Generative models for graph-based protein design

John Ingraham, Vikas K. Garg, Regina Barzilay, Tommi Jaakkola
Computer Science and Artificial Intelligence Lab, MIT
{ingraham, vgarg, regina, tommi}@csail.mit.edu

Presenter: Yukang Yang

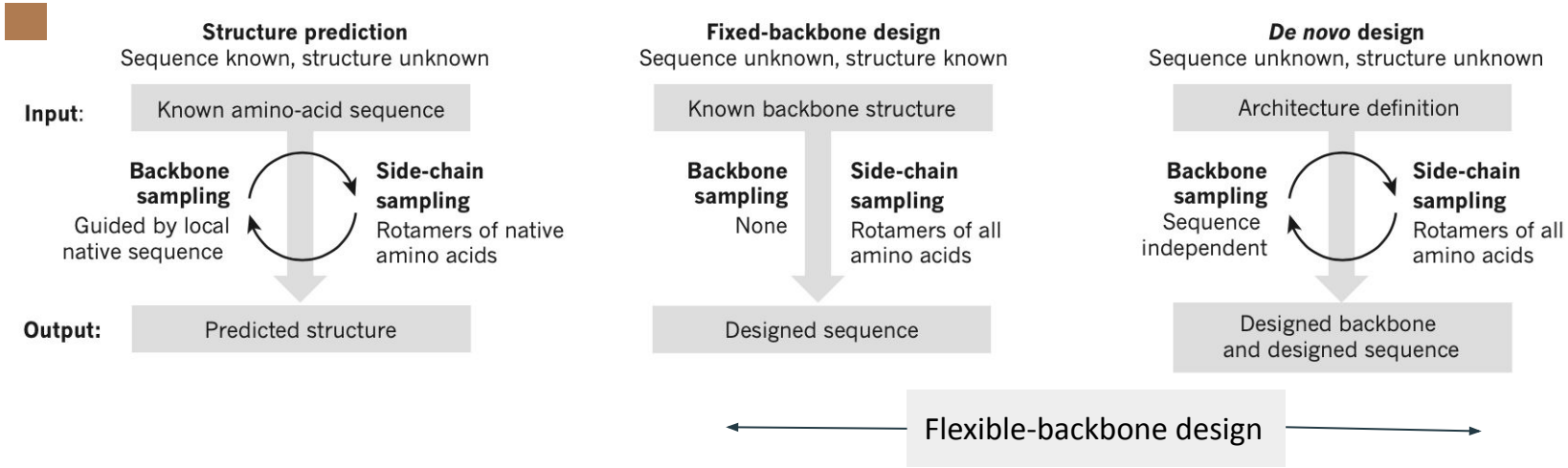
Paper: https://openreview.net/pdf?id=ByMEAHrgLB
Codes: https://github.com/jingraham/neurips19-graph-protein-design

Introduction

- Computational Protein Design (i.e., *Inverse protein folding problem*)

$$\mathcal{F}_\theta : \mathcal{X} \mapsto \mathcal{S}$$ the amino acids sequence $S = \{s_i : 1 \le i \le n\}$

  - Desired **structure** $X = \{x_i \in R^3 : 1 \le i \le n\}$
  - Desired functional properties



- Related Work

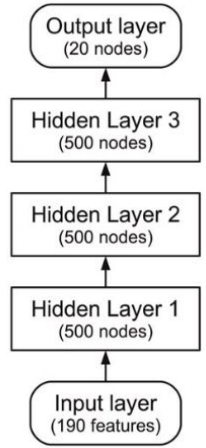
Bottom-up: optimizing Energy Function

Rosetta

Top-down: *Conditional* Generative Model

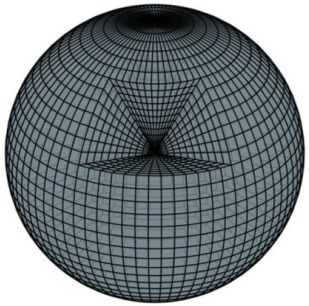
MLP-based

SPIN2



- Cons: Feature Design

CNN-based



Cons:

