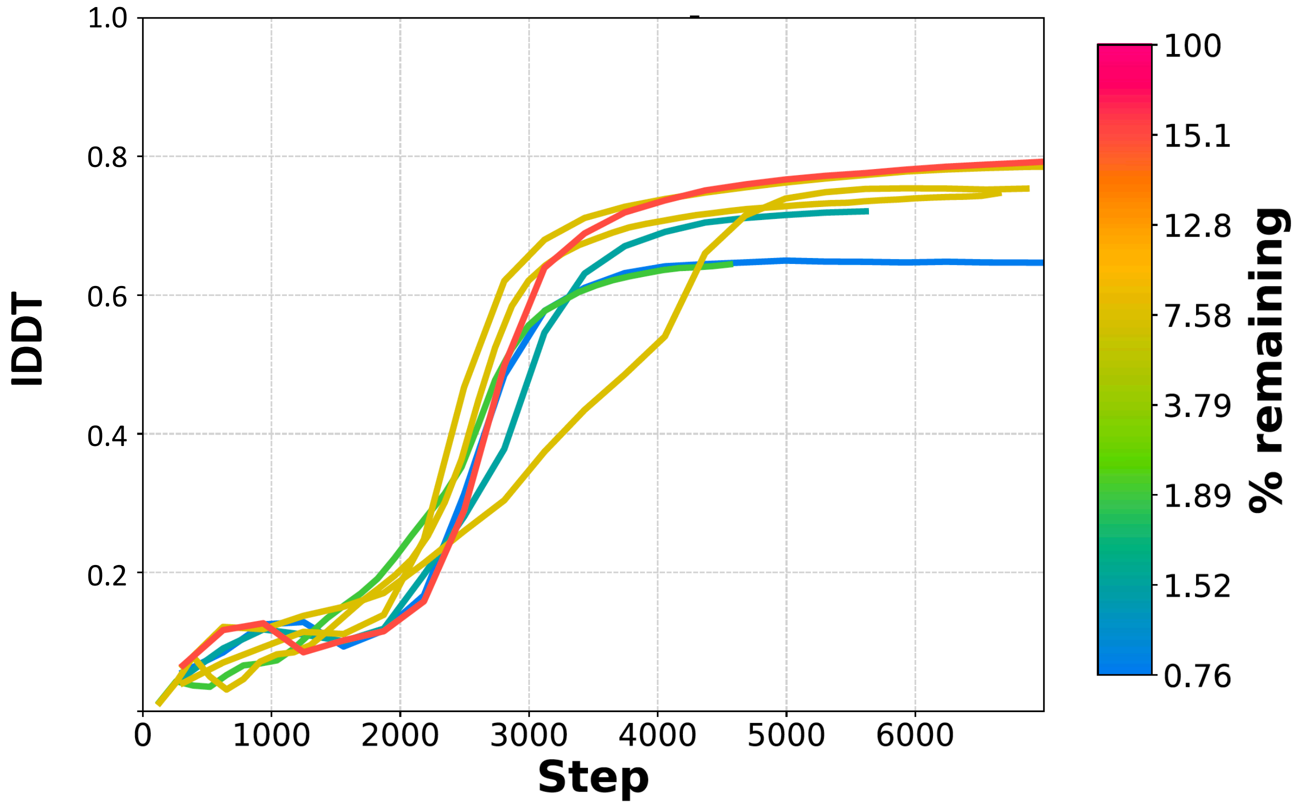


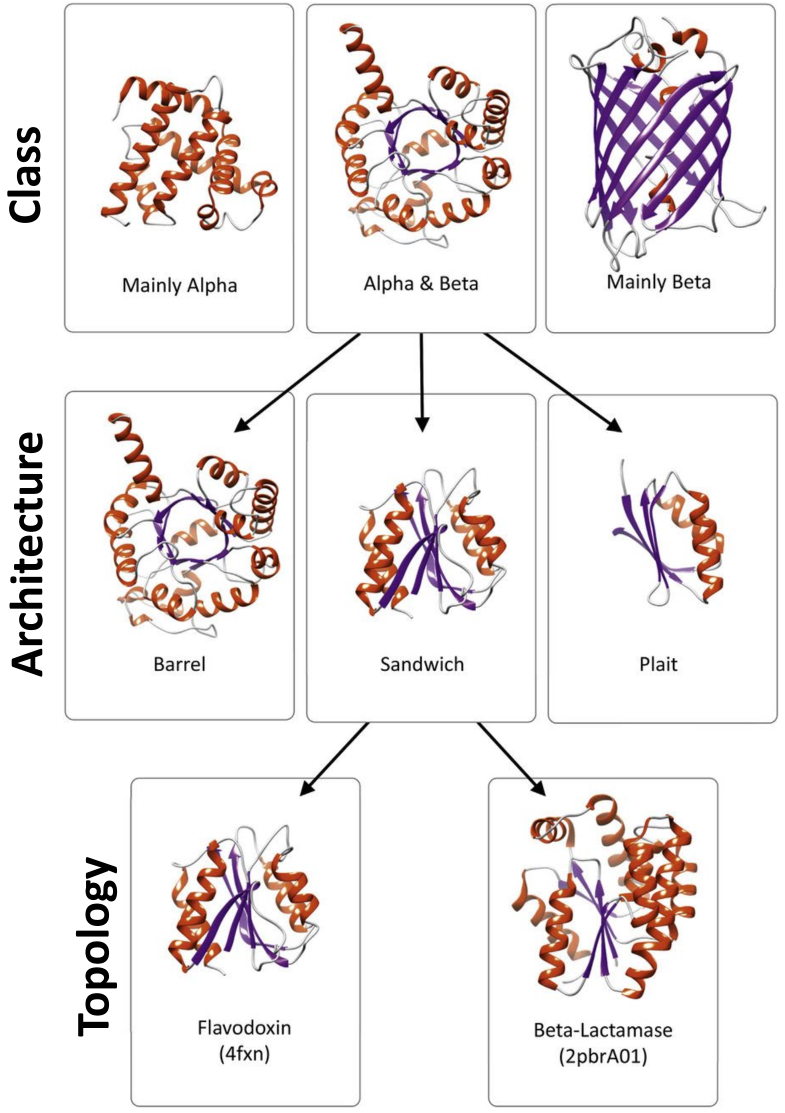


How well does it