Genentech
A Member of the Roche Group

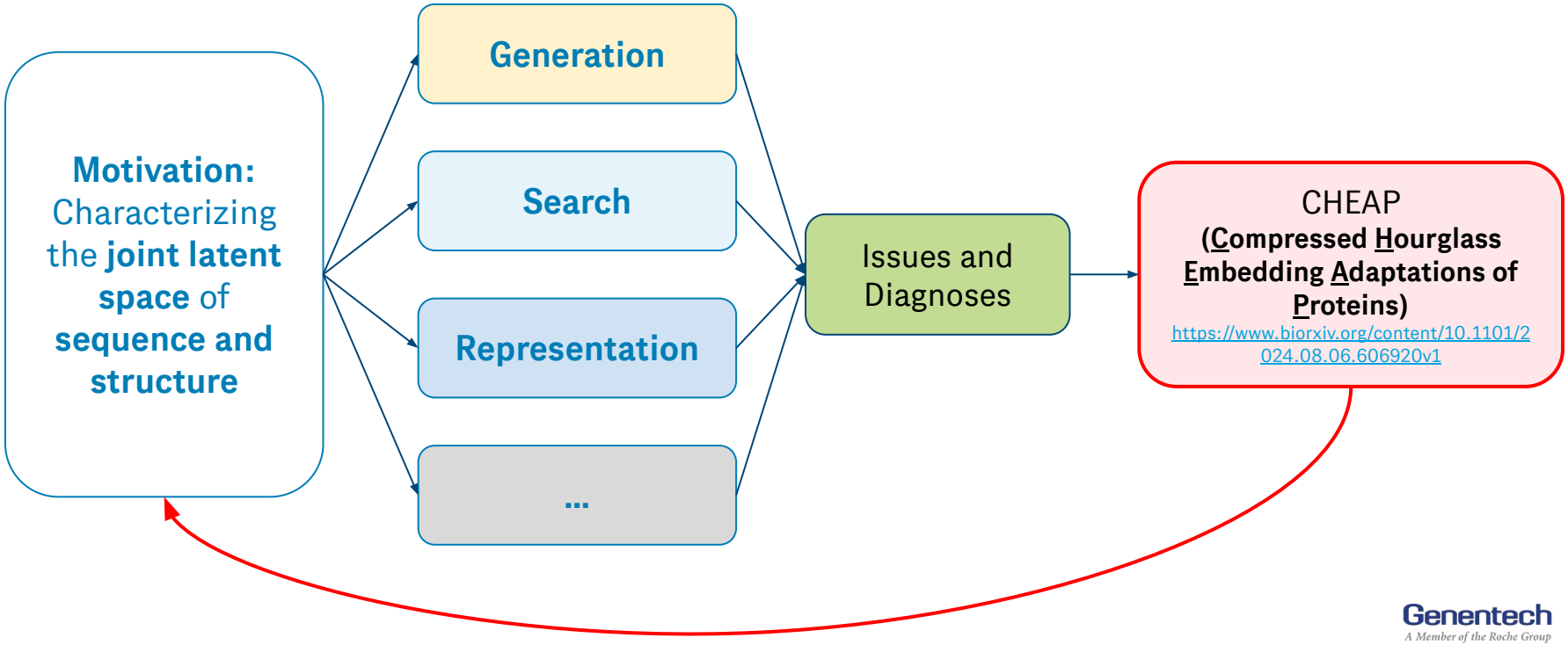
Tokenized and Continuous Embedding Compressions of Protein Sequence and Structure

October 22, 2024
Stanford AI + Biomedicine Seminar

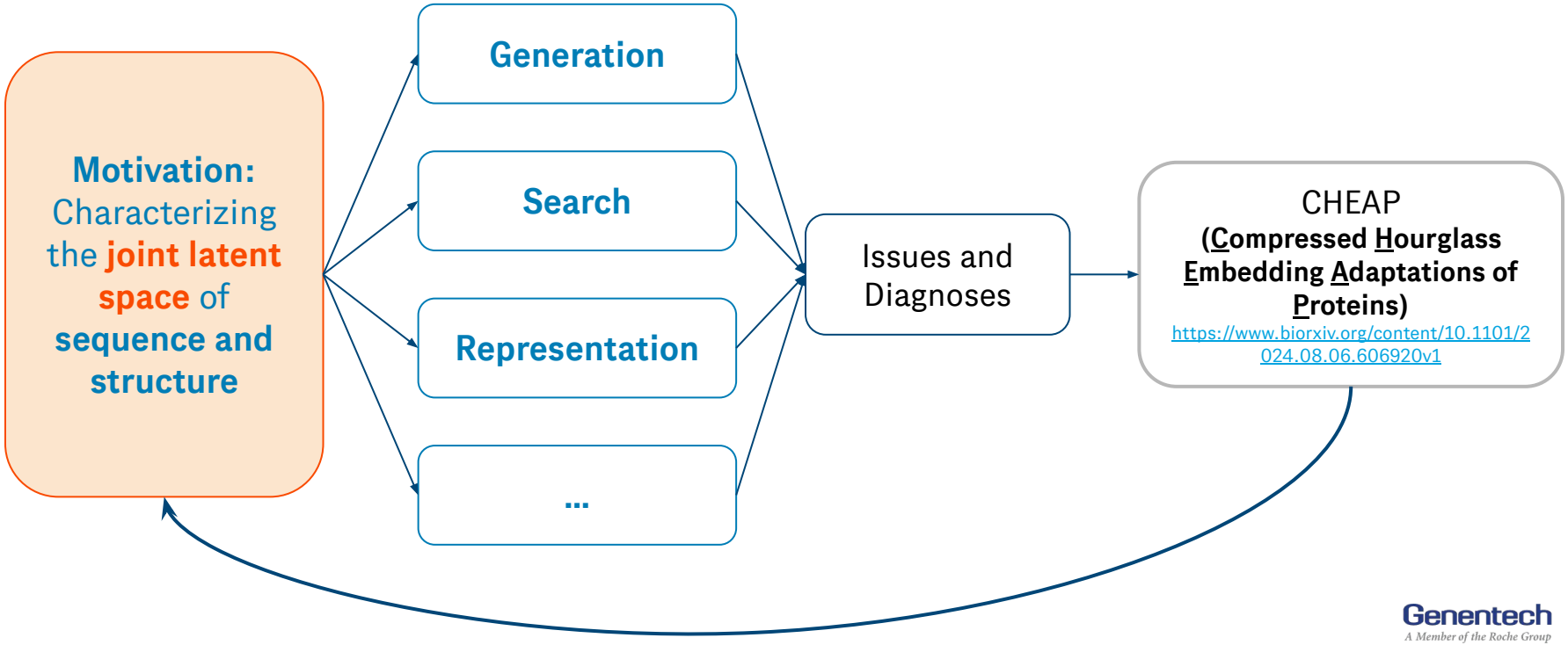
Amy X. Lu
UC Berkeley / BAIR
Prescient Design / Genentech

Paper: bit.ly/cheap-proteins
GitHub: github.com/amyxlu/cheap-proteins

Agenda

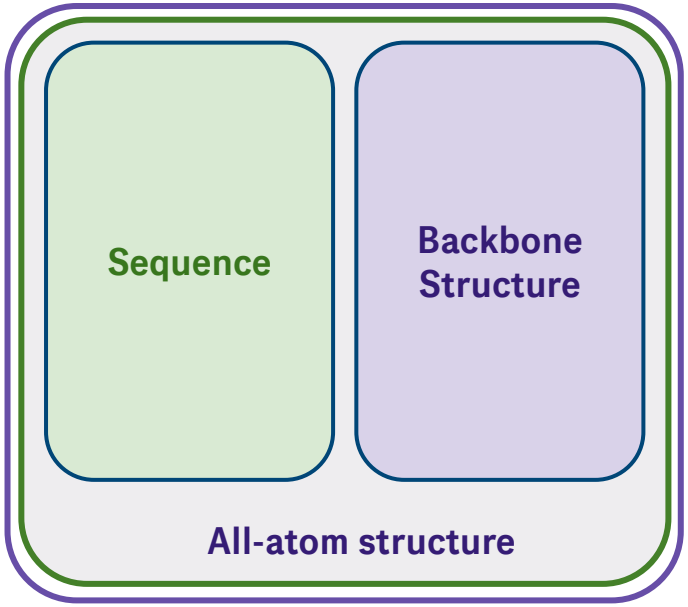

Motivation: Characterizing the joint latent space of sequence and structure
Generation
Search
Representation
...
Issues and Diagnoses
CHEAP
(Compressed Hourglass Embedding Adaptations of Proteins)
https://www.biorxiv.org/content/10.1101/2024.08.06.606920v1

Agenda


Motivation: Characterizing the joint latent space of sequence and structure
Generation
Search
Representation
...
Issues and Diagnoses
CHEAP
(Compressed Hourglass Embedding Adaptations of Proteins)
https://www.biorxiv.org/content/10.1101/2024.08.06.606920v1

Motivation: **Obtaining a joint embedding of structure & sequence from sequence alone**

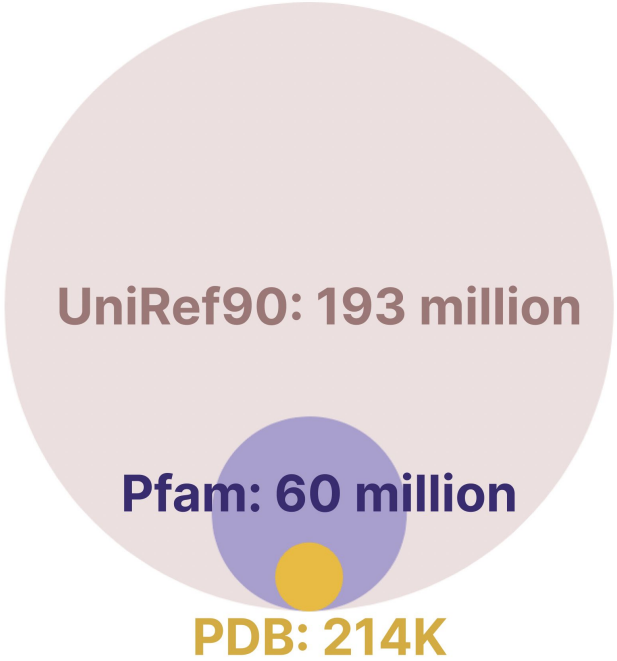
- Existing protein representation models often capture either p(sequence) or p(structure), limiting flexibility
- Desiderata:
  - Capture the joint embedding of sequence and structure
  - Can be explicitly decoded back to structure and sequence
  - Can be captured from sequence alone



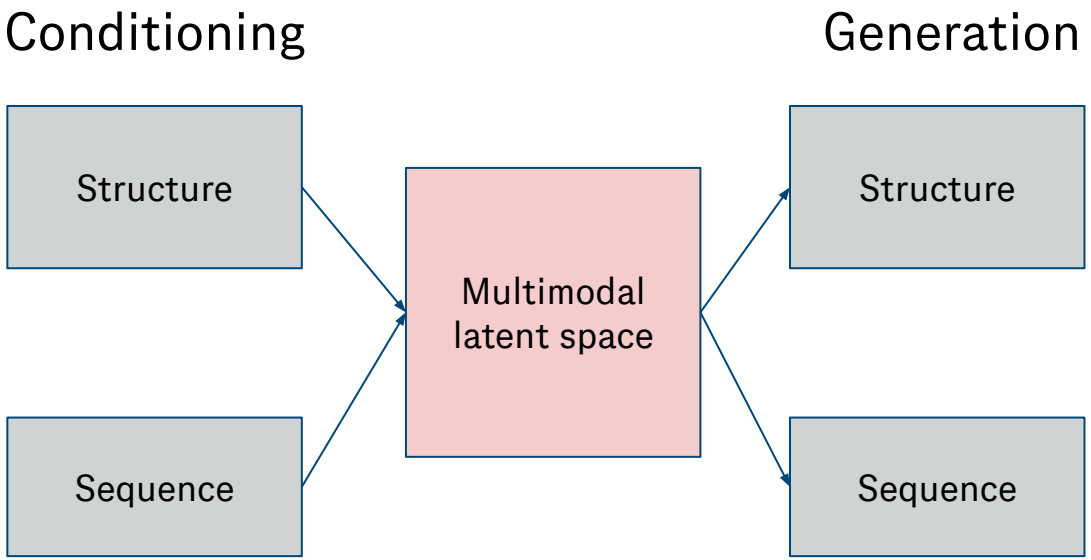
All-atom structure is a superset of sequence information!

Motivation: **Sequence databases offer better data distribution coverage and function label abundance**

- Structure databases have strong priors which may not always be useful:
  - biased towards crystallizable proteins
  - sequence database sizes approaches internet-scale data, while structure databases are much smaller



Motivation: **Directly capturing the joint distribution is flexible**



Being able to characterize a joint latent space allows flexibly conditioning by and generating either modality.

Motivation: **Direct sampling from the joint distribution is natural**

Structure generation + inverse folding

$p(\text{branches})$

Generate a picture of tree branches.

Here is the image of tree branches you requested. If you need any modifications or additional details, feel free to ask!

$p(\text{apples}|\text{branches})$

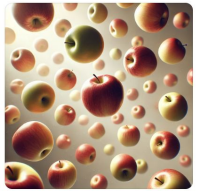
Add apples to this tree branch.

Here's the updated image with apples added to the tree branches. If you need any further adjustments, just let me know!

Sequence generation + folding

$p(\text{apples})$

Generate a picture of apples in midair.



Here is the image of apples suspended in midair. I hope it captures the surreal effect you were looking for!

$p(\text{branches}|\text{apples})$

Add branches to these apples.

Here is the updated image with branches holding the apples in place. The branche touch while maintaining the surreal effect. I hope this meets your expectations!