- Slow inference
- Complex preprocessing

Graph-based

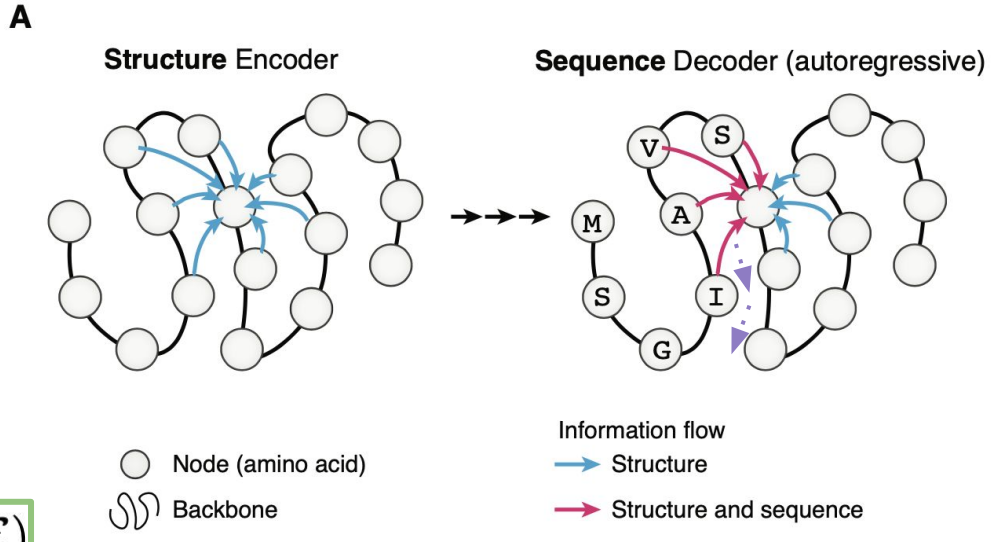
Representing protein structure as a graph

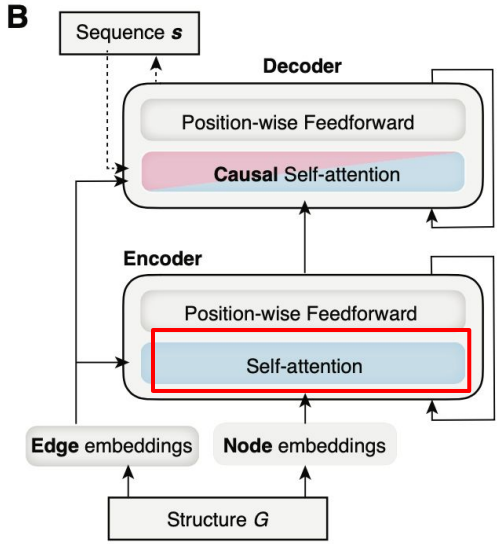
Pros:

- Computational efficiency
- Inductive bias
- Representational flexibility

Method

Structured Transformer





$$\mathcal{G} = (\mathcal{V}, \mathcal{E})$$

$$\mathcal{V} = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_N\}$$ Node Features

$$\mathcal{E} = \{\boldsymbol{e}_{ij}\}_{i \neq j}$$ Edge Features

$$p(\boldsymbol{s}|\boldsymbol{x}) = \prod_i p(s_i|\boldsymbol{x}, \boldsymbol{s}_{<i})$$

Inductive Bias:

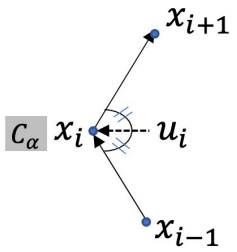
- Invariance
- Locally Informative

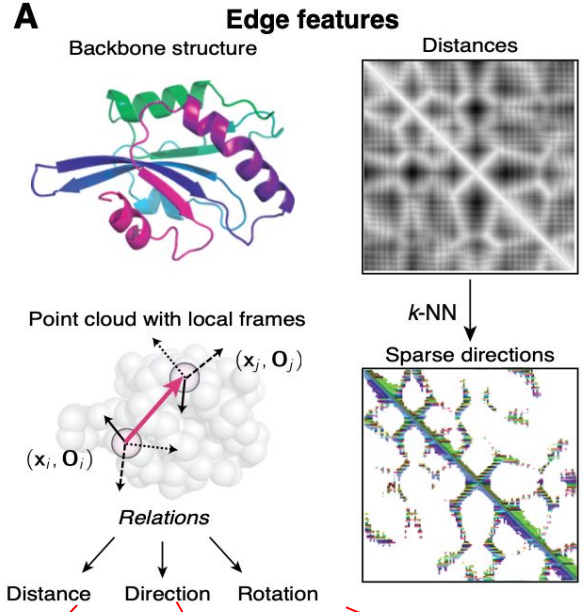
Relative Spatial Encodings

Local Frame

$$\boldsymbol{O}_i = [\boldsymbol{b}_i \;\; \boldsymbol{n}_i \;\; \boldsymbol{b}_i \times \boldsymbol{n}_i]$$

$$\boldsymbol{u}_i = \frac{\boldsymbol{x}_i - \boldsymbol{x}_{i-1}}{||\boldsymbol{x}_i - \boldsymbol{x}_{i-1}||}, \quad \boldsymbol{b}_i = \frac{\boldsymbol{u}_i - \boldsymbol{u}_{i+1}}{||\boldsymbol{u}_i - \boldsymbol{u}_{i+1}||}, \quad \boldsymbol{n}_i = \frac{\boldsymbol{u}_i \times \boldsymbol{u}_{i+1}}{||\boldsymbol{u}_i \times \boldsymbol{u}_{i+1}||}$$





k-NN (**sparsity**)

$\{\boldsymbol{e}_{ij}\}_{j \in \mathbf{N}(i,k)}$ (k=30)

long-range dependencies in sequence

⇕

short-range in 3D space

Edge Features

Structural Encoding

$$\boldsymbol{e}_{ij}^{(s)} = \left(\mathbf{r}\left(||\boldsymbol{x}_j - \boldsymbol{x}_i||\right), \quad \boldsymbol{O}_i^T \frac{\boldsymbol{x}_j - \boldsymbol{x}_i}{||\boldsymbol{x}_j - \boldsymbol{x}_i||}, \quad \mathbf{q}\left(\boldsymbol{O}_i^T \boldsymbol{O}_j\right) \right)$$

$\boldsymbol{e}_{ij}^{(p)}$

Positional Encoding

Attention:

- Encoder

$$a_{ij}^{(\ell)} = \frac{\exp(m_{ij}^{(\ell)})}{\sum_{j' \in \mathrm{N}(i,k)} \exp(m_{ij'}^{(\ell)})} \qquad m_{ij}^{(\ell)} = \frac{{q_i^{(\ell)}}^\top z_{ij}^{(\ell)}}{\sqrt{d}}$$

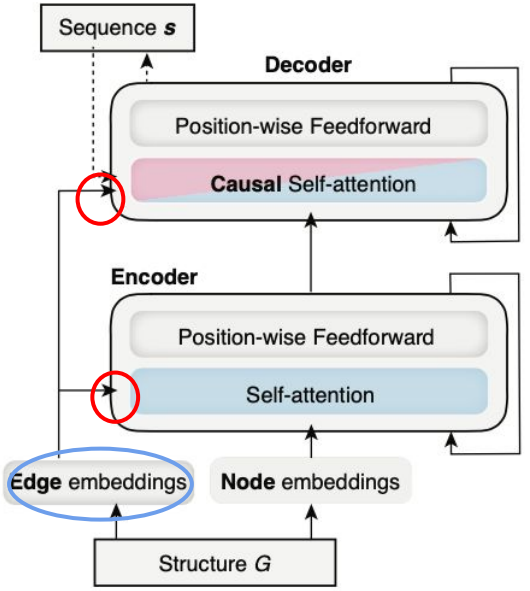
$$\boxed{h_i^{(\ell)} = \sum_{j \in N(i,k)} a_{ij}^{(\ell)} v_{ij}^{(\ell)}}$$