learn? (data reductions)



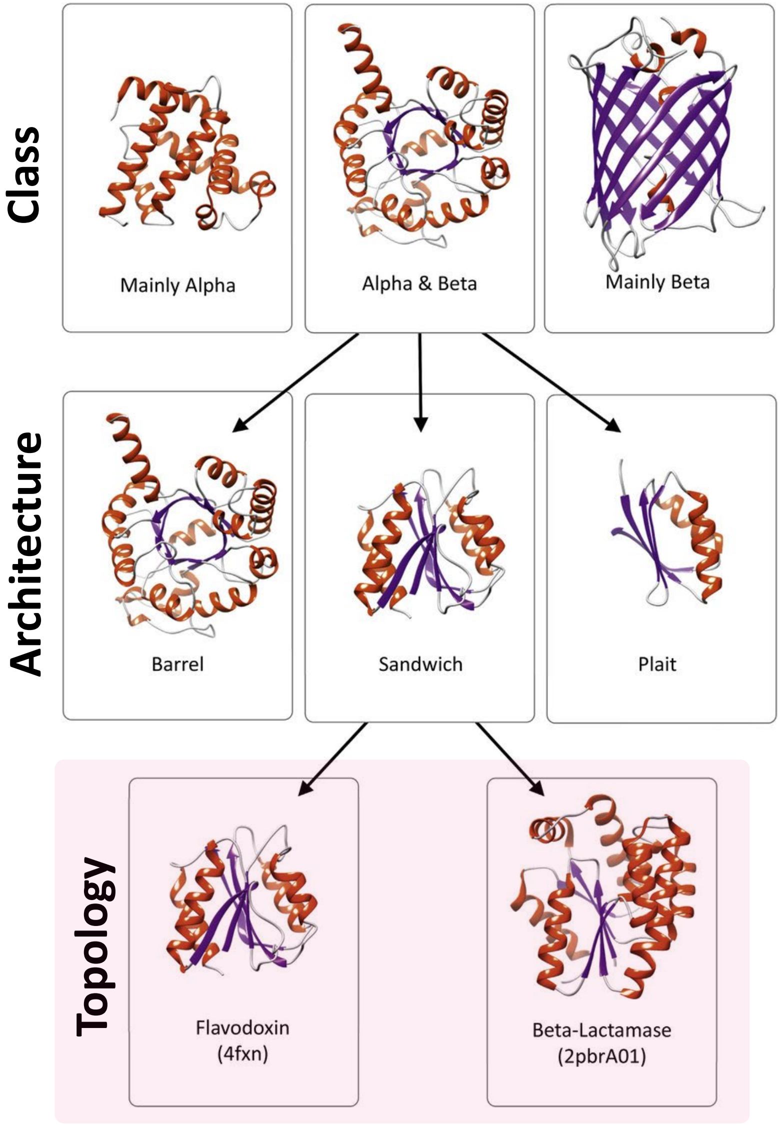
How well does it learn? (data reductions)