Co-generation

$p(\text{apples}, \text{branches})$

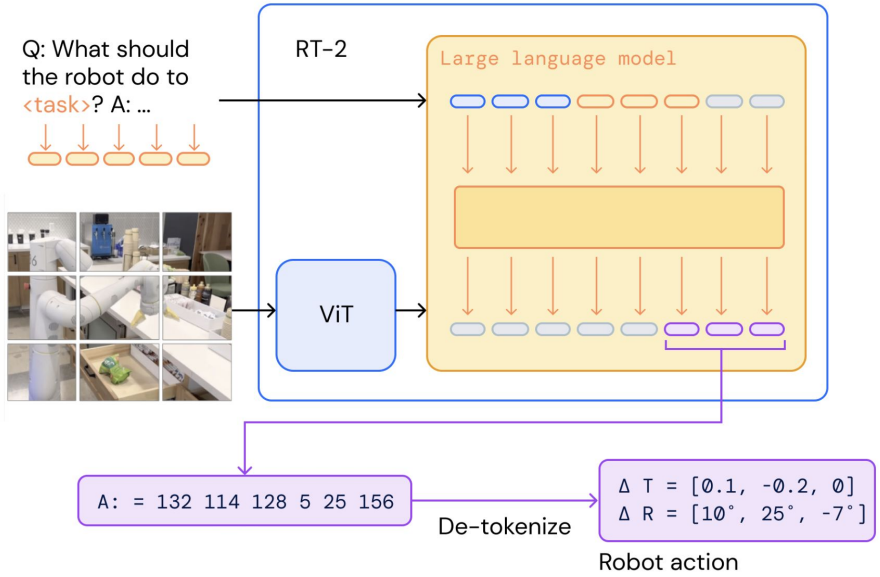
Generate a picture of an apple tree.

Here is the image of an apple tree you requested! Let me know if you need any adjustments or further assistance.

Motivation: **Large pretrained models capture useful priors for decision making**

- Multimodal pretrained models offer useful priors
  - e.g. VLMs in robotics

→ can we use information captured by AlphaFold2, etc. as a “foundation model” for decision making in protein engineering?



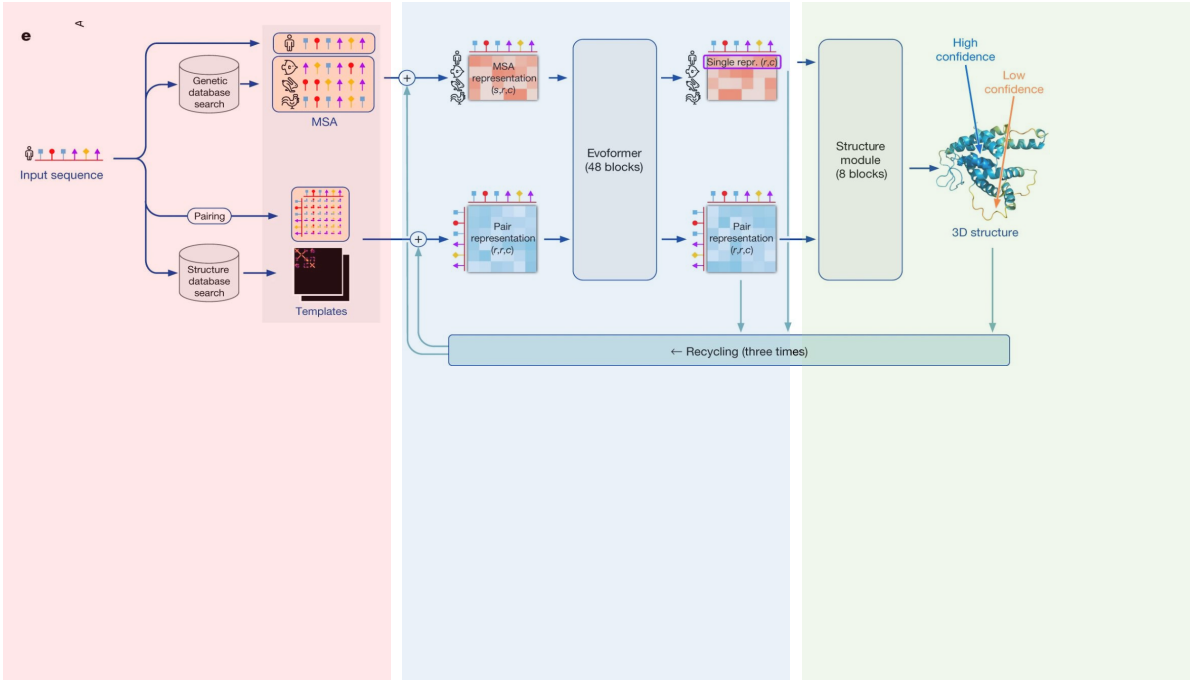
RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control

How can we repurpose the joint representation of p(sequence, structure) in protein folding models for downstream tasks?

Refresher: ESMFold for sequence-to-structure prediction

AlphaFold2:

Uses an explicit retrieval step



harness additional sequence-based priors

learn structural features from sequence latents

generate structures

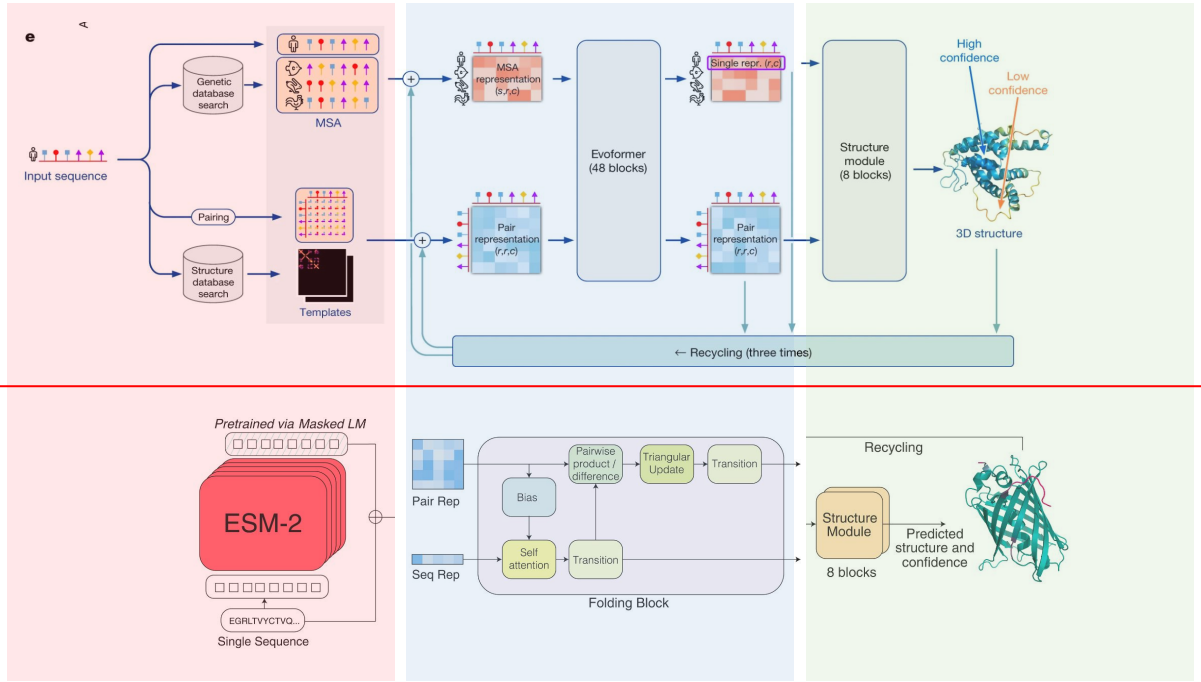
Refresher: ESMFold for sequence-to-structure prediction

AlphaFold2:

Uses an explicit retrieval step

ESMFold:

Replaces retrieval step with a **language model**



harness additional sequence-based priors

learn structural features from sequence latents

generate structures

Genentech
A Member of the Roche Group

main esm / esm / esmfold / v1 / esmfold.py ↑ Top

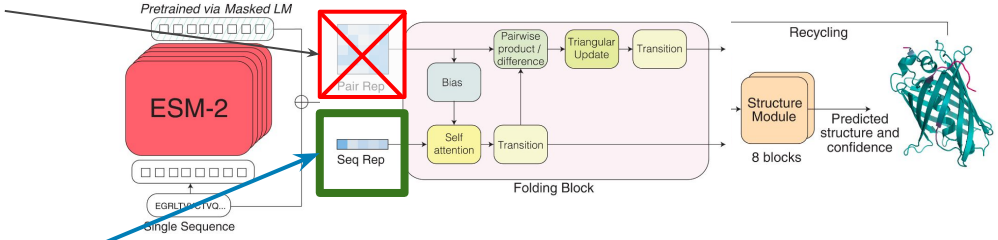
Code Blame 364 lines (305 loc) · 13.6 KB Raw

```
152     def forward(
185         # === ESM ===
186         esmaa = self._af2_idx_to_esm_idx(aa, mask)
187
188         if masking_pattern is not None:
189             esmaa = self._mask_inputs_to_esm(esmaa, masking_pattern)
190
191         esm_s, esm_z = self._compute_language_model_representations(esmaa)
192
193         # Convert esm_s to the precision used by the trunk and
194         # the structure module. These tensors may be a lower precision if, for example,
195         # we're running the language model in fp16 precision.
196         esm_s = esm_s.to(self.esm_s_combine.dtype)
197         esm_s = esm_s.detach()
198
199         # === preprocessing ===
200         esm_s = (self.esm_s_combine.softmax(0).unsqueeze(0) @ esm_s).squeeze(2)
201
202         s_s_0 = self.esm_s_mlp(esm_s)
203         if self.cfg.use_esm_attn_map:
204             esm_z = esm_z.to(self.esm_s_combine.dtype)
205             esm_z = esm_z.detach()
206             s_z_0 = self.esm_z_mlp(esm_z)
207         else:
208             s_z_0 = s_s_0.new_zeros(B, L, L, self.cfg.trunk.pairwise_state_dim)
209
210         s_s_0 += self.embedding(aa)
211
212         structure: dict = self.trunk(
213             s_s_0, s_z_0, aa, residx, mask, no_recycles=num_recycles
214         )
```

Observation: at inference time, the pairwise input is initialized as zeros…

Observation: at inference time, the pairwise input is initialized as zeros…

→ LM embedding captures sufficient inductive biases for structure, but requires **only sequence data** during training!



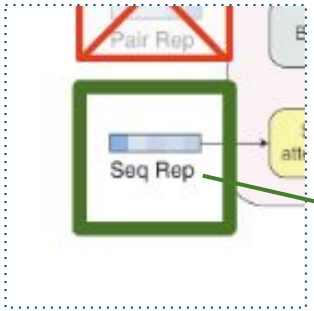
harness additional sequence-based priors

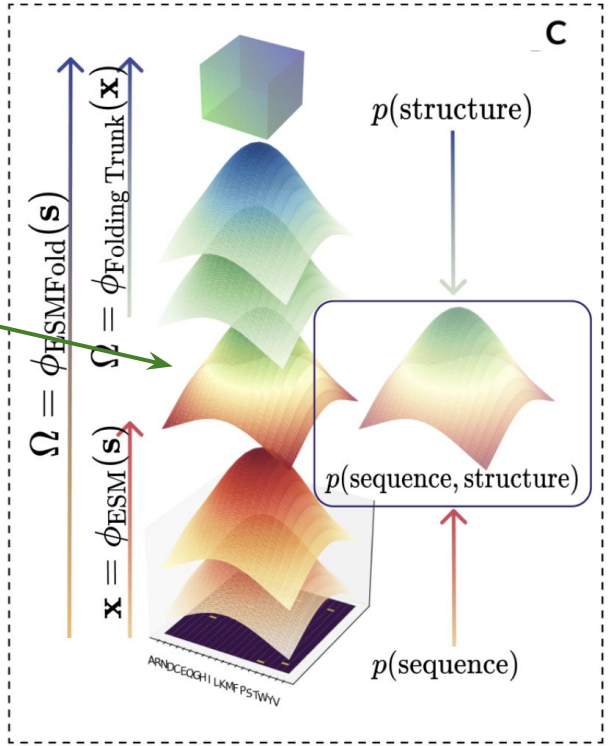
learn structural features from sequence latents

generate structures

Observation: at inference time, the pairwise input is initialized as zeros...

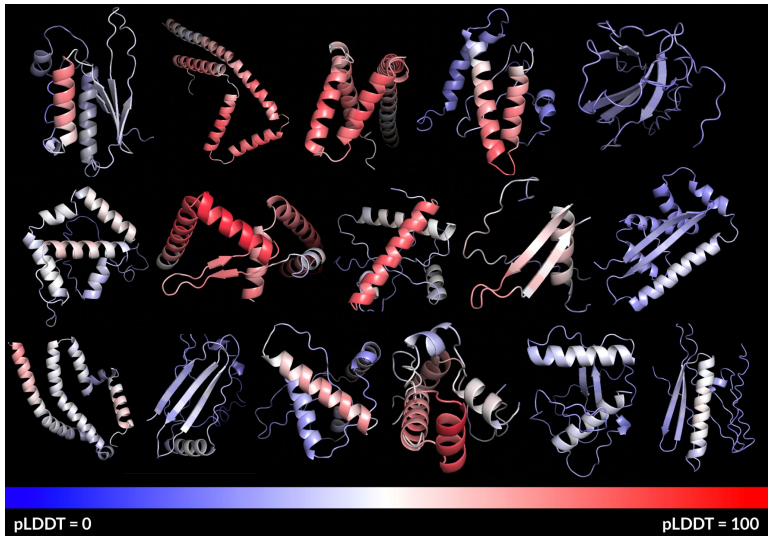
→ LM embedding captures sufficient inductive biases for structure, but requires **only sequence data** during training!





Consider this latent space as a **joint representation of protein sequence and structure that can be obtained from sequence only.**

an early attempt at diffusing in this latent space…



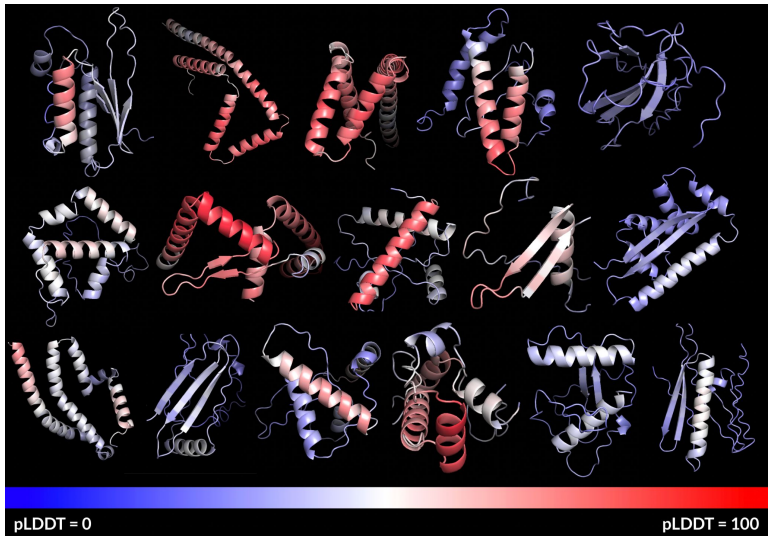
We are able to learn structural folds, despite using only sequence inputs!