Key & Value

$$\boxed{r_{ij} = (h_j, e_{ij})}$$

$$v_{ij}^{(l)} = W_v^{(l)} r_{ij}$$

$$z_{ij}^{(l)} = W_z^{(l)} r_{ij}$$



- Decoder

Causally consistent

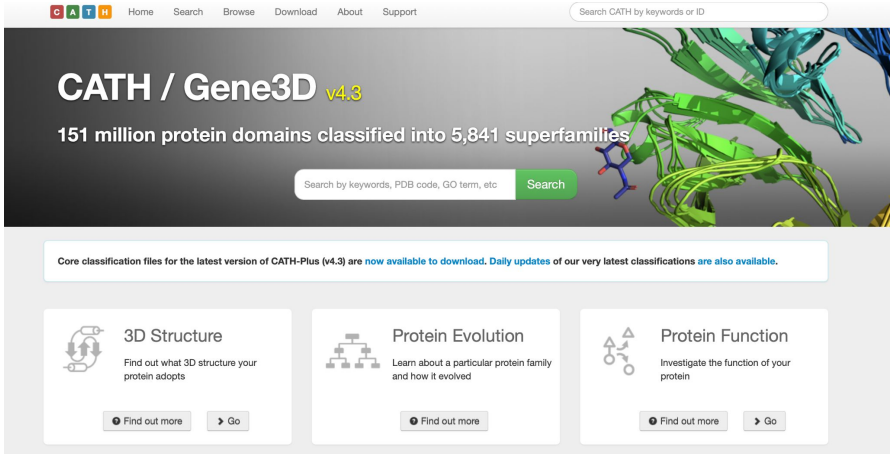
$$r_{ij}^{(\mathrm{dec})} = \begin{cases} (h_j^{(\mathrm{dec})}, e_{ij}, \mathbf{g}(s_j)) & i > j \\ (h_j^{(\mathrm{enc})}, e_{ij}, \mathbf{0}) & i \leq j \end{cases}$$

Results

- Dataset: **CATH** 4.2

a **hierarchical** domain classification of the three-dimensional (3D) structures of proteins

- **C**lass: secondary structure content
- **A**rchitecture: shape revealed by the orientations of the secondary structure units
- **T**opology: sequential connectivity of secondary structure elements
- **H**omologous superfamily: whether the domains are evolutionarily related



Structure-split setting

- Training set: 18024 chains
- Validation set: 608 chains
- Test set: 1120 chains

No CAT overlap

http://www.cathdb.info

- Evaluation

Single mutations may cause a protein to break or misfold

Sequence  3D structure

Many protein sequences may design the **same** 3D structure

Sequence Similarity x

- ❖ Likelihood-based: Perplexity
- ❖ Native sequence recovery
- ❖ Experimental comparison

❖ Likelihood-based: Perplexity

Table 1: **Null perplexities** for common statistical models of proteins.

Null model	Perplexity	Conditioned on
Uniform	20.00	-
Natural frequencies	17.83	Random position in a natural protein
Pfam HMM profiles	11.64	Specific position in a specific protein family

Perplexity $\propto$ 1/probability

$$\log(perplexity(S)) = -\frac{1}{m}\sum_{i=1}^{m} \log p(w_i|w_1, \cdots, w_{i-1})$$

Table 2: **Per-residue perplexities for protein language modeling** (lower is better). The protein chains have been cluster-split by CATH topology, such that test includes only unseen 3D folds. While a structure-conditioned language model can generalize in this structure-split setting, unconditional language models struggle.

