Protein Structure Tokenizer for Efficient Learning

Hyunkyu Jung[134] Luiz Felipe Vecchietti[3] Meeyoung Cha[13] Ho Min Kim[24]

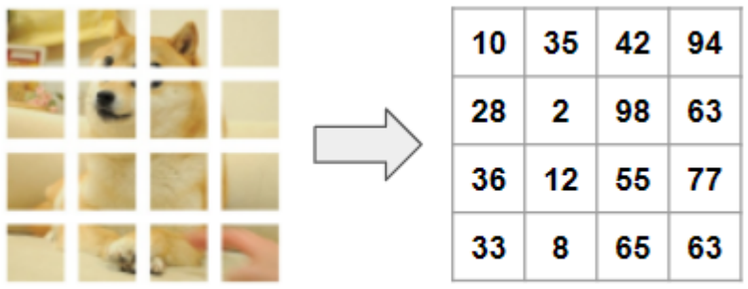
[1]KAIST, School of Computing [2]KAIST, Graduate School of Medical Science and Engineering
[3]IBS, Data Science Group [4]IBS, Protein Communication Group

{dino8egg}@kaist.ac.kr, {lfelipesv, mcha, kimhm}@ibs.re.kr

Introduction

High-dimensional data learning:
- Requires Large Memory
- Requires Long Training Time

Data Tokenization (Data Quantization):
A method that converts high-dimensional data into low-dimensional data.



10	35	42	94
28	2	98	63
36	12	55	77
33	8	65	63

Image Quantization Example

Here, we propose a **Protein Structure Tokenizer** which enables efficient protein data representation for fast learning with lower memory use.

Data

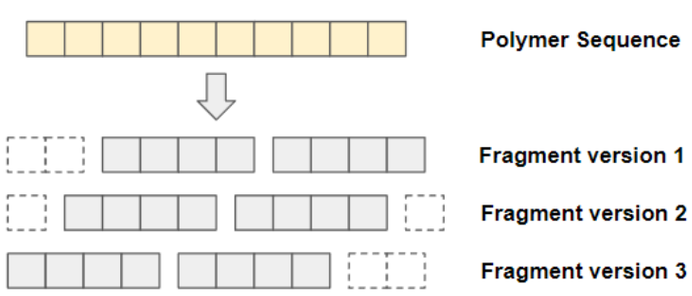
Query on RCSB website (PDB)
- No DNA and RNA structure on pdb file.
- 16 ≤ (# of polymer residues per deposited model) ≤ 512
- 16 ≤ (# of polymer residues per Assembly file) ≤ 512
- 16 ≤ (Polymer Entity Sequence) ≤ 512

Remove Assembly Files with given condition
- Assembly files w.o. Ca, C, N, O atoms.
- Assembly files contains amino acid other than basic 20 amino acids.
- Assembly file which is too large. (≥ 4096 Bytes)

→ 65,000 training set, 7,443 testing set

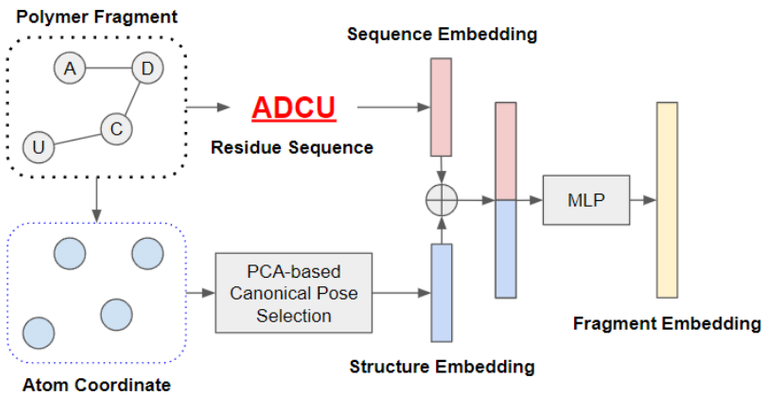
Methods

1. Polymer Fragment Generation
We split the polymer sequence to generate the **polymer fragments** which sequence leng th set to 4 or 8. For robust representation, we consider all splitting cases.