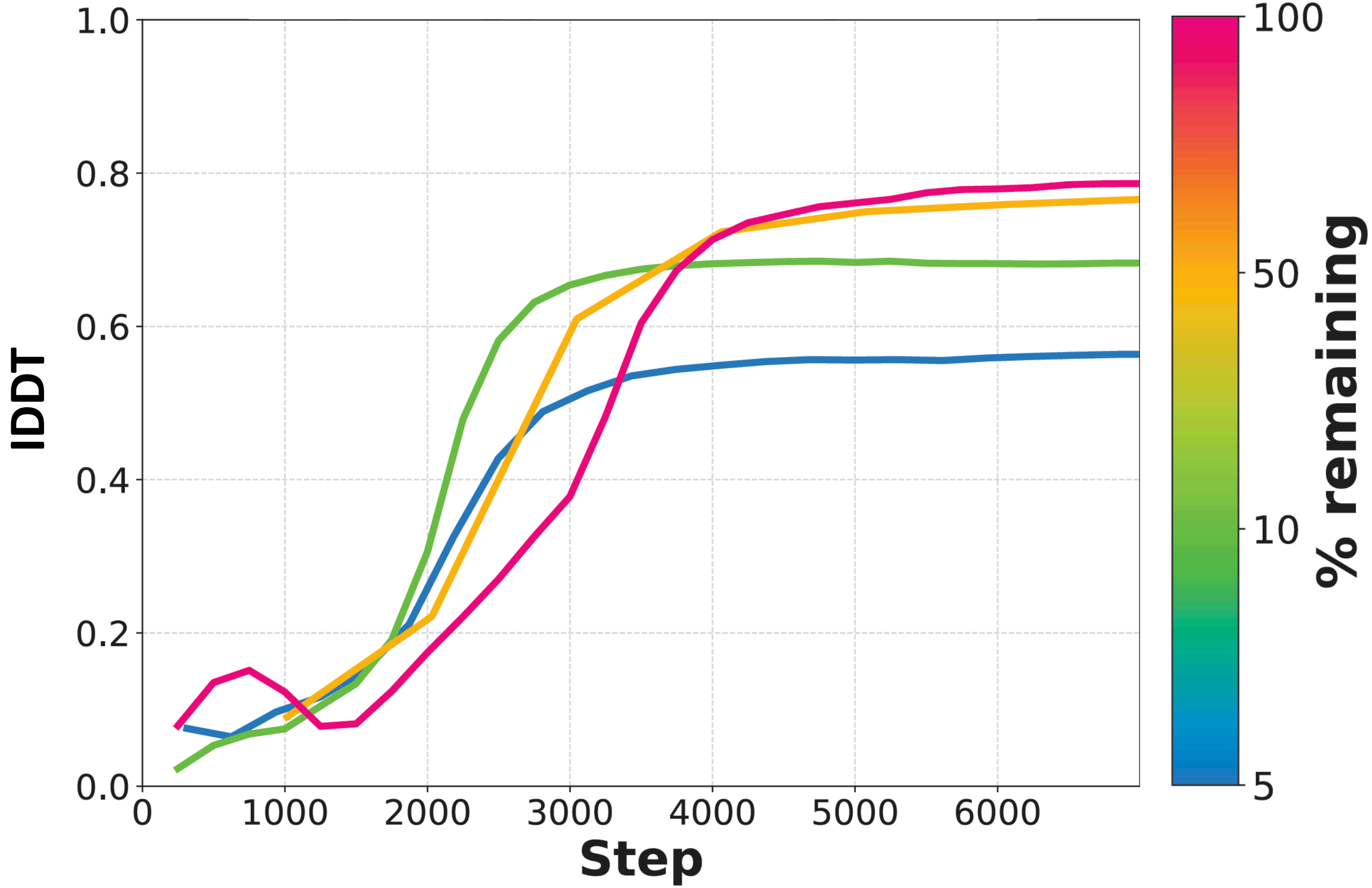


CATH classification

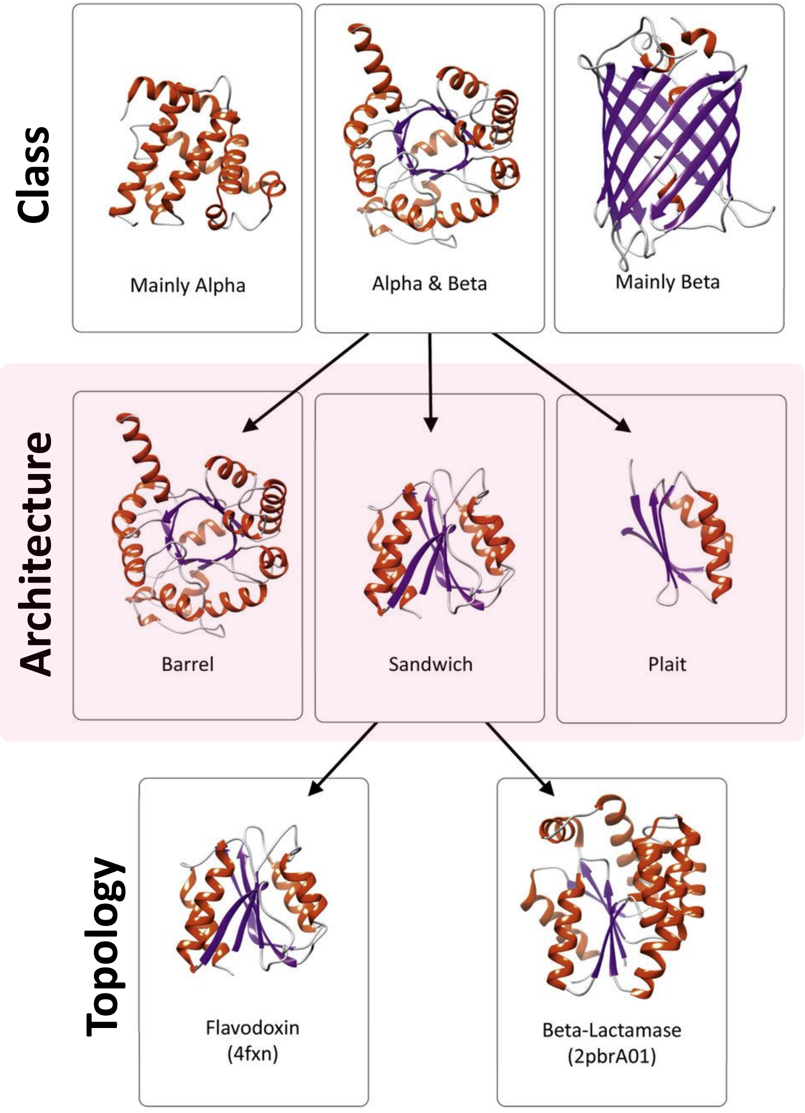
How well does it learn? (data reductions)