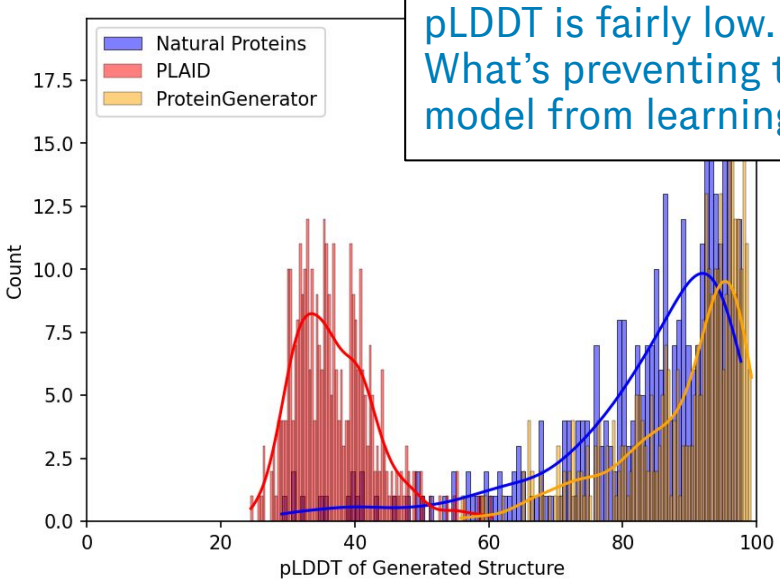
Empirically considering this latent space as a joint distribution is a go ✅

PLAID v0.5: Generating Protein Sequence and Structure Without Structural Training Data
Amy X. Lu, Kevin K. Yang, Pieter Abbeel

ICML 2024 Workshop on Machine Learning for Life and Material Sciences

an early attempt at diffusing in this latent space…





pLDDT is fairly low. What's preventing the model from learning?

PLAID v0.5: Generating Protein Sequence and Structure Without Structural Training Data
Amy X. Lu, Kevin K. Yang, Pieter Abbeel

ICML 2024 Workshop on Machine Learning for Life and Material Sciences

Genentech
A Member of the Roche Group

Issues and hypotheses -> CHEAP

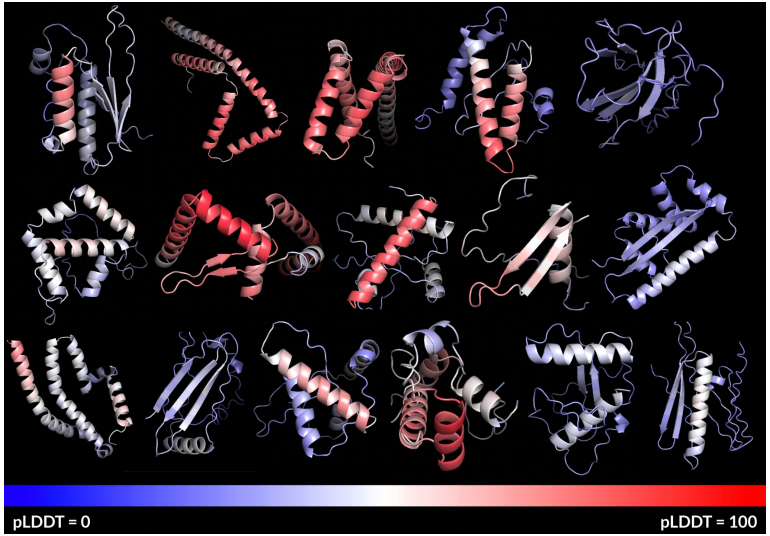
- Latent space requires regularization

> In order to avoid arbitrarily high-variance latent spaces, we experiment with two different kinds of regularizations. The first variant, *KL-reg.*, imposes a slight KL-penalty towards a standard normal on the learned latent, similar to a VAE [46, 69], whereas *VQ-reg.* uses a vector quantization layer [96] within the decoder. This model can be interpreted as a VQGAN [23] but with the quantization layer absorbed by the decoder.

High-Resolution Image Synthesis with Latent Diffusion Models

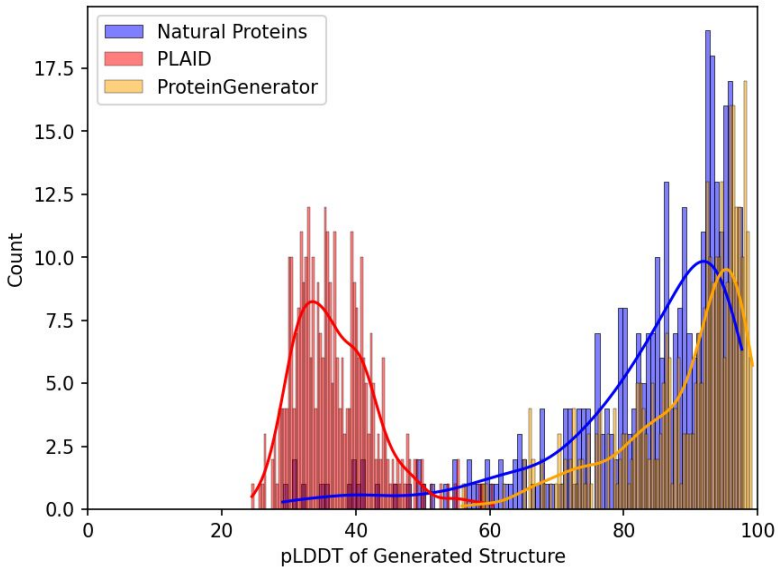
Issues and hypotheses -> CHEAP

- Latent space requires regularization
- Training data only allows for length of 128 due to memory constraints
  - Some samples show the curvatures of a beta barrel, but sequence length limits seeing a full beta barrel



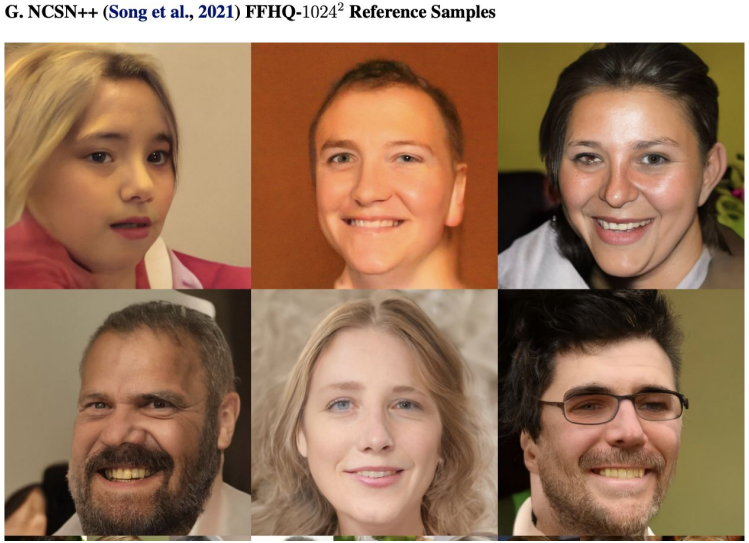
Issues and hypotheses -> CHEAP

- Latent space requires regularization
- Training data only allows for length of 128 due to memory constraints
  - Some samples show the curvatures of a beta barrel, but sequence length limits seeing a full beta barrel
    - Need to shorten the protein?
- pLDDT is not designed to assess generation from evolutionary scale datasets
  - Biased towards generative models trained on the same data as AF2, i.e. PDB



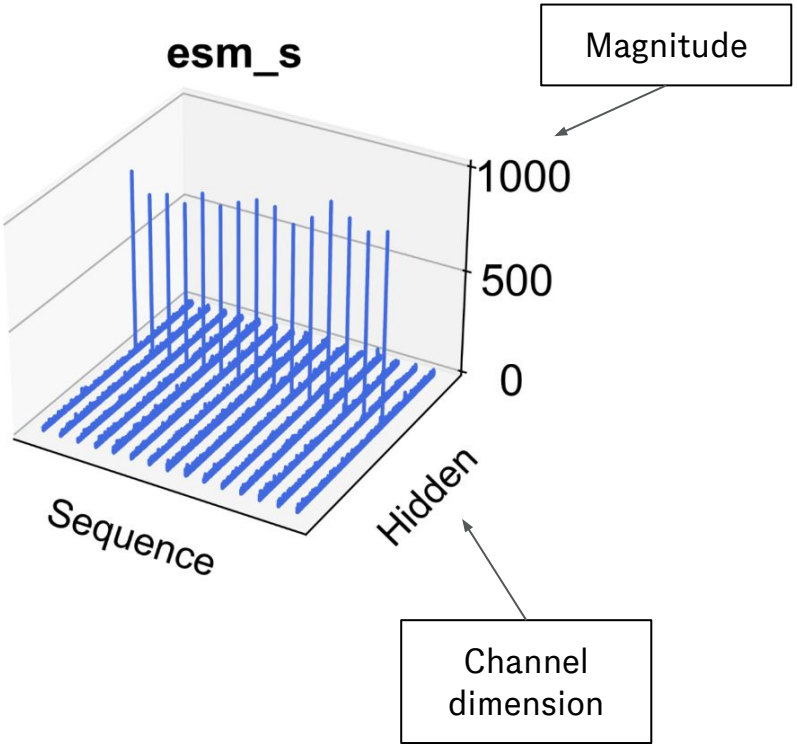
Issues and hypotheses -> CHEAP

- Latent space requires regularization
- Training data only allows for length of 128 due to memory constraints
  - Some samples show the curvatures of a beta barrel, but sequence length limits seeing a full beta barrel
    - Need to shorten the protein?
- pLDDT is not designed to assess generation from evolutionary scale datasets
  - Biased towards generative models trained on the same data as AF2, i.e. PDB
- Large latent space corresponds to high-resolution image generation
  - in LDMs, latent space is 64 x 4 x 4, as opposed to ours, which is 512 x 1024

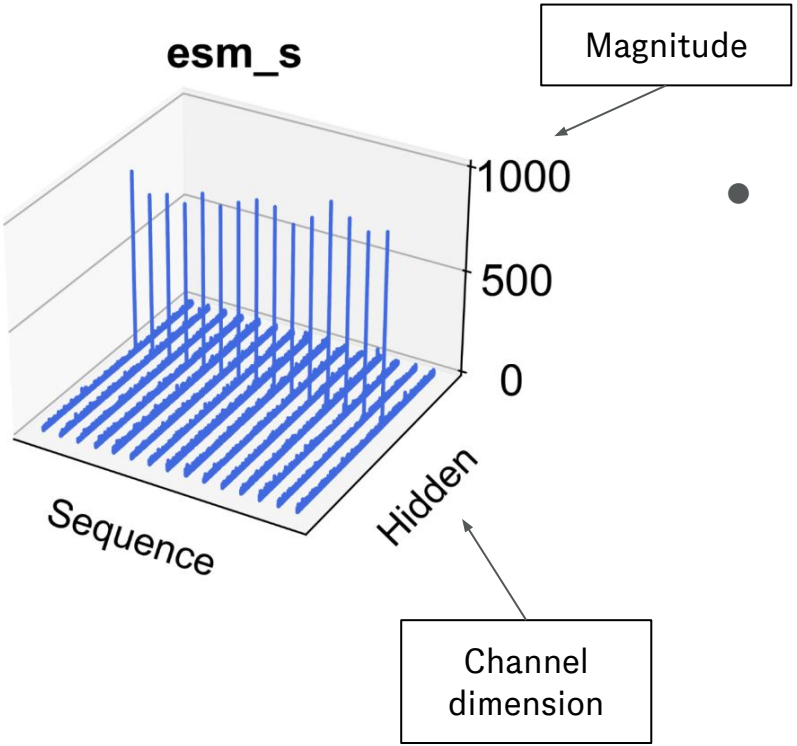


Diffusion models in their naive formulation often fail for 1024 x 1024 resolution generation.