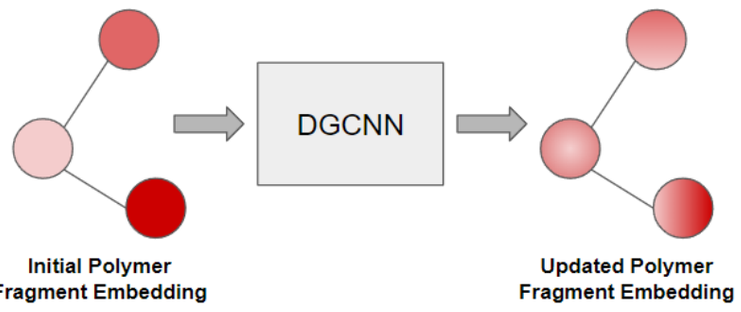


Methods

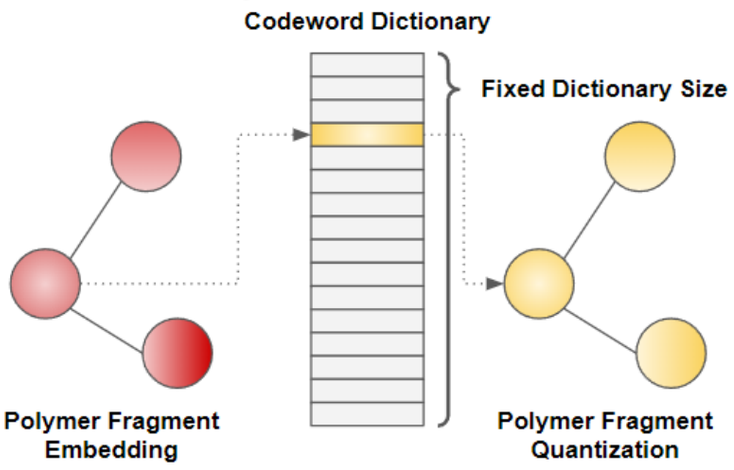
2. Intra-Fragment Embedding
We compute an **intra-fragment embedding** by considering both residue sequence and backbone coordinate of a given polymer fragment.



3. Inter-Fragment Embedding
We update the polymer fragment embedding by adopting neighbor fragment information using DGCNN network.

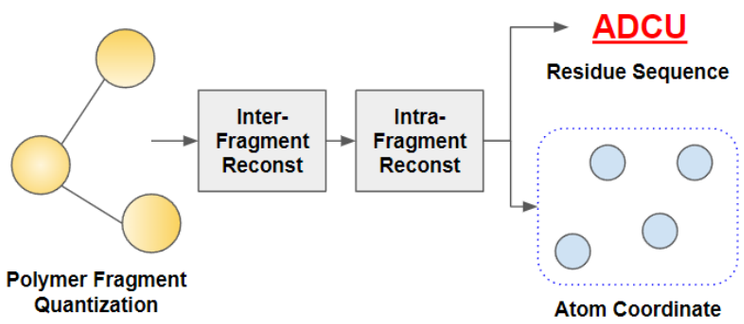


4. Codeword Quantization



We tokenize(quantize) a polymer fragment by matching each embedding into fixed a number of cases. We set the dictionary size to 8192.

