Clustering for Protein Representation Learning

Ruijie Quan, Wenguan Wang, Fan Ma,
Hehe Fan, Yi Yang

https://openaccess.thecvf.com/content/CVPR2024/html/Quan_Clustering_for_Protein_Representation_Learning_CVPR_2024_paper.html

Burak Suyunu

LifeLU Reading Group

31.10.2024

Motivation

Background on Protein Representation:

- Protein structures, functions, and classifications are defined by **critical amino acids** and **spatial configurations**.
- Traditional models often **treat amino acids equally**, failing to capture the nuanced role of certain amino acids crucial to protein functionality.

Research Motivation:

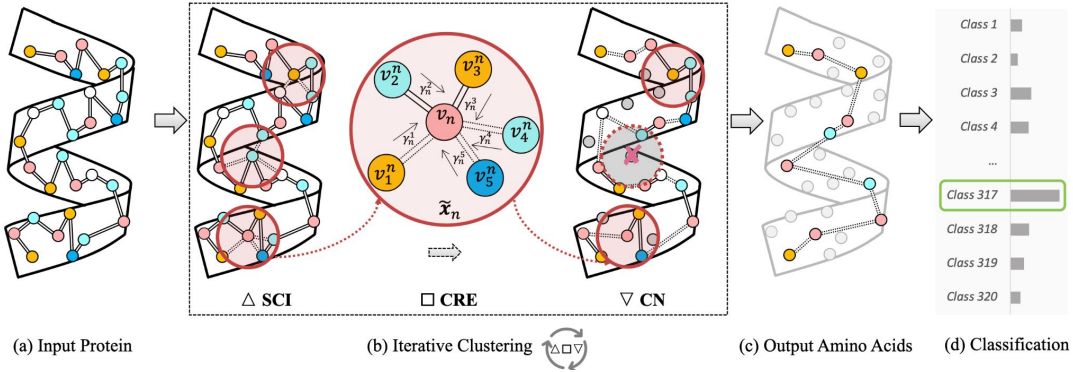
- **Critical amino acids** impact protein folding and functionality **disproportionately**.
- **Example:** A single amino acid change can drastically alter protein function, as in the case of **sickle cell anemia**.
    - Sickle cell anemia results from a single amino acid change in hemoglobin, causing it to form abnormal fibers that distort red blood cell shape.

Objective: To create a **protein representation learning model** that identifies and prioritizes **critical amino acids** using **clustering** techniques for improved predictive performance in tasks like fold classification and enzyme reaction classification.

Methodology - Overview of Neural Clustering Framework

Framework Concept:

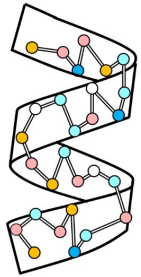
- Treats each protein as a graph, where nodes represent amino acids and edges show spatial or sequential connections.
- Clustering framework identifies critical components iteratively for a hierarchical, informative protein representation.



Methodology - Overview of Neural Clustering Framework

Notation and Input:

- A protein is represented as a graph $P = (V, E, Y)$, where:
  - $V = \{v_1, \ldots, v_N\}$ represents the nodes (amino acids),
  - E represents edges denoting spatial/sequential connections,
  - Y is the set of labels (classification targets).



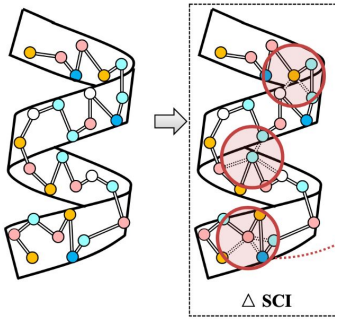
(a) Input Protein

- $\{x_1, \cdots, x_N\}$ to denote the features of V, where $x_n \in \mathbb{R}^{256}$ is the feature vector of amino acid v_n.
  - one-hot encoding of amino acid types, the orientations, sequential and spatial positions of amino acids.
- A to denote the adjacency matrix of V.

Methodology - Overview of Neural Clustering Framework

Pipeline Steps:

1. **Spherical Cluster Initialization (SCI)**: Clusters are initialized based on amino acid neighbors within a fixed spatial radius.