Test set	Short	Single chain	All
Structure-conditioned models			
Structured Transformer (ours)	**8.54**	**9.03**	**6.85**
SPIN2 [8]	12.11	12.61	-
Language models			
LSTM ($h = 128$)	16.06	16.38	17.13
LSTM ($h = 256$)	16.08	16.37	17.12
LSTM ($h = 512$)	15.98	16.38	17.13
Test set size	94	103	1120

$$Perplexity = e^{\frac{\sum_i^m L_{nll}}{m}}$$

Protein profiles

Ablations:

ProteinMPNN

Message Passing Neural Networks (MPNN)

$$\Delta h_i = \sum_j \mathrm{MLP}(h_i, h_j, e_{ij})$$

Node features	Edge features	Aggregation	Short	Single chain	All
Rigid backbone					
Dihedrals	Distances, Orientations	Attention	8.54	9.03	6.85
Dihedrals	Distances, Orientations	PairMLP	**8.33**	**8.86**	**6.55**
C_α angles	Distances, Orientations	Attention	9.16	9.37	7.83
Dihedrals	Distances	Attention	9.11	9.63	7.87
Flexible backbone					
C_α angles	Contacts, Hydrogen bonds	Attention	11.71	11.81	11.51
SPIN2 [8]			12.11	12.61	-

❖ Native sequence recovery

Method	Recovery (%)	Speed (AA/s) CPU	Speed (AA/s) GPU
Rosetta 3.10 `fixbb`	17.9	4.88×10^{-1}	N/A
Ours ($T = 0.1$)	**27.6**	$\mathbf{2.22 \times 10^{2}}$	$\mathbf{1.04 \times 10^{4}}$

(a) Single chain test set (103 proteins)

Method	Recovery (%)
Rosetta, `fixbb` 1	33.1
Rosetta, `fixbb` 2	38.4
Ours ($T = 0.1$)	**39.2**

(b) Ollikainen benchmark (40 proteins)

Table 4: **Improved reliability and speed compared to Rosetta.** (a) On the 'single chain' test set, our model more accurately recovers native sequences than Rosetta `fixbb` with greater speed (CPU: single core of Intel Xeon Gold 5115, GPU: NVIDIA RTX 2080). This set includes NMR-based structures for which Rosetta is known to not be robust [46]. (b) Our model also performs favorably on a prior benchmark of 40 proteins. All results reported as median of average over 100 designs.

- More Accurate
- Faster

$$p(\boldsymbol{s}|\boldsymbol{x}) = \prod_i p(s_i|\boldsymbol{x}, \boldsymbol{s}_{<i}) \quad \Longrightarrow \quad p^{(T)}(\mathbf{s}|\boldsymbol{x}) = \prod_i \frac{p(s_i|\boldsymbol{x}, \boldsymbol{s}_{<i})^{1/T}}{\sum_a p(a|\boldsymbol{x}, \boldsymbol{s}_{<i})^{1/T}}$$

Biased sampling

Rosetta : state-of-the-art framework for computational protein design

❖ Experimental comparison

Mutation effects

Table 5: **Structure-conditioned likelihoods correlate with mutation effects in *de novo*-designed miniproteins**. Shown are Pearson correlation coefficients (R, higher is better) between the log-likelihoods of mutated sequences and high-throughput mutation effect data from a systematic design of miniproteins [6]. Each design (column) includes 775 experimentally tested mutant protein sequences.

Design	$\beta\beta\alpha\beta\beta_{37}$	$\beta\beta\alpha\beta\beta_{1498}$	$\beta\beta\alpha\beta\beta_{1702}$	$\beta\beta\alpha\beta\beta_{1716}$	$\alpha\beta\beta\alpha_{779}$
Rigid backbone	0.47	0.45	0.12	0.47	0.57
Flexible backbone	0.50	0.44	0.17	0.40	0.56

Design	$\alpha\beta\beta\alpha_{223}$	$\alpha\beta\beta\alpha_{726}$	$\alpha\beta\beta\alpha_{872}$	$\alpha\alpha\alpha_{134}$	$\alpha\alpha\alpha_{138}$
Rigid backbone	0.36	0.11	0.21	0.24	0.33
Flexible backbone	0.33	0.21	0.23	0.36	0.41

log-likelihoods of mutated sequences

Pearson correlation coefficients

high-throughput mutation effect data

Take-away

Structured Transformer

- **Graph-based** Transformer
- + Inductive Bias: 3D **structural** encodings, spatial **locality**
- Improved **perplexities**
- Compared to the SOTA protein design program, more **accurate and faster**

*Showing the potential of being able to **efficiently** design and engineer protein sequence with **structurally-guided** generative models…*

Q & A