A closer look at the latent space of ESMFold…

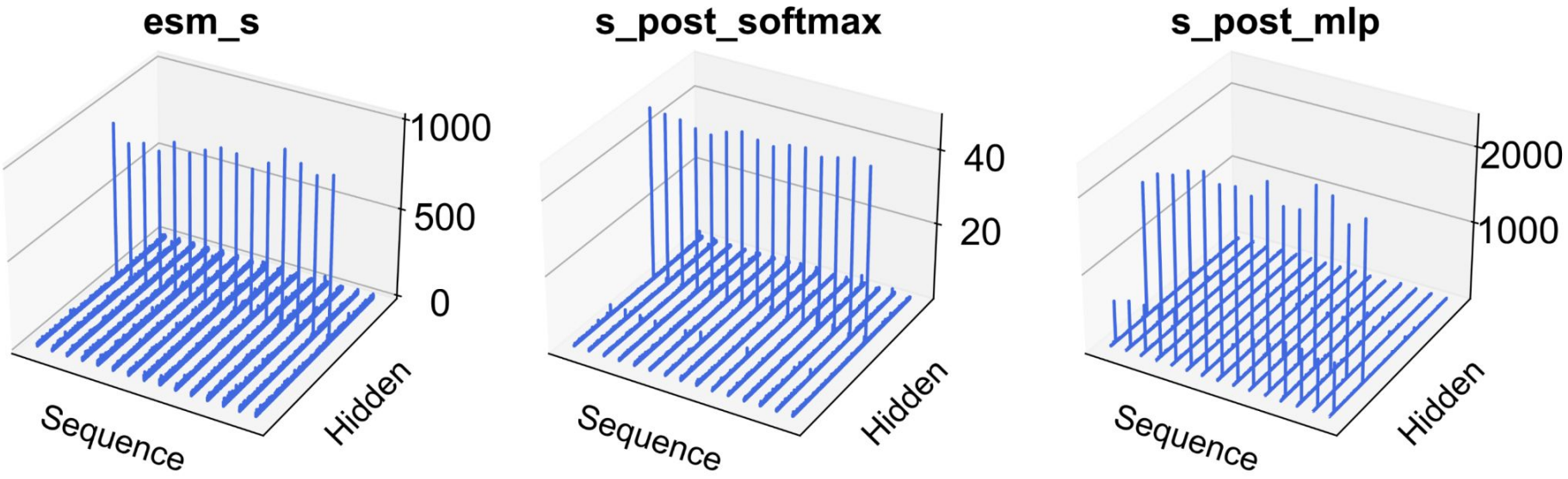


…ESMFold latent space exhibits pathologically large values



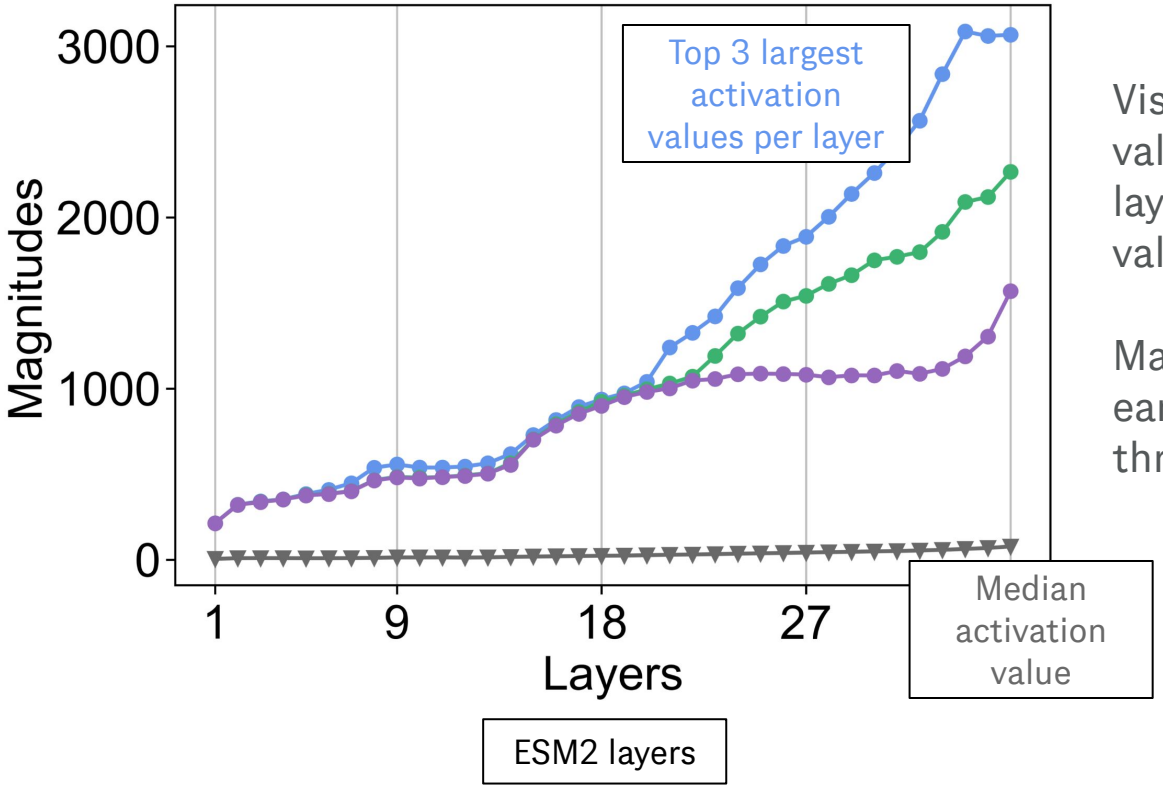
- Some channels exhibit very high mean values, regardless of the input.
  - Implications for generation: data distribution is no longer Gaussian distributed

…ESMFold latent space exhibits pathologically large values



Not just an issue for this particular layer…

~~ESMFold~~ ESM2 latent space exhibits pathologically large values



Visualizing the top 3 highest values in intermediate ESM2 layers, against the median value.

Massive activations begin in early layers, and accumulate throughout the model.

~~ESMFold~~ Large transformer model latent spaces exhibits pathologically large values

A pervasive issue across large transformer models!

[Submitted on 27 Feb 2024 (v1), last revised 14 Aug 2024 (this version, v2)]

Massive Activations in Large Language Models

Mingjie Sun, Xinlei Chen, J. Zico Kolter, Zhuang Liu

We observe an empirical phenomenon in Large Language Models (LLMs) -- very few activations exhibit significantly larger values than others (e.g., 100,000 times larger). We call them massive activations. First, we demonstrate the widespread existence of

~~ESMFold~~ Large transformer model latent spaces exhibits pathologically large values

A pervasive issue across large transformer models!

[Submitted on 27 Feb 2024 (v1), last revised 14 Aug 2024 (this version, v2)]

Massive Activations in Large Language Models

Mingjie Sun, Xinlei Chen, J. Zico Kolter, Zhuang Liu

We observe an empirical phenomenon in Large Language Models (LLMs) -- very few activations exhibit significantly larger values than others (e.g., 100,000 times larger). We call them massive activations. First, we demonstrate the widespread existence of

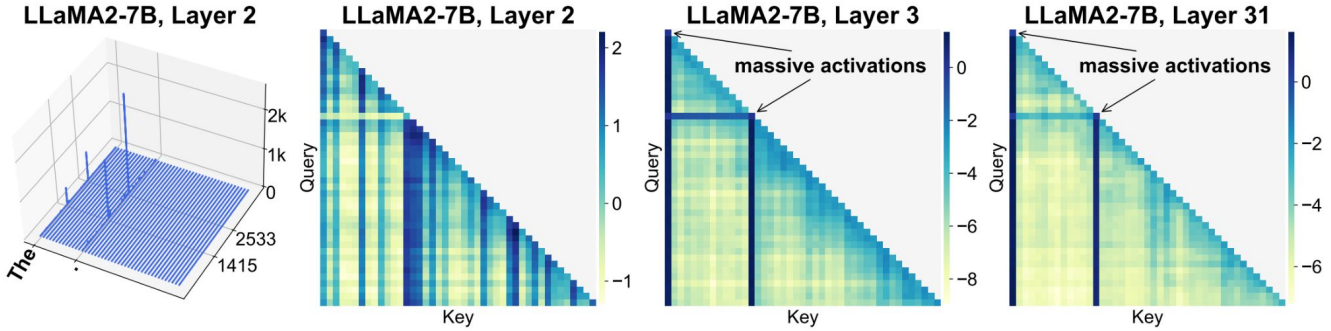
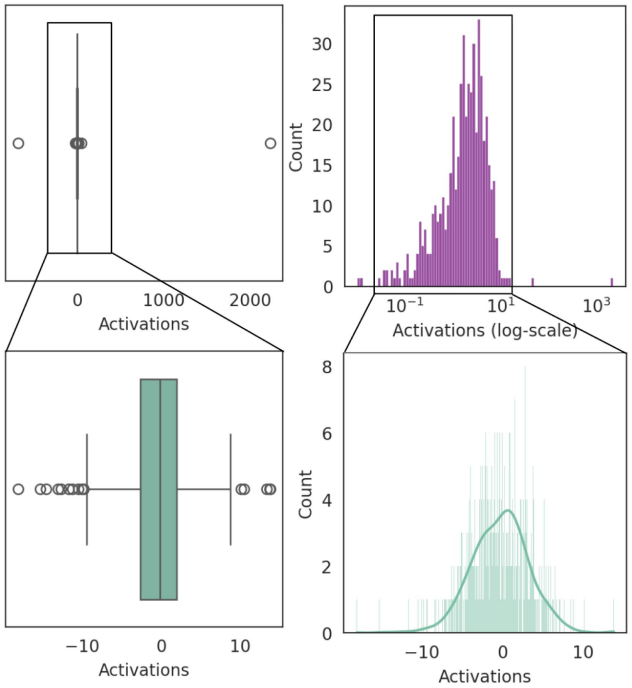


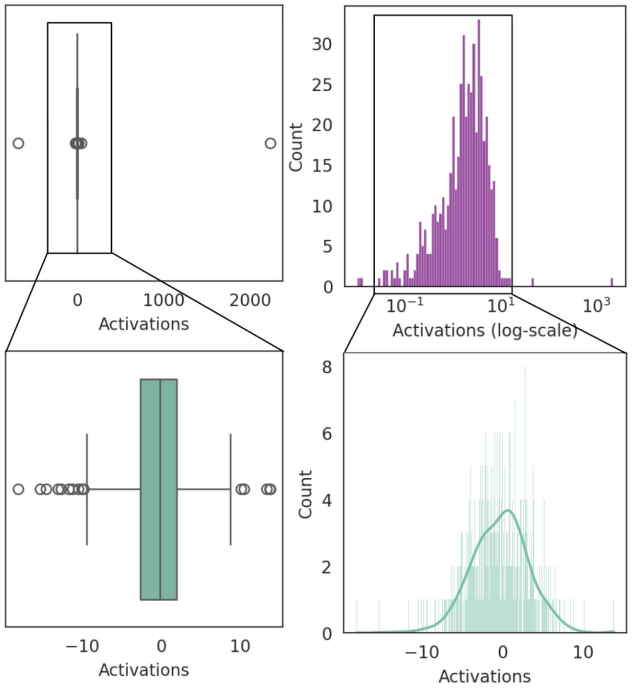
Figure 5: Attention patterns *before* and *after* massive activations appear in LLaMA2-7B. For each layer, we visualize average attention logits (unnormalized scores before softmax) over all heads, for an input sequence.

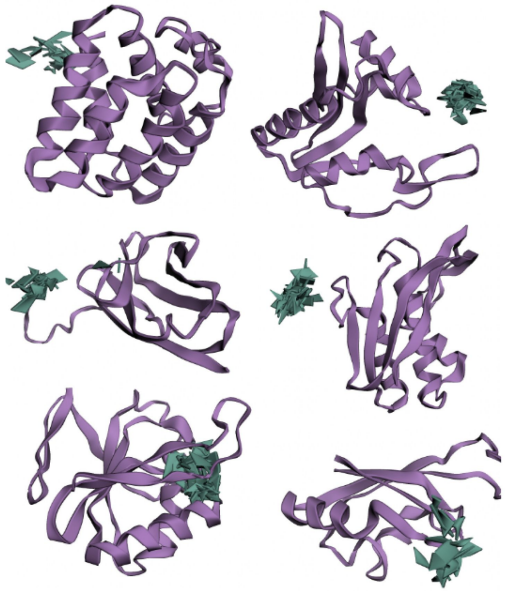
Genentech
A Member of the Roche Group

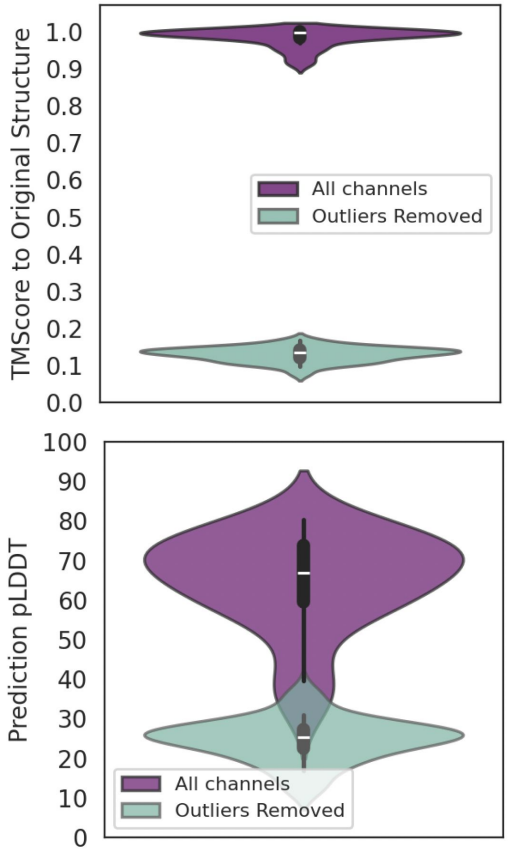
What if we just remove these wacky channels?



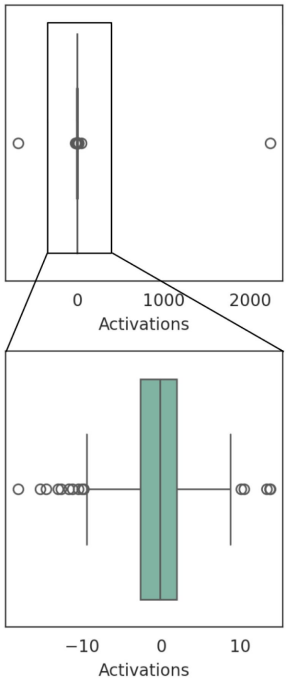
What if we just remove these wacky channels?

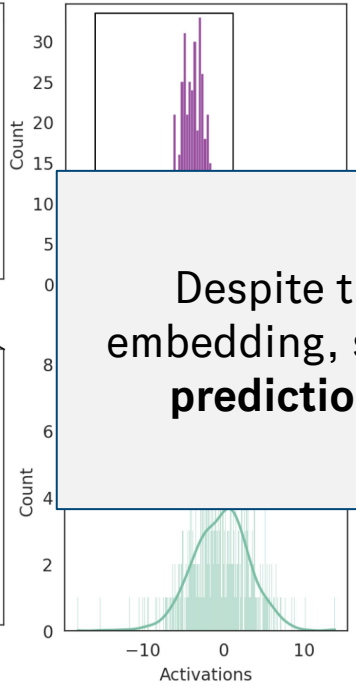




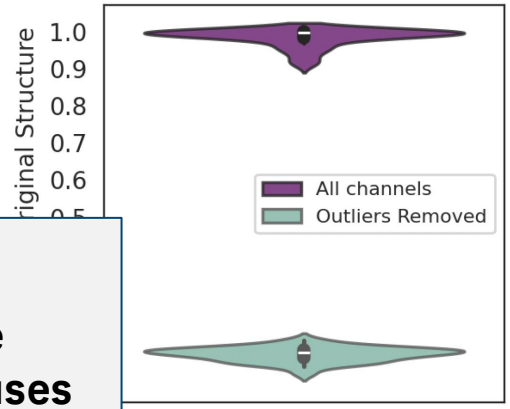


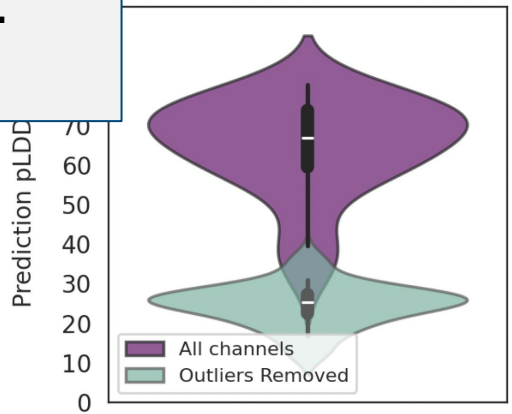
What if we just remove these wacky channels?