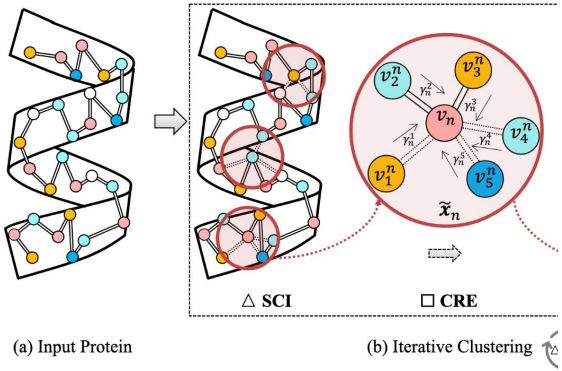


(a) Input Protein

Methodology - Overview of Neural Clustering Framework

Pipeline Steps:

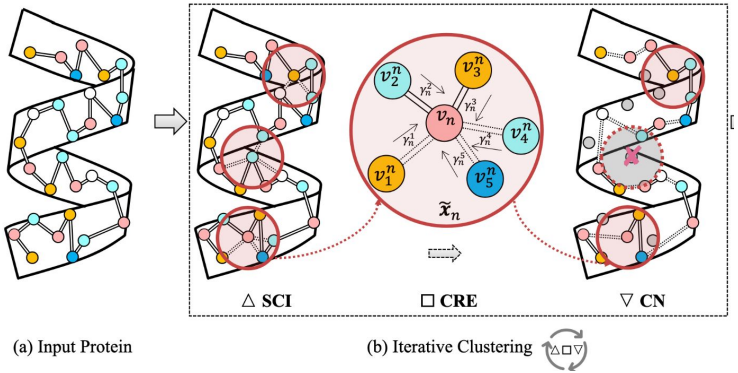
1. **Spherical Cluster Initialization (SCI)**: Clusters are initialized based on amino acid neighbors within a fixed spatial radius.
2. **Cluster Representation Extraction (CRE)**: For each cluster, features of neighboring nodes are aggregated into a single representative vector.



Methodology - Overview of Neural Clustering Framework

Pipeline Steps:

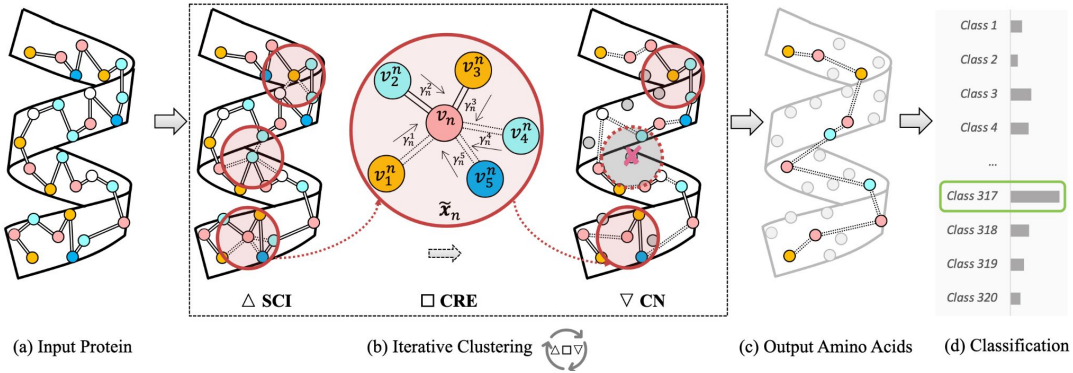
1. **Spherical Cluster Initialization (SCI)**: Clusters are initialized based on amino acid neighbors within a fixed spatial radius.
2. **Cluster Representation Extraction (CRE)**: For each cluster, features of neighboring nodes are aggregated into a single representative vector.
3. **Cluster Nomination (CN)**: A Graph Convolutional Network (GCN) scores clusters; top-scoring clusters advance to the next iteration, capturing critical amino acids progressively.



Methodology - Overview of Neural Clustering Framework

Pipeline Steps:

1. **Spherical Cluster Initialization (SCI)**: Clusters are initialized based on amino acid neighbors within a fixed spatial radius.
2. **Cluster Representation Extraction (CRE)**: For each cluster, features of neighboring nodes are aggregated into a single representative vector.
3. **Cluster Nomination (CN)**: A Graph Convolutional Network (GCN) scores clusters; top-scoring clusters advance to the next iteration, capturing critical amino acids progressively.