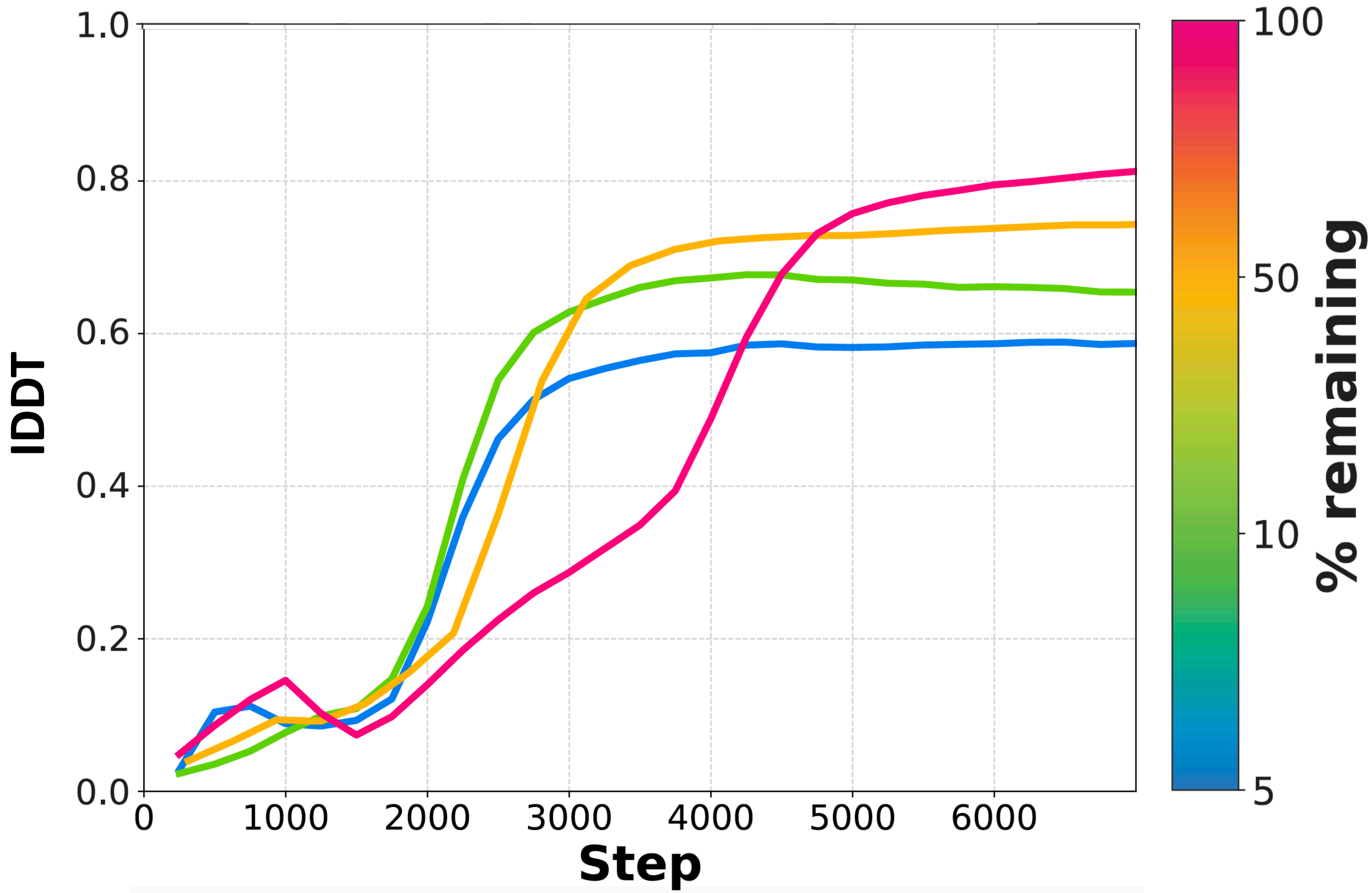
CATH classification



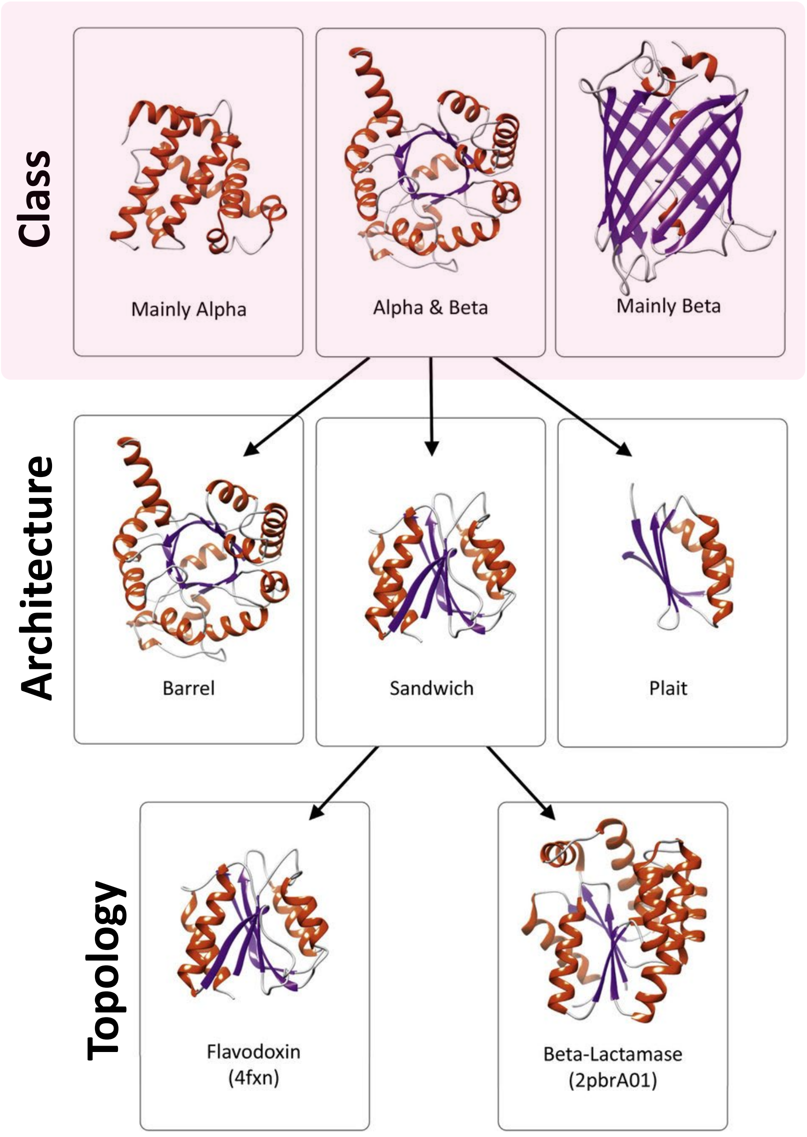
How well does it learn? (data reductions)