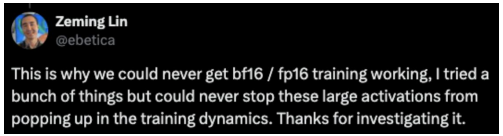
Despite there being **1024** channels in the embedding, simply **removing 3 channels causes prediction ability to entirely deteriorate.**





Why should we care about these massive activations?

- Training stability
- Model compression and 8-bit quantization
- Model interpretability
- ...



`LLM.int8()`: **8-bit Matrix Multiplication for Transformers at Scale**

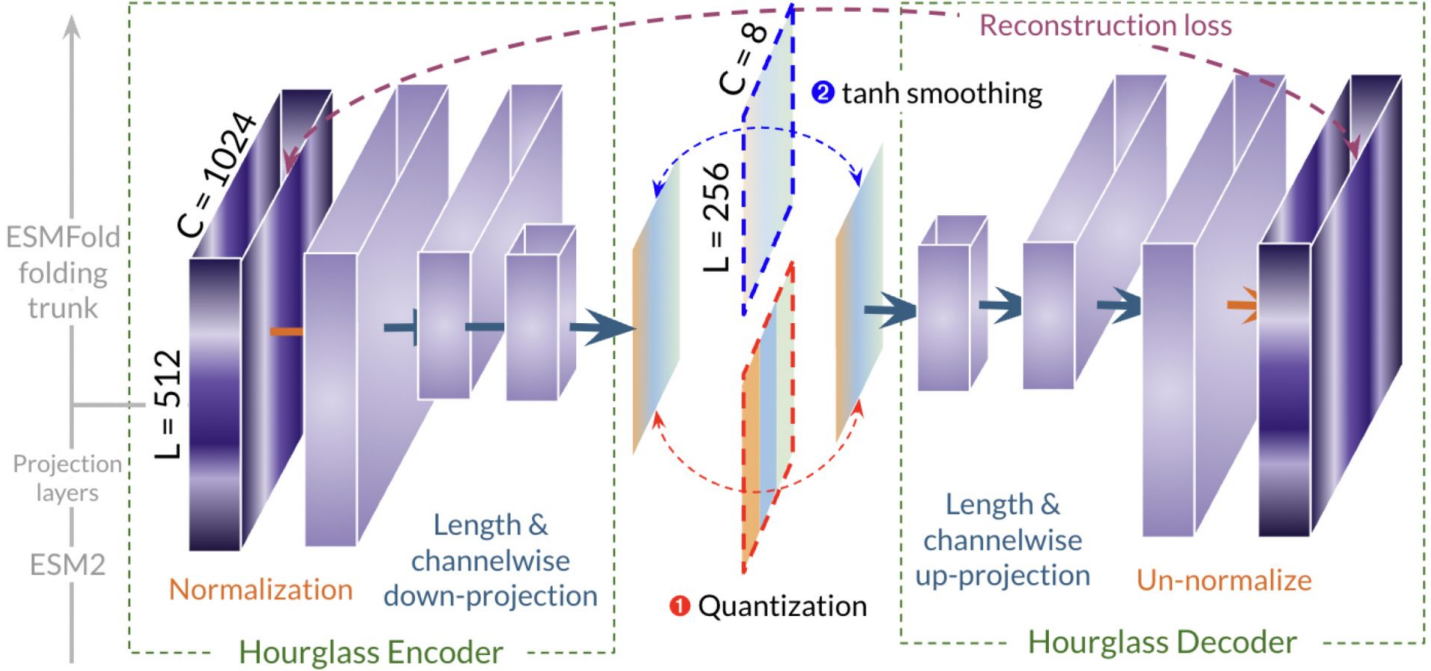
If removing 3 channels can remove performance, is the information evenly distributed through all the channels?

If not, can we **compress these channels?**

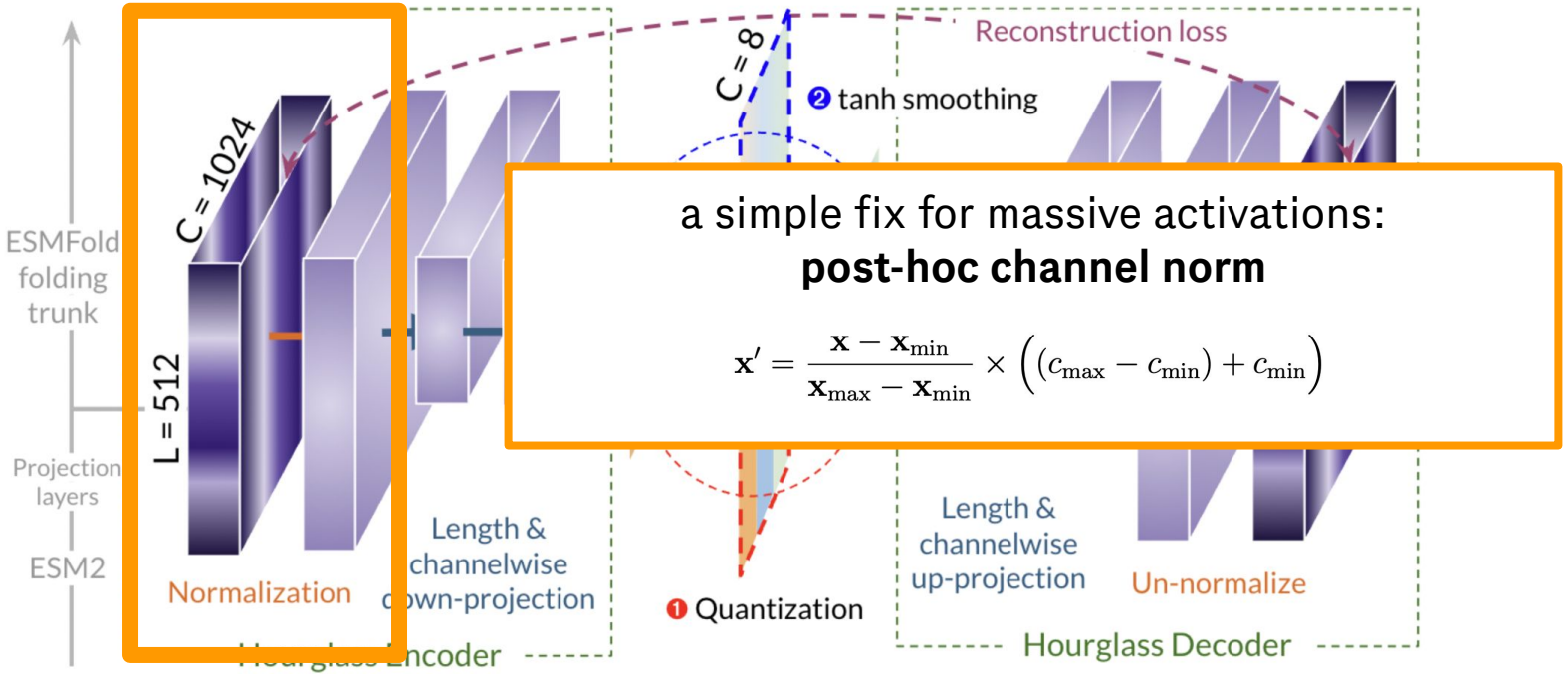
Why compress?

- More portable representation
- Better understanding of protein folding internals
- Compressed data distributions are easier to learn during generative modeling

An autoencoder for protein embedding compression



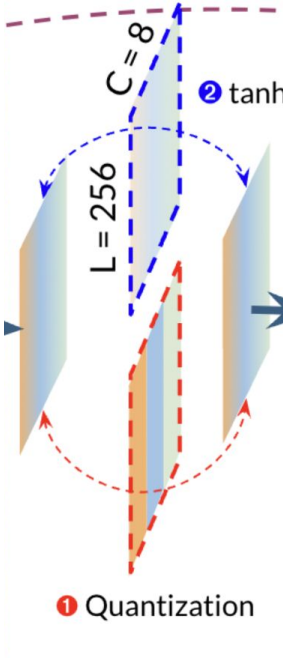
An autoencoder for protein embedding compression



Obtaining CHEAP embeddings

1. Tokenized

- Discretize embeddings using FSQ
  - ‘snaps’ continuous encoder values to discrete bins

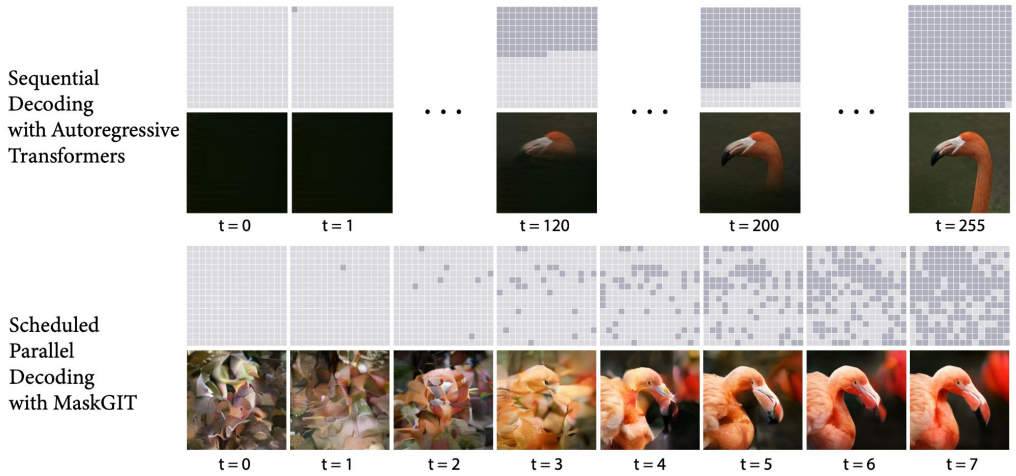


2. Continuous

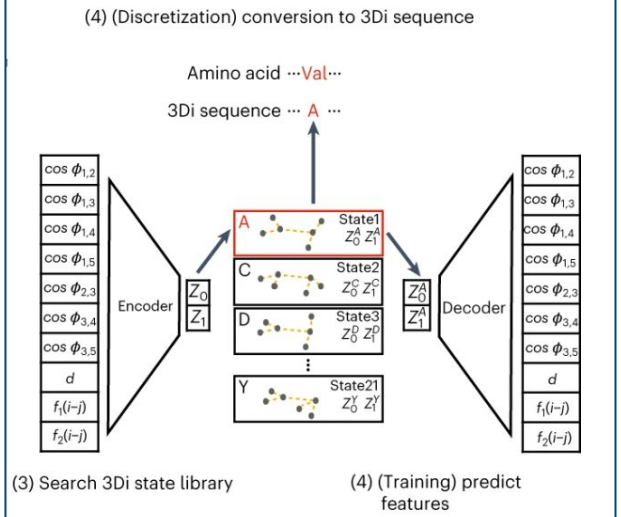
- Take the output of the downprojecting autoencoder
  - apply tanh to bound values between [-1, 1], to bound values during diffusion

Side note: why tokenized representations?

Tokenized representations can be helpful for our downstream aims of generation and search:

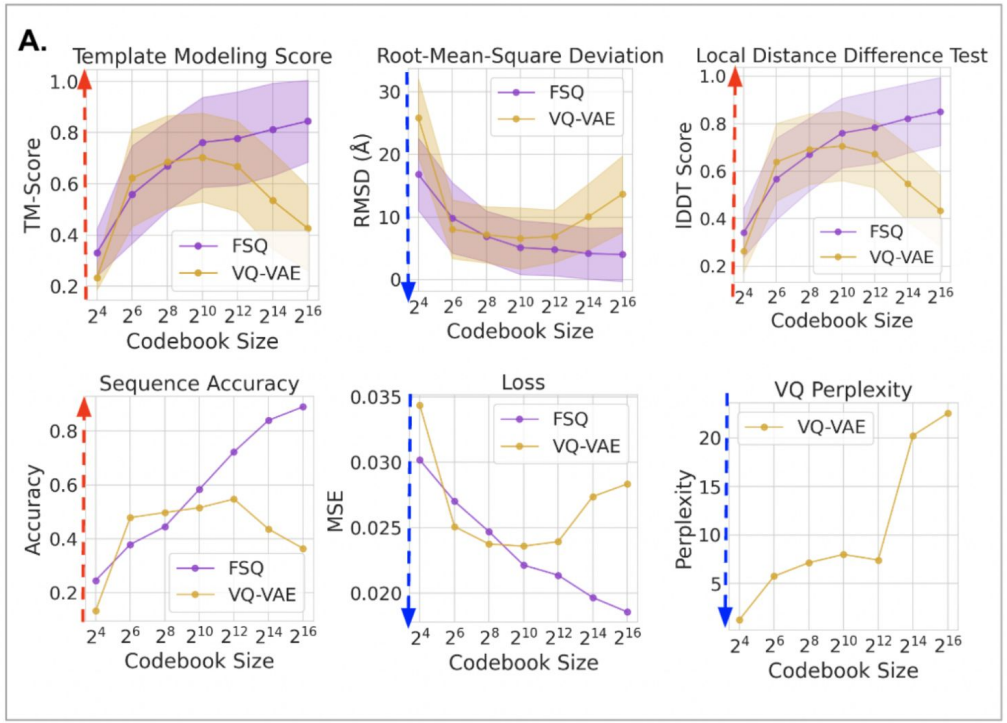


MaskGIT: Masked Generative Image Transformer

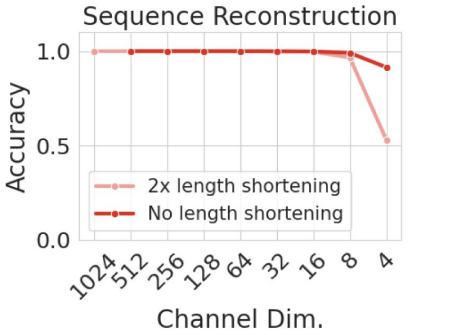


Foldseek

All-atom structural tokenizer, obtained from sequence alone



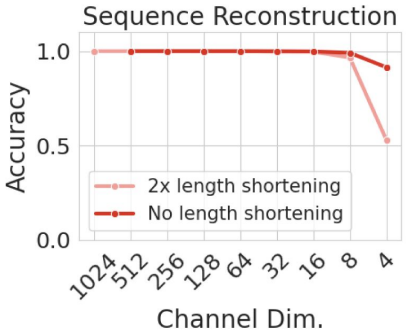
..yes, we *can* compress the embeddings:



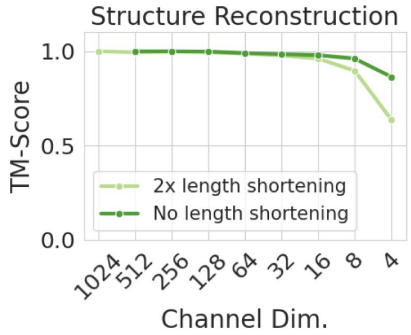
Compression performance with respect to sequence:

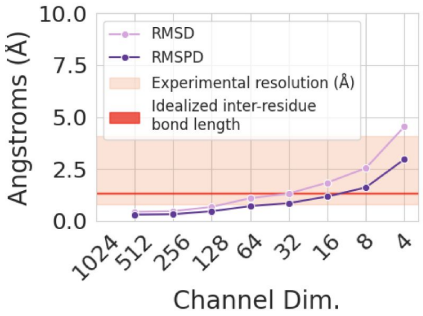
We can compress up to 8x, and sacrifice very little performance.