(a) Input Protein (b) Iterative Clustering (c) Output Amino Acids (d) Classification

Methodology - Spherical Cluster Initialization (SCI)

Objective: Form initial clusters by grouping amino acids that are spatially or sequentially close, allowing the model to focus on local neighborhoods, capturing the 3D arrangement of amino acids crucial for functionality.

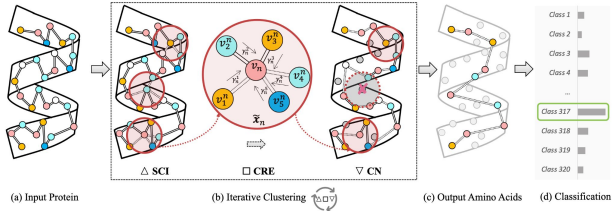
Cluster Formation:

- For each amino acid v_n, initialize a cluster H_n consisting of neighboring nodes within a fixed radius r, where v_n is regarded as the **medoid** node of the cluster

$$H_n = \{v_n^1, \ldots, v_n^K \mid d(v_n, v_n^k) \leq r\}$$

  where $d(v_n, v_n^k)$ represents the distance between v_n and v_n^k, and K is the number of neighboring nodes within r.

- This radius r is a hyperparameter, adjusted across iterations to balance spatial context capture and noise.
- In subsequent iterations *(t > 1)*, we use the nominated N_{t-1} amino acids from the previous **t-1-th** iteration to initialize the clusters.
- The adjacency matrix A is regenerated in each SCI process with considering the connectivity among amino acids.



Methodology - Cluster Representation Extraction (CRE)

Objective: Summarize each cluster by generating a representative feature vector that captures both primary (1D) and tertiary (3D) structural information, with attention focusing on the most relevant amino acids within the cluster.

Feature Calculation:

- For each amino acid v_n^k in the cluster H_n, calculate its feature vector x_{n_k}:

$$x_{n_k} = f(g_{n_k}, o_{n_k}, d_{n_k}, s_k, e_k)$$

where:

- $g_{n_k} = (z_k - z_n)$: relative spatial coordinates,
- o_{n_k}: 3D orientation vector,
- $d_{n_k} = ||z_k - z_n||$: spatial distance,
- s_k: sequential position,
- e_k: amino acid encoding (e.g., one-hot).

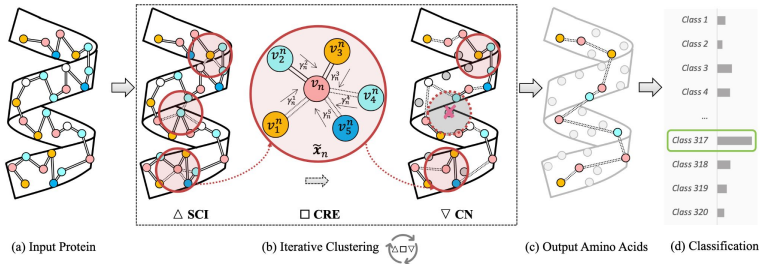
Cluster Attention Mechanism:

- Attention score α_{n_k} between medoid x_n and each x_{n_k} in H_n using a learnable vector w:

$$\alpha_{n_k} = \frac{\exp(w \cdot [x_n, x_{n_k}])}{\sum_{j=1}^{K} \exp(w \cdot [x_n, x_{n_j}])}$$

- Weighted sum to generate cluster representation $\tilde{x}_n$:

$$\tilde{x}_n = \sum_{k=1}^{K} \alpha_{n_k} x_{n_k}$$



Methodology - Cluster Nomination (CN)

Objective: Select the most representative clusters to focus on critical amino acids while iteratively refining the protein representation.

Scoring with Graph Convolutional Network (GCN):

- For each cluster representation $\tilde{x}_n$, a nomination score ϕ_n is computed using GCN:

$$\phi_n = \sigma(W_1\tilde{x}_n + \sum_{m=1}^{N_t} A_{n,m}(W_2\tilde{x}_n - W_3\tilde{x}_m))$$

where:

- W_1, W_2, W_3 are learnable parameters,
- σ is the ReLU activation function,
- $A_{n,m}$ is the adjacency matrix indicating edges between clusters.