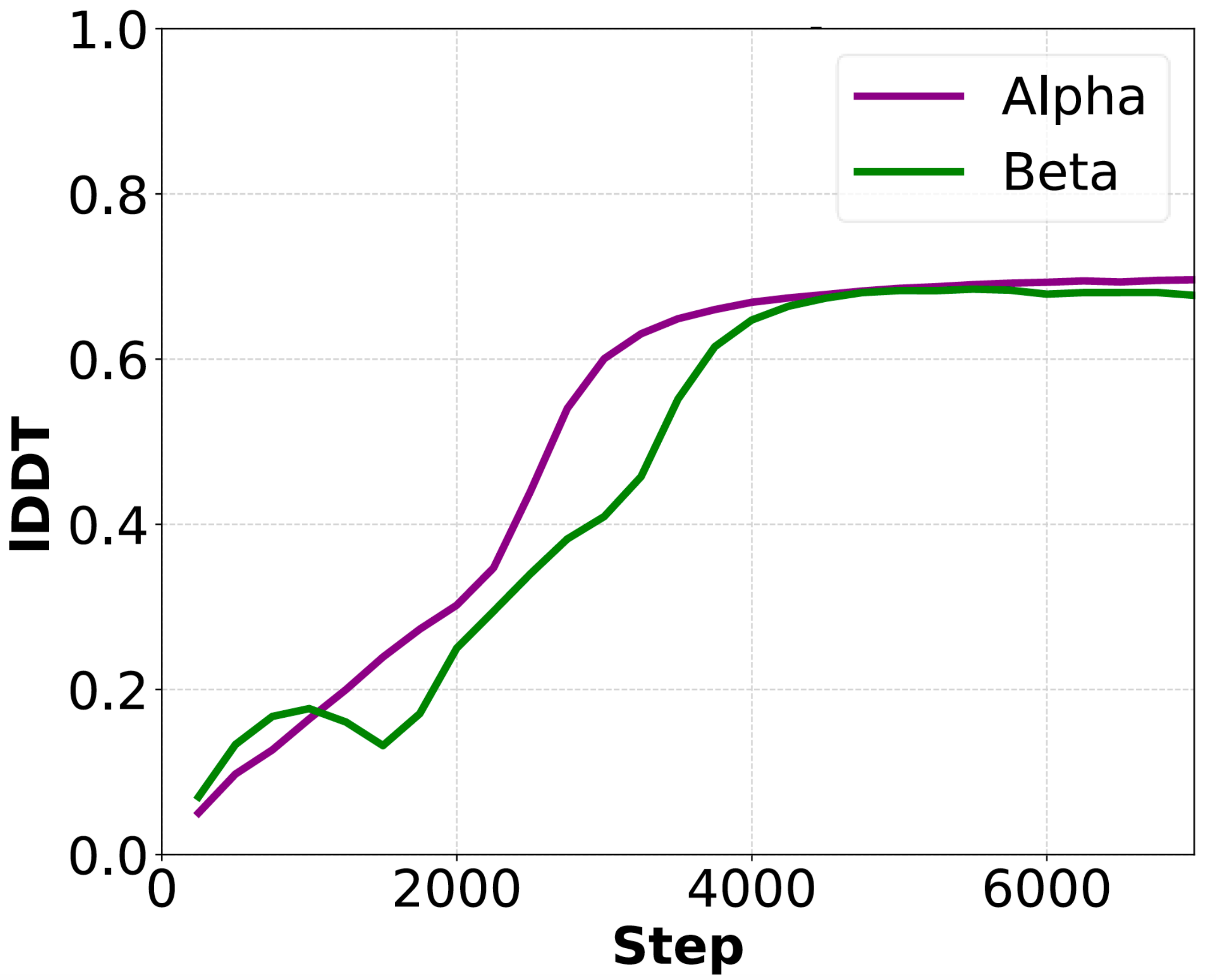
CATH classification