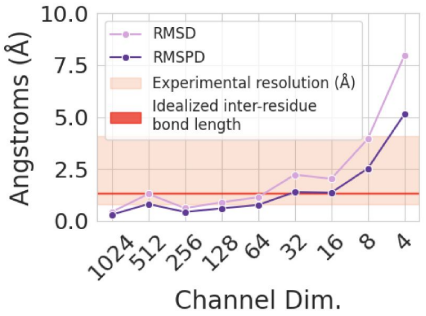
..yes, we *can* compress the embeddings:



Compression performance with respect to sequence:



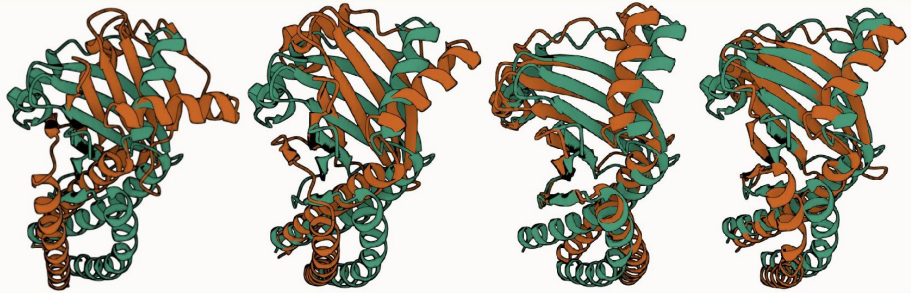


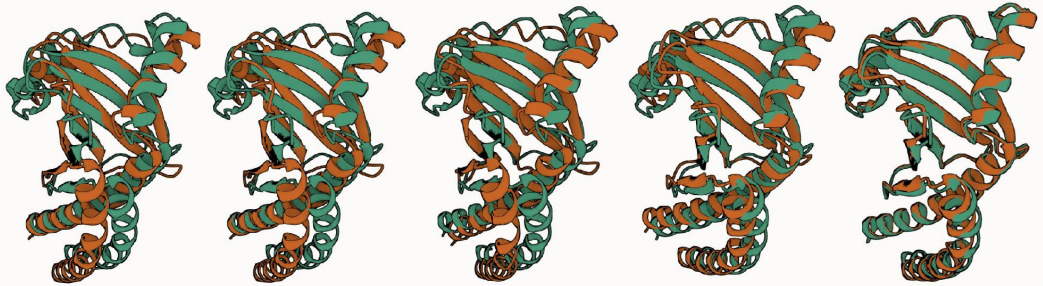


Compression performance with respect to structure:

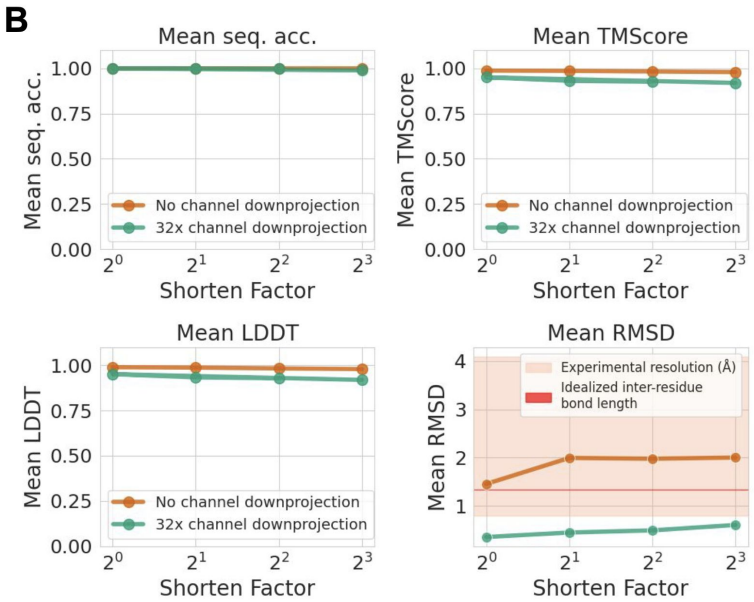
Sequence information is easier to retain than structure.

..yes, we *can* compress the embeddings:

Dim.	4	8	16	32
				
TMScore	0.789	0.920	0.968	0.995

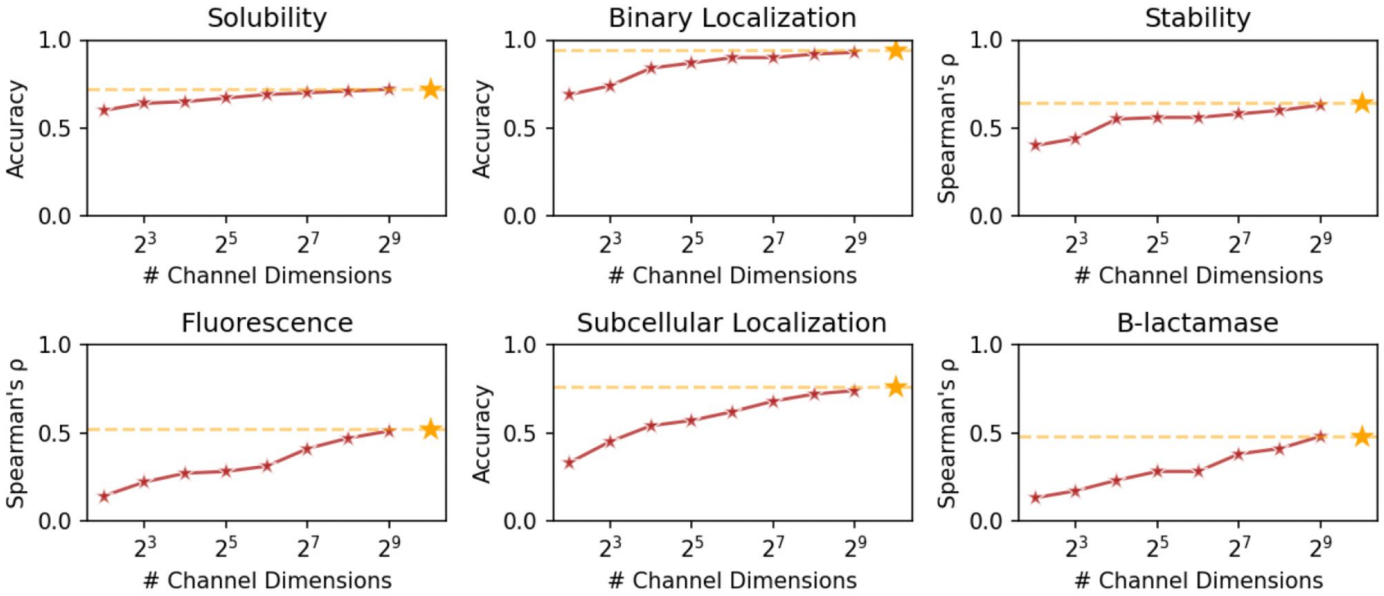
64	128	256	512	1024
				
0.991	0.999	1.000	0.997	1.000

We can compress lengthwise and channelwise:



What does this mean for how structural information is shared across residue positions?

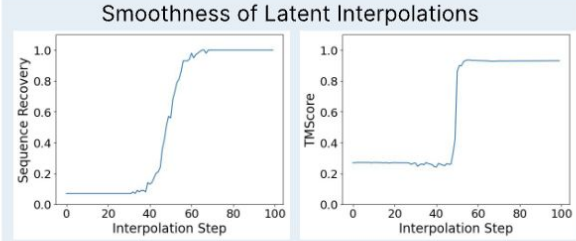
What about function information?



Performance degradation with compression is much more gradual.
What does this imply about the information content captured in pLMs with respect to downstream tasks?

Does the autoencoding scheme “fix” the irregular latent space?

Original

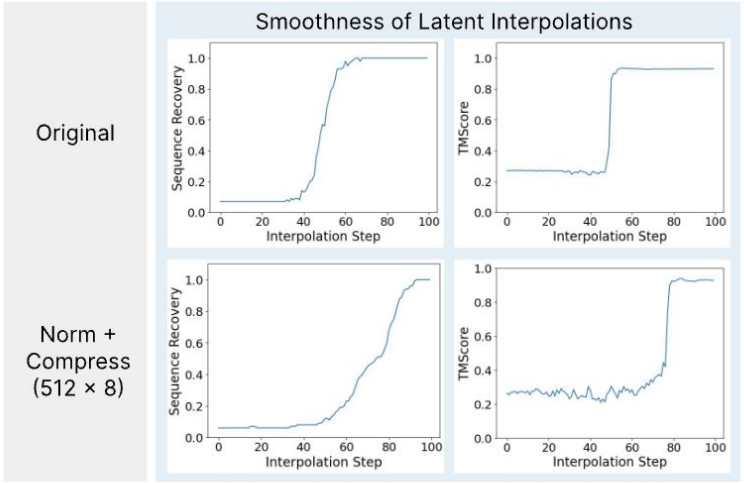


- Despite linearly interpolating in the latent space, the decoded sequence and structure changes very abruptly.

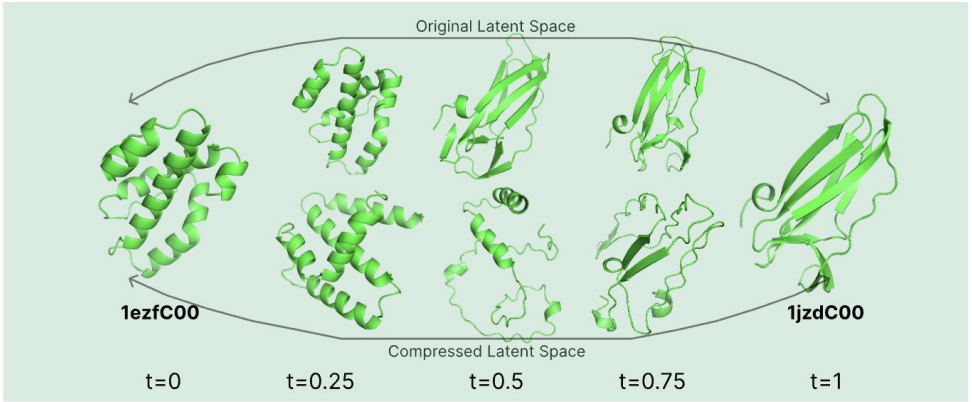
sequence space

structure space

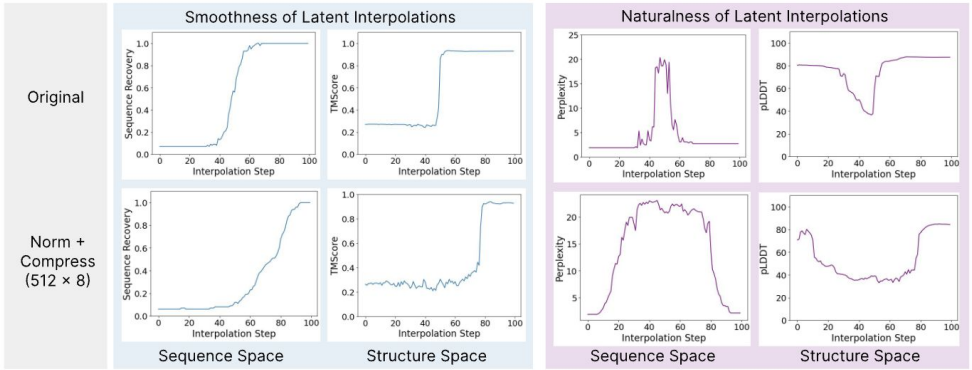
Does the autoencoding scheme “fix” the irregular latent space?