Selection of Top Clusters:

- Clusters are ranked by ϕ_n, and the top N_t clusters are selected for the next iteration:

$$N_t = \lceil \omega \cdot N_{t-1} \rceil$$

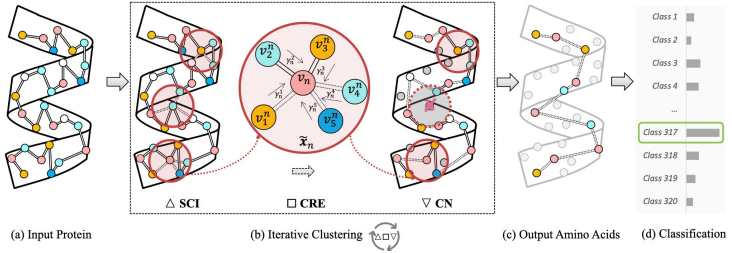
where ω (nomination fraction) controls the number of clusters passed to the next stage.

- **Weighted Cluster Representation**:
  - The calculated nomination scores ϕ_n are used to weight the cluster features:

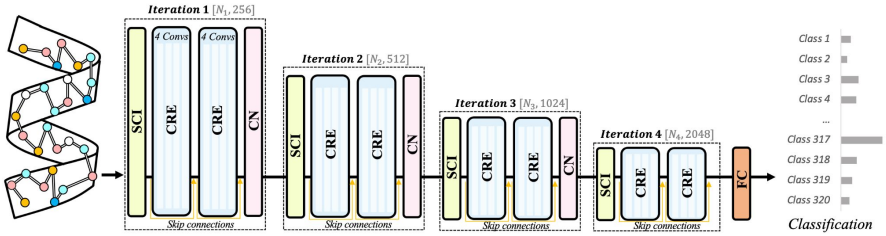
$$\hat{X}_c = \Phi \cdot X_c$$

where:

- $\Phi = [\phi_1, \phi_2, \ldots, \phi_{N_t}]^T$ (vector of scores),
- $X_c = [\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_{N_t}]$ (matrix of cluster representations),
- $\cdot$ denotes the Hadamard (element-wise) product.



Iterative Clustering Process



Process:

- **Input**: Start with all amino acids in the first iteration ***(N=N_0)***.
- **Iterations**: Repeat SCI, CRE, and CN for T iterations, with each iteration focusing on increasingly specific amino acid subsets.
- **Final Representation**: After ***T*** iterations, the remaining ***N_T*** clusters represent a condensed, informative protein structure suitable for classification tasks.
- **Classification**: Project the final amino acid features into a Y-dimensional vector with a fully connected layer, followed by sigmoid and binary cross-entropy loss (for single-label) or softmax and cross-entropy loss (for multi-label) to predict protein properties.

Implementation:

- **Set T = 4**, suggesting the neural clustering framework consists of four iterations.
- **Stack 2 CRE** blocks to learn the representation of the selected N_t amino acids
- **Increase** the value of the **cluster radius r** applied in SCI step as the number of iterations increases ⇒ **r, 2r, 3r, and 4r**, respectively.
- **Set cluster nomination fraction ω = 40%** for the CN step which determines the proportion of clusters selected as the medoid nodes for the next iteration, suggesting a suitable trade-off between preserving information and reducing redundancy and complexity.
- **Adopt skip connection** per CRE block to facilitate information flow and ease network training and **rotation invariance**.

Experiments

- **Enzyme Commission (EC) number prediction** seeks to anticipate the EC numbers of diverse proteins that elucidate their role in catalyzing biochemical reactions.
    - Multi-label classification task evaluated by the protein-centric maximum F-score: Fmax.
- **GO term prediction** aims to forecast whether a protein belongs to certain GO terms. Three sub-tasks:
    - Molecular function (MF) term prediction consisting of 489 classes,
    - Biological process (BP) term prediction including 1943 classes,
    - Cellular component (CC) term prediction with 320 classes.
    - Multi-label classification task evaluated by the protein-centric maximum F-score: Fmax.
- **Protein fold classification** aims to predict the fold class label of a protein. It contains three different evaluation scenarios:
    - Fold, where proteins belonging to the same superfamily are excluded during training,
    - Superfamily, where proteins from the same family are not included during training
    - Family, where proteins from the same family are used during training.
    - Single-label classification task evaluated by mean accuracy.
- **Enzyme reaction classification** endeavors to predict the enzyme-catalyzed reaction class of a protein, utilizing all four levels of the EC number to portray reaction class.
    - Single-label classification task evaluated by mean accuracy.