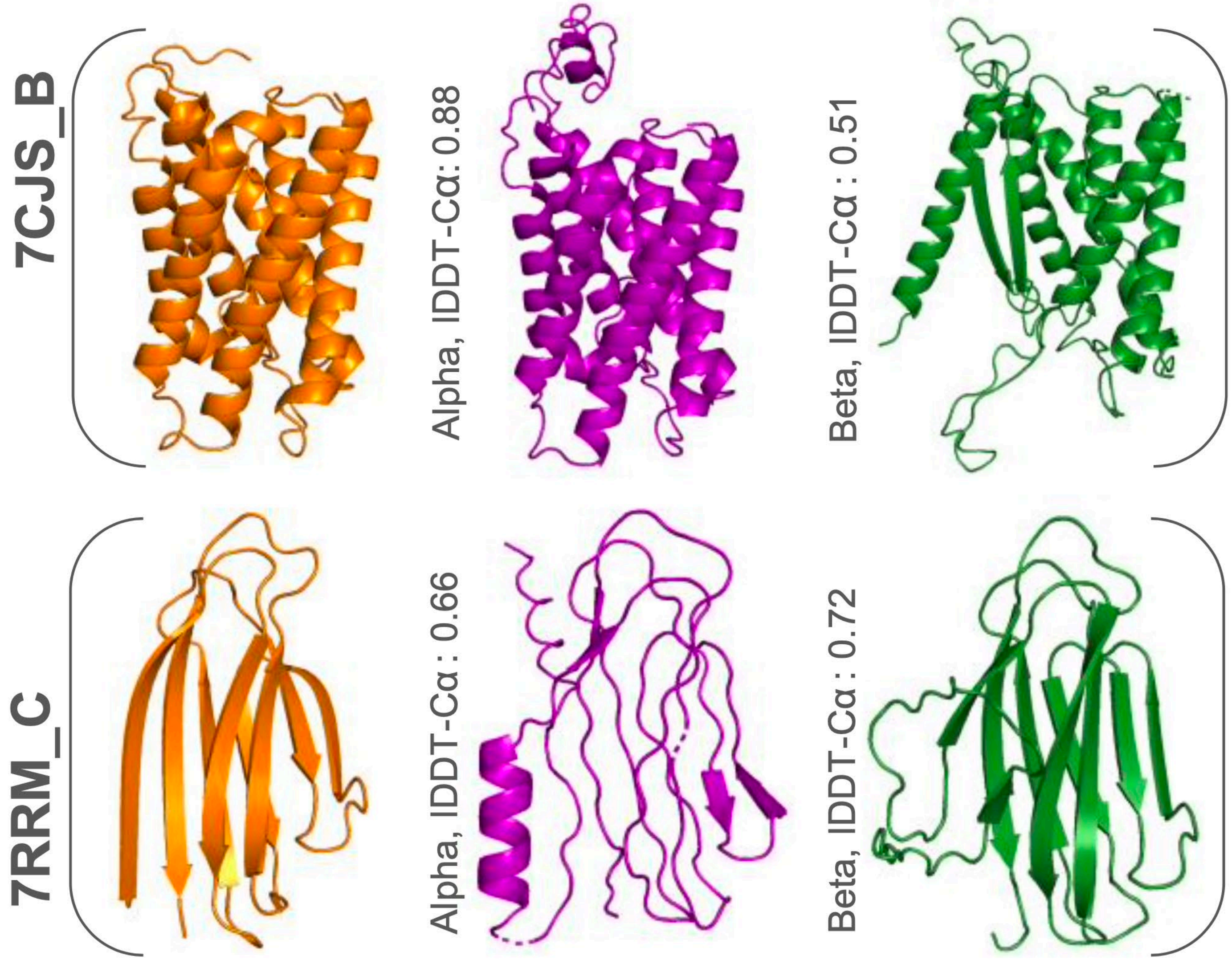
How well does it learn? (data reductions)