5. Polymer Fragment Reconstruction
For self-supervised learning, we reconstruct the polymer residue and atom coordinate.

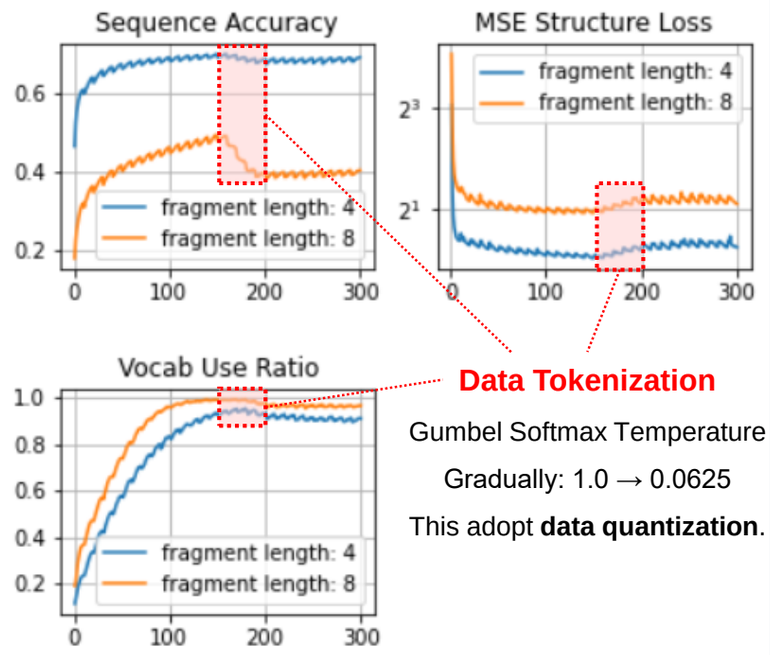


Experiment

Loss Function
S: Residue Sequence, P: Atom Coordinate

$$\mathcal{L}_{Total} = \mathcal{L}_{type} + \lambda_1 \mathcal{L}_{structure} + \lambda_2 \mathcal{L}_{KL}$$
$$\mathcal{L}_{type} = \mathcal{L}_{CE}(\mathbf{S}_j, \hat{\mathbf{S}}_j)$$
$$\mathcal{L}_{structure} = \mathcal{L}_{MSE}(\mathbf{P}_j, \hat{\mathbf{P}}_j)$$
$$\mathcal{L}_{KL} = D_{\mathrm{KL}}\left[\mathcal{Q}_\phi(\mathbf{z}_j \mid \mathbf{S}_j, \mathbf{P}_j), \mathcal{P}_\theta(\mathbf{z}_j \mid \tilde{\mathbf{S}}_j, \tilde{\mathbf{P}}_j)\right]$$

Result & Discussion



Data Tokenization
Gumbel Softmax Temperature
Gradually: 1.0 → 0.0625
This adopt **data quantization**.

Polymer Fragment Training Result

- Both tokenizer use **over 90%** of the Codeword dictionary well.
- The $Accuracy^{length=4}_{Reconstruct}$ is about **0.7** even **20^4 > 8192**.
- The $Accuracy^{length=8}_{Reconstruct}$ is about **0.4** even **20^8 >> 8192**.
- The MSE Loss of atom coordinate reconstruction **decreased** well.

We introduce **Protein Structure Tokenizer**
- Make **Efficient Representation**
- Enable **Fast Learning**
- **Lower Memory** Use
- Similar role as vision tokenizer on ViT

Reference

[1] Yu, Xumin, et al. "Point-bert: Pre-training 3d point cloud transformers with masked point modeling." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022.
[2] Ramesh, Aditya, et al. "Zero-shot text-to-image generation." International Conference on Machine Learning. PMLR, 2021.
[3] Dosovitskiy, Alexey, et al. "An image is worth 16x16 words: Transformers for image recognition at scale." arXiv preprint arXiv:2010.11929 (2020).
[4] Jang, Eric, Shixiang Gu, and Ben Poole. "Categorical reparameterization with gumbel-softmax." arXiv preprint arXiv:1611.01144 (2016).
[5] Wang, Yue, et al. "Dynamic graph cnn for learning on point clouds." Acm Transactions On Graphics (tog) 38.5 (2019): 1-12.
[6] Li, Feiran, et al. "A closer look at rotation-invariant deep point cloud analysis." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021.

This work was supported by the Institute for Basic Science in South Korea [IBS-R029-C2, IBS-R030-C1].