Experiments

Table 1. F_{max} on EC and GO prediction and Accuracy (%) on fold and reaction classification. [†] denotes results taken from [85] and [*] denotes results taken from [38] and [36] (§4.1-§4.4).

Method	Publication	EC	GO			Fold Classification				Reaction
			BP	MF	CC	Fold	Super.	Fam.	Avg.	
ResNet [66]	*NeurIPS 2019*	0.605	0.280	0.405	0.304	10.1	7.21	23.5	13.6	24.1
LSTM [66]	*NeurIPS 2019*	0.425	0.225	0.321	0.283	6.41	4.33	18.1	9.61	11.0
Transformer [66]	*NeurIPS 2019*	0.238	0.264	0.211	0.405	9.22	8.81	40.4	19.4	26.6
GCN [46]	*ICLR 2017*	0.320	0.252	0.195	0.329	16.8*	21.3*	82.8*	40.3*	67.3*
GAT [79]	*ICLR 2018*	0.368	0.284†	0.317†	0.385†	12.4	16.5	72.7	33.8	55.6
GVP [44]	*ICLR 2021*	0.489	0.326†	0.426†	0.420†	16.0	22.5	83.8	40.7	65.5
3DCNN [15]	*Bioinform 2018*	0.077	0.240	0.147	0.305	31.6*	45.4*	92.5*	56.5*	72.2*
GraphQA [7]	*Bioinform 2021*	0.509	0.308	0.329	0.413	23.7*	32.5*	84.4*	46.9*	60.8*
New IEConv [36]	*ICLR 2022*	0.735	0.374	0.544	0.444	47.6*	70.2*	99.2*	72.3*	87.2*
GearNet [94]	*ICLR 2023*	0.810	0.400	0.581	0.430	48.3	70.3	99.5	72.7	85.3
ProNet [81]	*ICLR 2023*	-	-	-	-	52.7	70.3	99.3	74.1	86.4
CDConv [22]	*ICLR 2023*	0.820	0.453	0.654	0.479	56.7	77.7	99.6	78.0	88.5
Ours	-	**0.866**	**0.474**	**0.675**	**0.483**	**63.1**	**81.2**	99.6	**81.3**	**89.6**

Diagnose Analysis

Table 2. Ablative experiments for the neural clustering algorithm. (a) An off-the-shelf clustering algorithm; (b) A simple average pooling method; (c) Randomly generate attention score γ_k^n. See §4.5 for details.

Method	EC	GO			Fold Classification				Reaction
		BP	MF	CC	Fold	Super.	Fam.	Avg.	
(a)	0.792	0.385	0.579	0.429	43.1	67.1	99.1	69.8	86.8
(b)	0.817	0.452	0.641	0.453	57.2	78.7	99.3	78.4	88.1
(c)	0.765	0.342	0.567	0.415	44.6	69.5	99.2	71.1	86.4
Ours	**0.866**	**0.474**	**0.675**	**0.483**	**63.1**	**81.2**	**99.6**	**81.3**	**89.6**

Table 4. Analysis of a different number of iterations. See details in §4.5.

T	EC	GO			Fold Classification				Reaction
		BP	MF	CC	Fold	Super.	Fam.	Avg.	
1	0.717	0.402	0.593	0.432	55.7	73.2	97.4	75.4	84.7
2	0.824	0.438	0.642	0.453	60.0	79.2	98.0	79.1	88.1
3	0.855	0.469	0.677	0.480	62.2	80.8	99.3	80.8	89.0
4	**0.866**	**0.474**	**0.675**	**0.483**	**63.1**	**81.2**	**99.6**	**81.3**	**89.6**
5	0.809	0.423	0.605	0.455	58.1	75.7	98.5	77.4	86.3

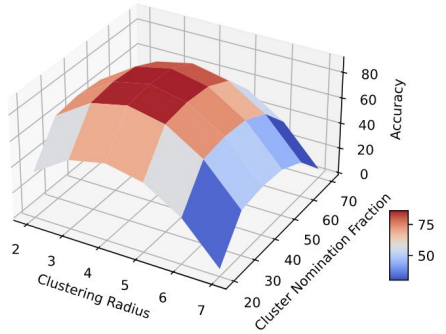


Figure 3. Performance change curve with different combinations of ω and r for enzyme reaction classification. See §4.5 for details.