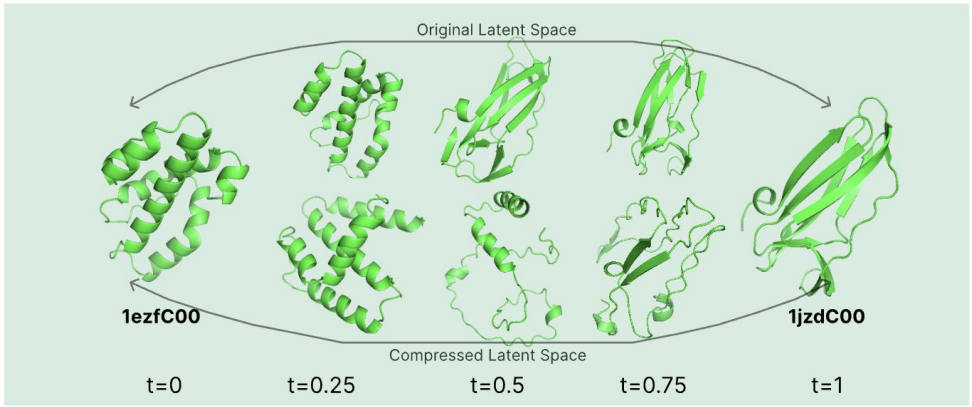


- Despite linearly interpolating in the latent space, the decoded sequence and structure changes very abruptly.
- After CHEAP regularization, the change is more gradual



PLM latent manifolds might be less “rugged” than true protein fitness landscapes





What makes for a good latent space?

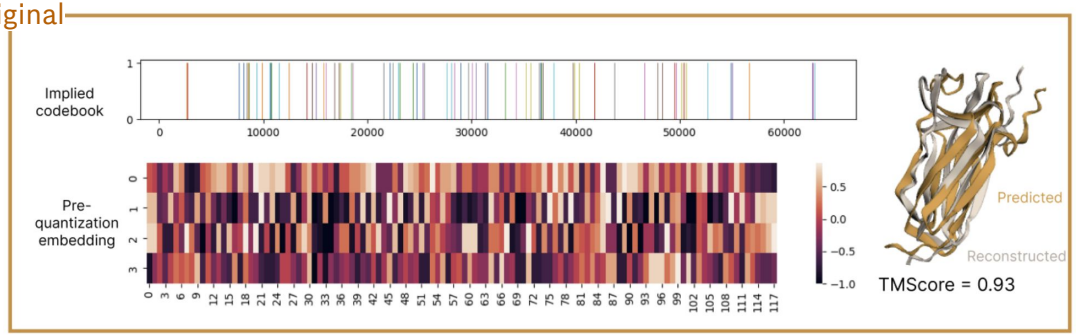
Should we want more of the latent space to map back to a “valid protein” for sampling purposes, or properly model the rugged protein landscape?

Do current PLM embeddings actually recapitulate protein fitness landscapes?

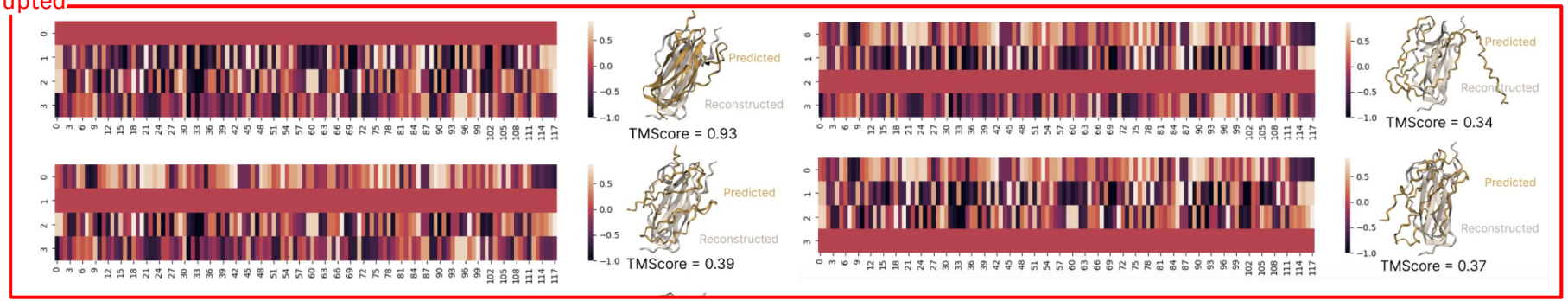
Genentech
A Member of the Roche Group

“Disrupting” and reconstructing in the token space

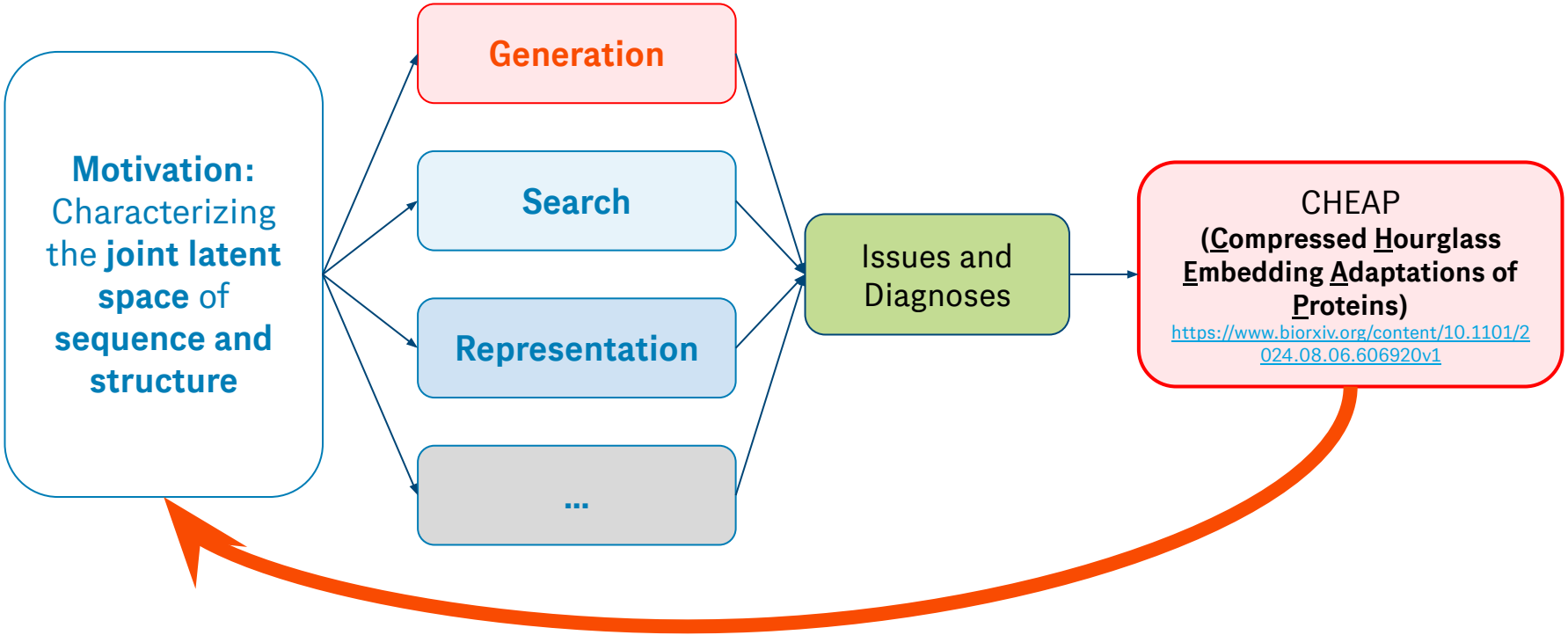
original



corrupted



Agenda

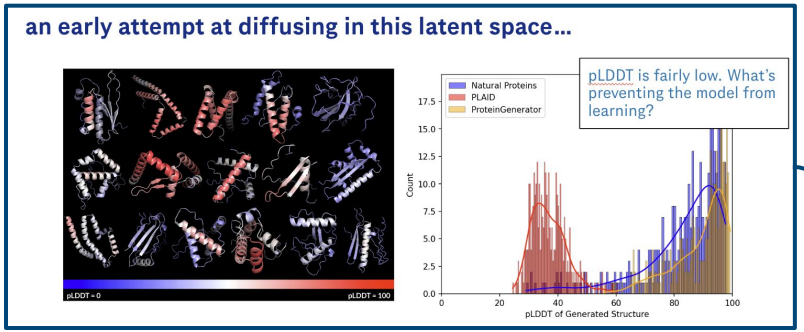

Motivation: Characterizing the joint latent space of sequence and structure
Generation
Search
Representation
...
Issues and Diagnoses
CHEAP
(Compressed Hourglass Embedding Adaptations of Proteins)
https://www.biorxiv.org/content/10.1101/2024.08.06.606920v1

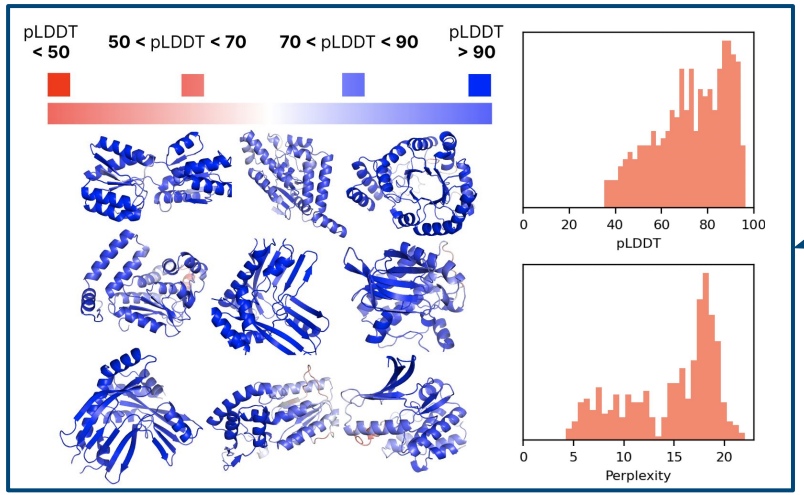
PLAID (Protein LAtent Induced Diffusion)

ongoing work!

tl;dr – now that we have a regularized & compressed embedding of p(sequence, structure), can we train a latent diffusion model for co-generation?

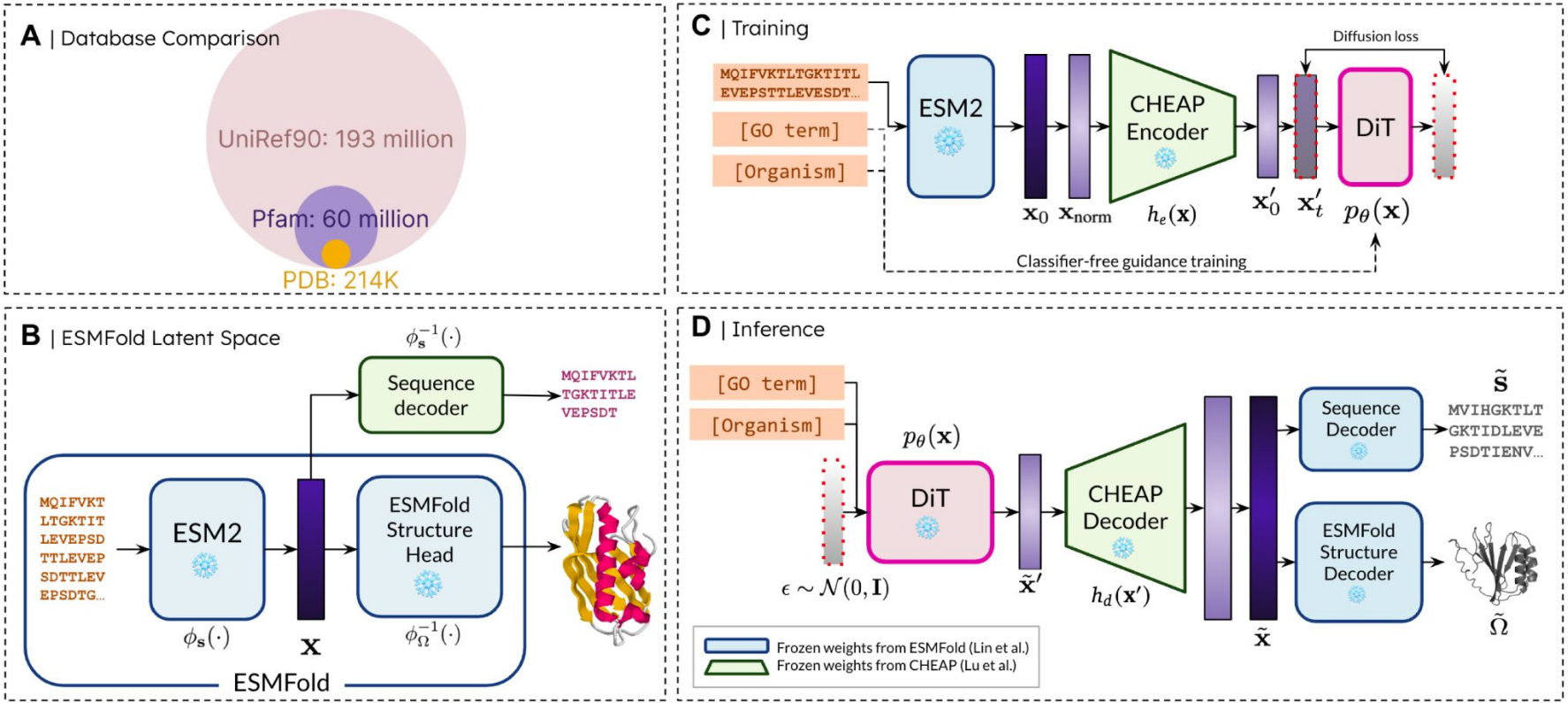
PLAID, again



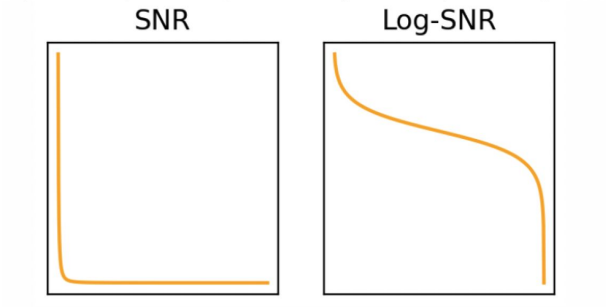


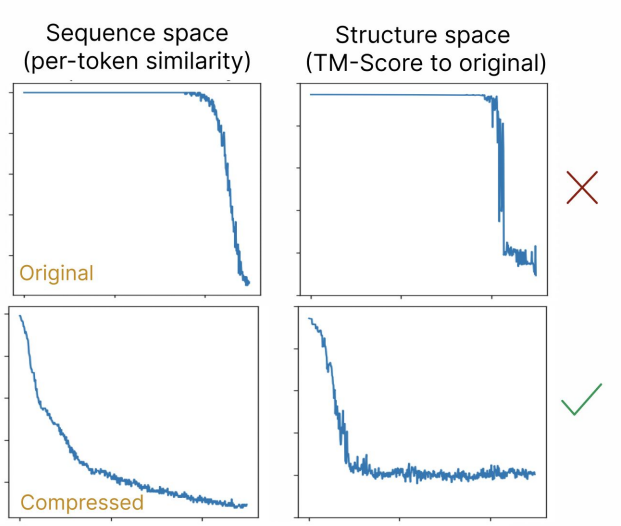
- Learn diffusion model in regularized and compressed latent space
  - mirrors the regularized autoencoder in LDM
- Can learn on longer sequences due to CHEAP shortening
- Use DiT instead of U-triangular self attention
  - allows for scaling up to higher parameter counts
- Scale up to 2B parameters with BS=2048