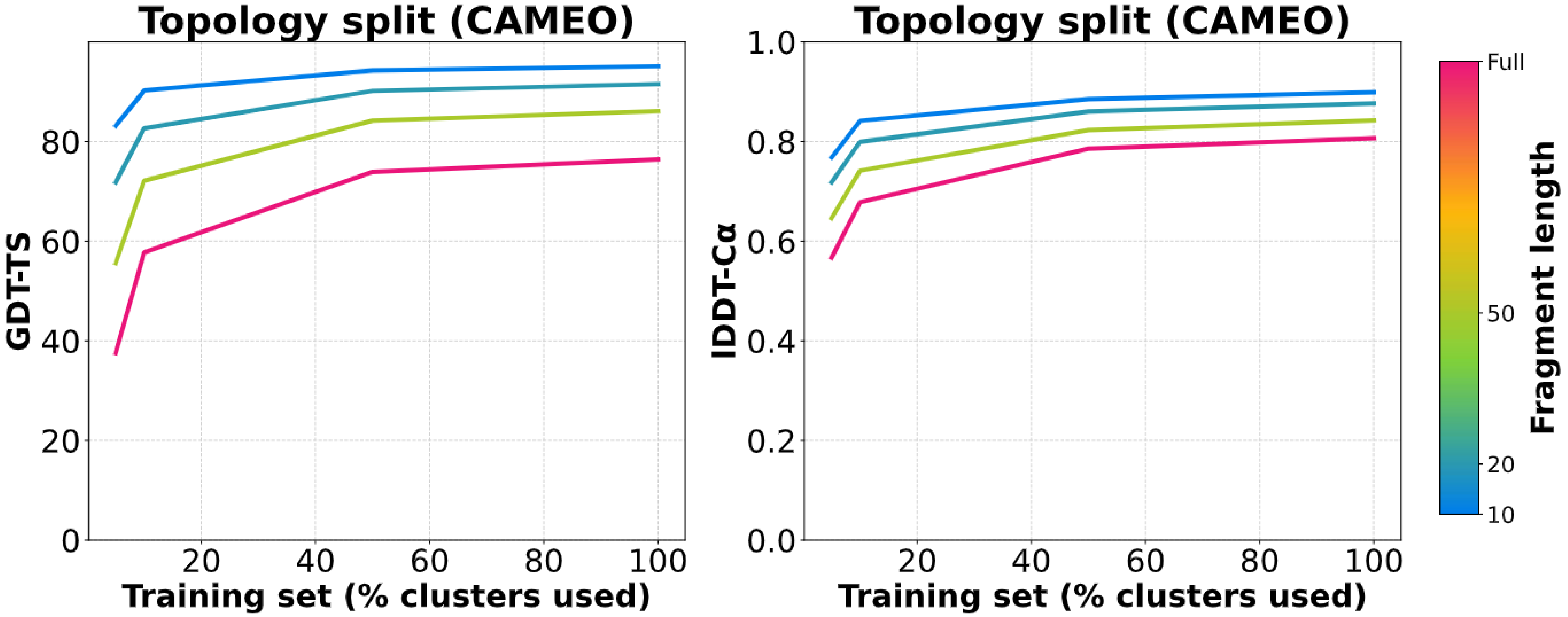
CATH classification



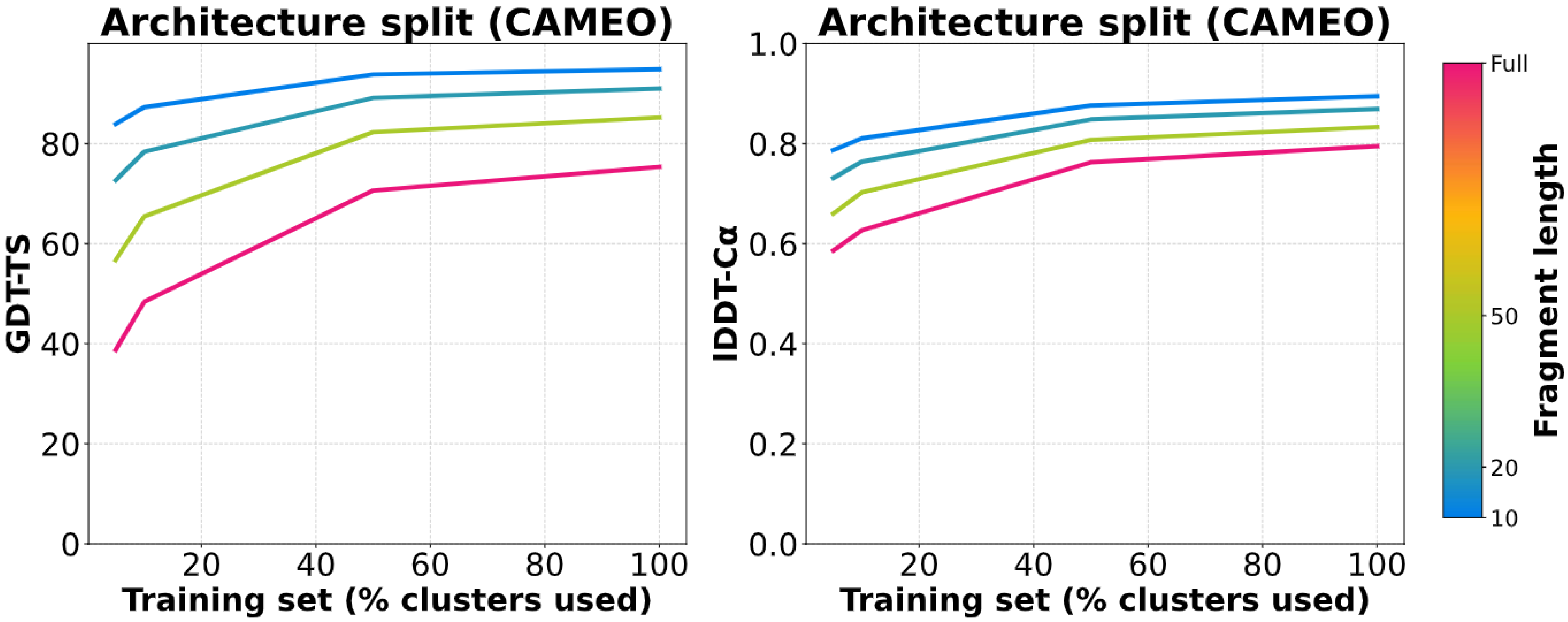
How well does it learn? (data reductions)