Diagnose Analysis

Table 3. Efficiency comparison to SOTA competitors on enzyme reaction classification. See §4.5 for details

Method	Acc.	Runing Time
New IEConv [36]	87.2%	75.3 ms
GearNet [94]	85.3%	*OOM*
ProNet [81]	86.4%	27.5 ms
CDConV [22]	88.5%	10.5 ms
Ours	89.6%	10.9 ms

Table 5. $u\%$ missing coordinates (§4.5).

$u\%$	Fold	Super.	Fam.
0%	**63.1**	**81.2**	**99.6**
5%	61.9	79.8	99.5
10%	60.1	78.7	99.5
20%	56.7	76.9	99.3
30%	50.2	73.6	99.2
40%	47.8	71.3	99.0

Table 6. Comparison results with existing protein language models. See details in §4.5.

Method	Pretraining Dataset	EC	GO			Fold Classification				Reaction
			BP	MF	CC	Fold	Super.	Fam.	Avg.	
DeepFRI [62]	Pfam 10M	0.631	0.399	0.465	0.460	15.3	20.6	73.2	36.4	63.3
ESM-1b [68]	UniRef50 24M	0.864	0.470	0.657	0.488	26.8	60.1	97.8	61.6	83.1
ProtBERT-BFD [20]	BFD 2.1B	0.838	0.279	0.456	0.408	26.6	55.8	97.6	60.0	72.2
IEConv (amino level) [37]	PDB 476K	-	0.468	0.661	0.516	50.3	80.6	99.7	76.9	88.1
LM-GVP [85]	UniRef100 0.21B	0.664	0.417	0.545	0.527	-	-	-	-	-
GearNet-Edge-IEConv [94]	AlphaFoldDB 805K	0.874	0.490	0.654	0.488	54.1	80.5	99.9	78.2	87.5
Ours	-	0.866	0.474	**0.675**	0.483	**63.1**	**81.2**	99.6	**81.3**	**89.6**