PLAID, again


A | Database Comparison
UniRef90: 193 million
Pfam: 60 million
PDB: 214K
C | Training
MQIFVKTLTGKTITL
EVEPSTTLEVESDT...
[GO term]
[Organism]
ESM2
CHEAP Encoder
DiT
Diffusion loss
$\mathbf{x}_0$ $\mathbf{x}_{\text{norm}}$ $h_e(\mathbf{x})$ $\mathbf{x}'_0$ $\mathbf{x}'_t$ $p_\theta(\mathbf{x})$
Classifier-free guidance training
B | ESMFold Latent Space
$\phi_{\mathbf{s}}^{-1}(\cdot)$
Sequence decoder
MQIFVKTL
TGKTITLE
VEPSDT
MQIFVKT
LTGKTIT
LEVEPSD
TTLEVEP
SDTTLEV
EPSDTG...
ESM2
ESMFold Structure Head
$\phi_{\mathbf{s}}(\cdot)$
$\mathbf{x}$
$\phi_\Omega^{-1}(\cdot)$
ESMFold
D | Inference
[GO term]
[Organism]
$p_\theta(\mathbf{x})$
DiT
$\epsilon \sim \mathcal{N}(0, \mathbf{I})$
$\tilde{\mathbf{x}}'$
CHEAP Decoder
$h_d(\mathbf{x}')$
$\tilde{\mathbf{x}}$
Sequence Decoder
ESMFold Structure Decoder
$\tilde{\mathbf{s}}$
MVIHGKTLT
GKTIDLEVE
PSDTIENV...
$\tilde{\Omega}$
Frozen weights from ESMFold (Lin et al.)
Frozen weights from CHEAP (Lu et al.)

Comparing noise schedules in original and compressed latent space:



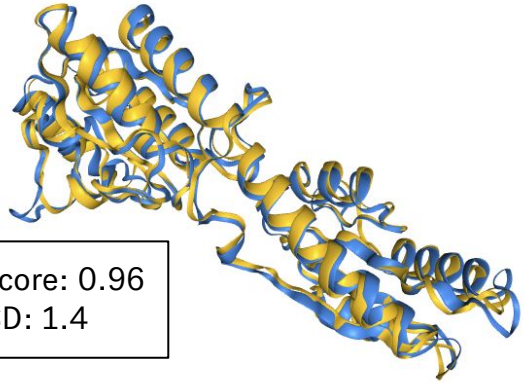


Noising in the CHEAP compressed space maps to noise in the sequence and structure space that is is closer to the true signal-to-noise ratio.

Samples demonstrate sequence and structural conservation

prompt: **“yeast” AND “6-phosphofructokinase activity”**

Search against the **structure database (PDB100)** to see if our samples are sensible…



TMScore: 0.96
RMSD: 1.4

- closest match: 3o8o [**Structure of phosphofructokinase**]
- organism: **Saccharomyces cerevisiae** (i.e. yeast)
- Sequence identity: 47.9%

Search against the **sequence database (UniRef90)** to see if our samples are sensible…

Score	Expect	Method	Identities	Positives	Gaps
327 bits(838)	3e-102	Compositional matrix adjust.	151/298(51%)	219/298(73%)	4/298(1%)

```
Query  2    MAIVNVGAPASGLNSAVRSLVRHCLSQGHTALAVINGFNGLCK--NDSAMKII-ECKWEQ  58
            +AI++VGAPA G+NSA R+ V +CL++GHT +A+ NGF GLC+  +D  +  + E KW
Sbjct  409  IAIIHVGAPAGGMNSATRAAVAYCLTRGHTPIAIHNGFPGLCRHHDDKPLGAVREVKWLD  468

Query  59   VNLWFAKGGSQFGTARTIFNSNDLELIFDKFEANEINGLLIIGGFNSYNALTVLRHNRQQ  118
            V  W +KGGS+ GT R++  S D+E     FE  + +GL +IGGF ++ A+  LR  R+
Sbjct  469  VEGWISKGGSEIGTNRSL-PSEDMEQTAKCFEQYKFDGLFVIGGFEAFTAVGELRKARKD  527

Query  119  YPEFKIPMIIIPATISNNVPGTAYSLGSESSLNALCTCVDKIKQTASASKRRVFVVETLG  178
            YP F IP++I+PATISNNVPGT YS+GS++ LNAL +  D IKQ+ASA++RRVFVVET G
Sbjct  528  YPAFNIPIVILPATISNNVPGTEYSIGSDTCLNALVSYCDAIKQSASATRRRVFVVETQG  587

Query  179  GTSGYIATMAGVCCGARSIYLPEQGIDLHKLDKDCDFLKQAFDKDSPYNKSGRIIIKNEA  238
            G SGYIAT+AG+  GA ++Y PE+GID+  L +D + L+++F  D   N++G++I++NE
Sbjct  588  GRSGYIATIAGLSIGATAVYTPEEGIDIKMLSRDIEHLRESFANDKGQNRAGKLILRNEH  647

Query  239  ASKVYSTNIIAQLIRDESNGKFDTRTSIPGHQQKGGTPTSLDRVYATKMGAKAMHFIK  296
            ASK Y+T +IA +IR+ES G+F++R ++PGH Q+GGTP+ +DRV A ++  K M FI+
Sbjct  648  ASKTYTTELIANMIREESKGRFESRLAVPGHVQQGGTPSPMDRVRAVRLAVKCMQFIE  705
```

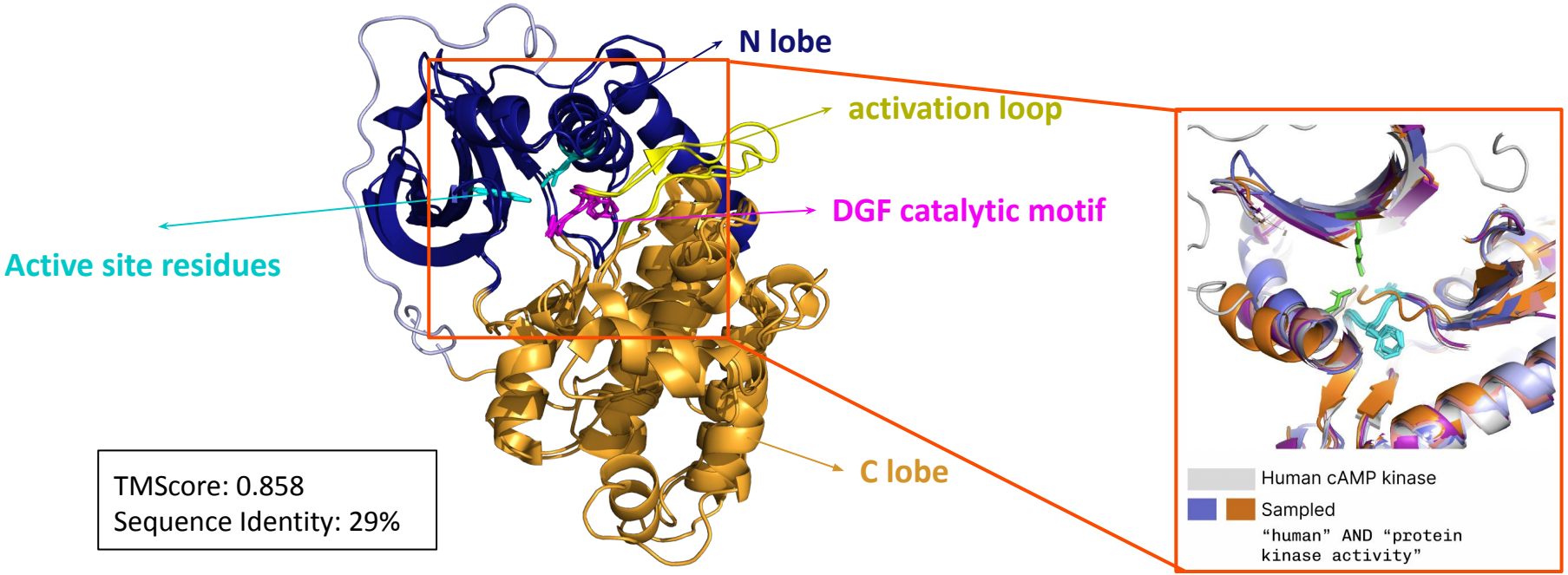
- closest match: PFK1 [**6-phosphofructokinase, alpha subunit**]
- organism: **Hypocenomyce scalaris** (also in the fungus kingdom)
- sequence identity: 50.67%

Genentech
A Member of the Roche Group

Examining active site conservation

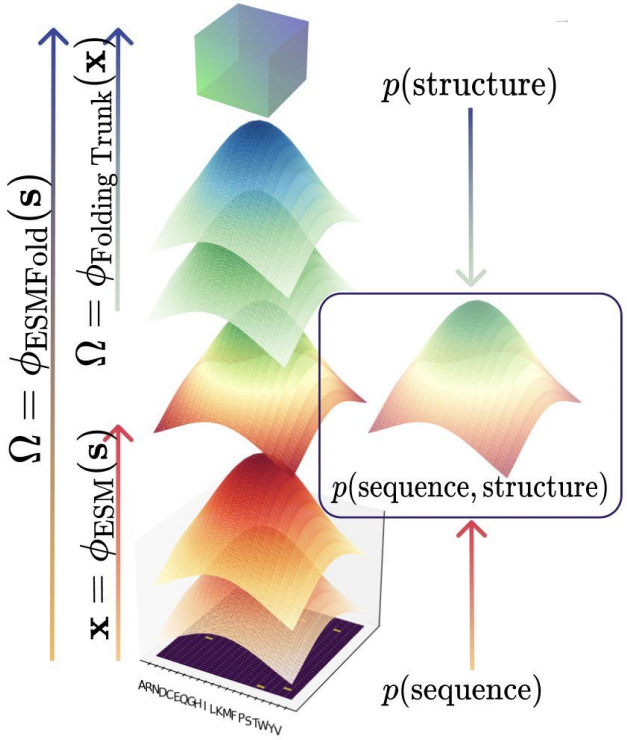
prompt: **“human” AND “protein kinase activity”**

Closest Foldseek neighbor: **6cd6 (human calcium/calmodulin-dependent protein kinase kinase 1)**

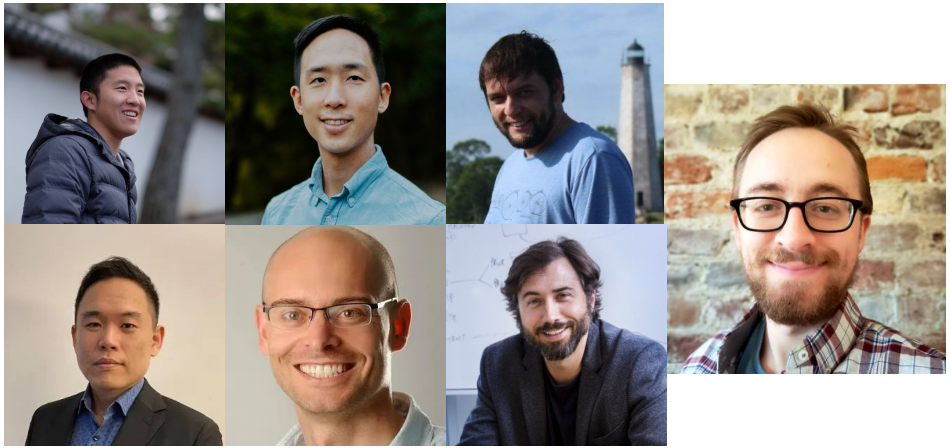


Takeaways

- The latent space of ESMFold is disorganized with massive activations
- Compressing the latent space shows that *many* channels might be extraneous for structure prediction
- Information content relating to sequence, structure, and function is not symmetrical
- CHEAP regularization helps with latent diffusion model training, leading to **an all-atom co-generation model with sequence database scale coverage**



Thanks!



Berkeley
Amy X. Lu
Wilson Yan
Pieter Abbeel

Microsoft Research
Kevin Yang

Prescient Design
Sai Pooja Mahajan
Sarah Robinson
Vladimir Gligorijevic
Kyunghyun Cho
Richard Bonneau
Nathan C. Frey

Paper: bit.ly/cheap-proteins

Code & weights: github.com/amyxlu/cheap-proteins

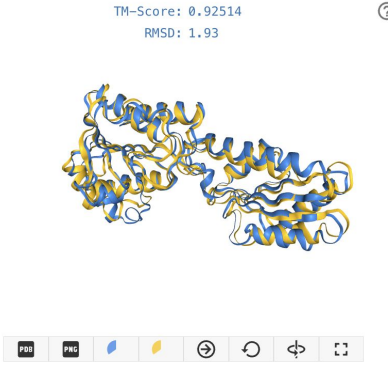
Paper

GitHub

@amyxlu amyxlu.github.io amyxlu@berkeley.edu

prompt: **“mouse” AND “6-phosphofructokinase activity”**

4xz2
human
e-value=48.2



3o8n
rabbit
e-value=46.5

- species conditioning is biased database composition
  - e.g. performance on “HUMAN” and “ECOLI” is better, since they are better represented in the database

Genentech
A Member of the Roche Group

Why GO terms and organism?

- generative protein design should propose designs that might be useful. What are some possible use cases?
  - being able to express in model organisms
  - humanization efforts
  - enzyme engineering

Organism: encourages generating samples that might express.
GO term: gives us finer control over monomer generation