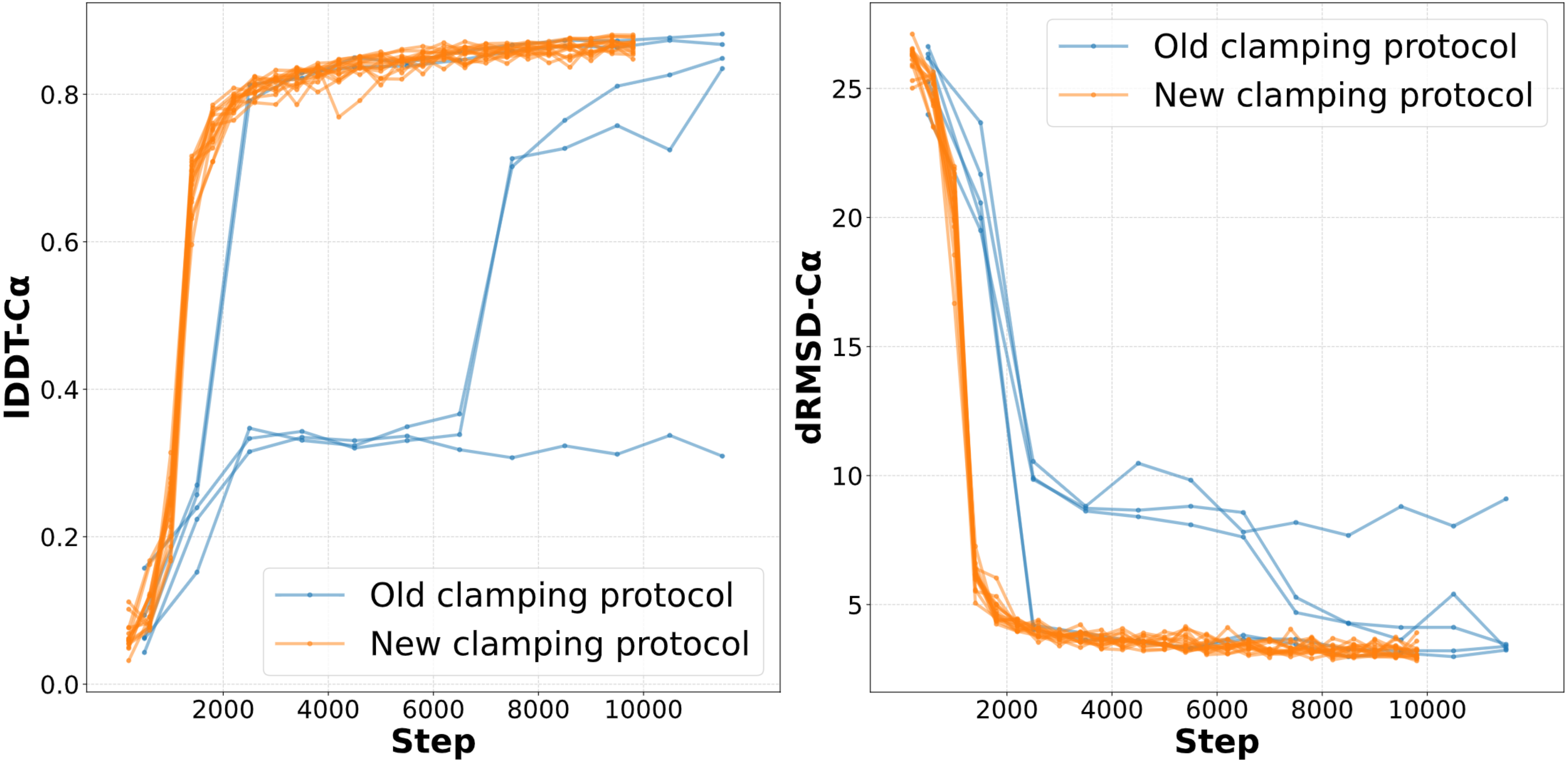
How well does it learn? (data reductions)



How well does it learn? (data reductions)



In the trenches of training (new features)



Where are we at?

Stage 1 (complete)

- Full implementation of AlphaFold v2.0.1, including training code
- Implementation of AlphaFold-Multimer inference code

Stage 2 (complete)

- Fresh retraining of model weights
- Demonstrate full reproduction capability

Stage 3 (academic-industry consortium)

- Open platform for machine-learned biomolecular modeling
  - Single sequence prediction (language models + AF2)
  - Structural priors / integration with experimental data
  - Multiple conformations and intrinsically disordered proteins
  - Protein-small molecule
  - Large multi-unit complexes
  - Protein design
  - Unnatural amino acids

OpenFold Software

Gustaf Ahdritz

Nazim Bouatta

Sachin Kadyan

Luna Xia

Will Gerecke

Dan Berenberg (NYU)

PyTorch Team

OpenFold Executive Committee

Lucas Nivon (Cyrus)

Brian Weitzner (Outpace Bio)

Yih-En Andrew Ban (Arzeda)

Andrew Watkins (Genentech)

OpenFold Organizing

Raul Rabadan (Columbia)

David Mobley (OMSF)

Karmen Condic-Jurkic (OMSF)

COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

PROGRAM FOR
MATHEMATICAL
GENOMICS