Visualization

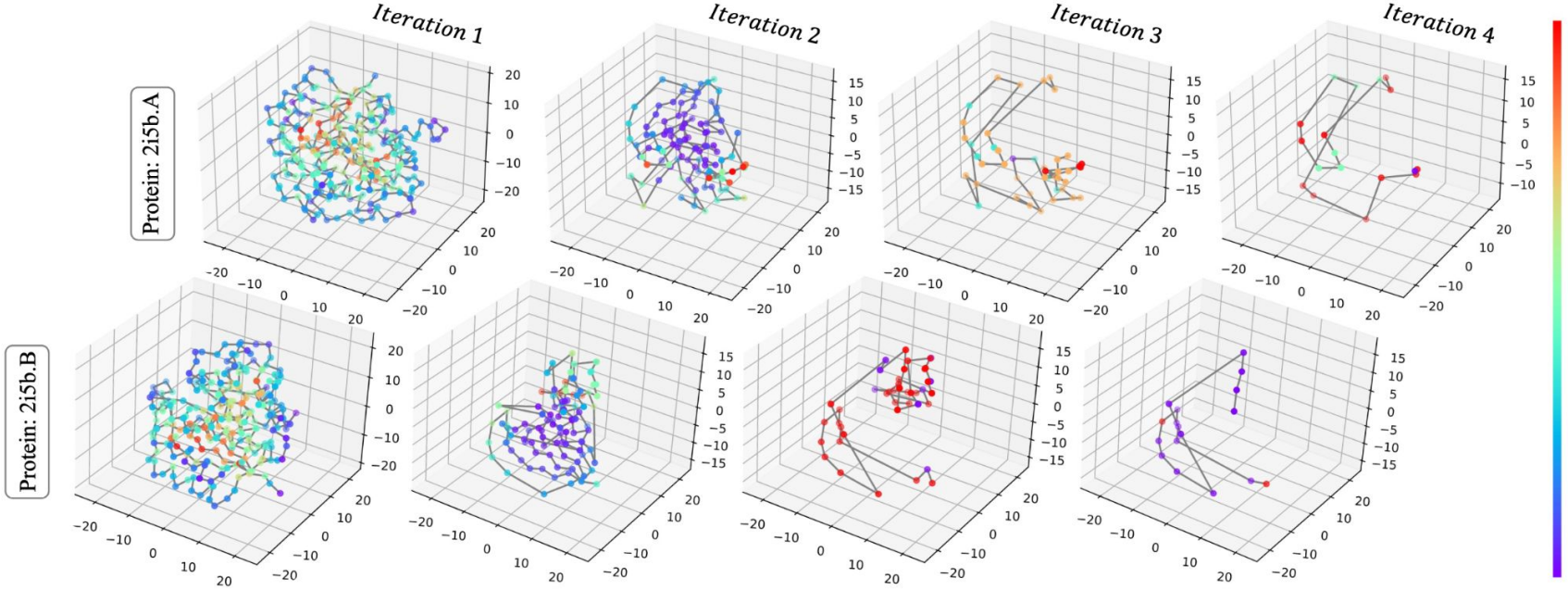


Figure 4. Visualization results of the protein structure at each iteration. The color of the node denotes the score calculated in CN step. See related analysis in §4.6.

Visualization

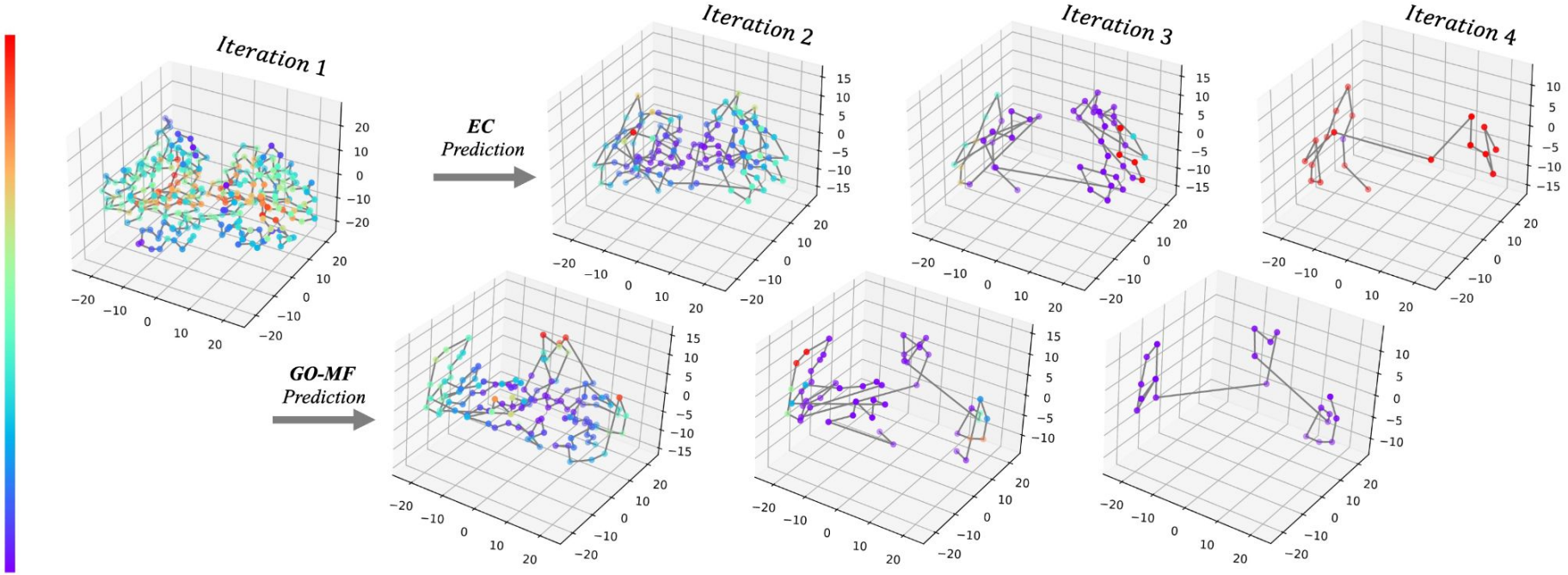


Figure 5. Clustering results for a protein exhibit variations across EC and GO-MF predictions. See related analysis in §4.6.

Visualization

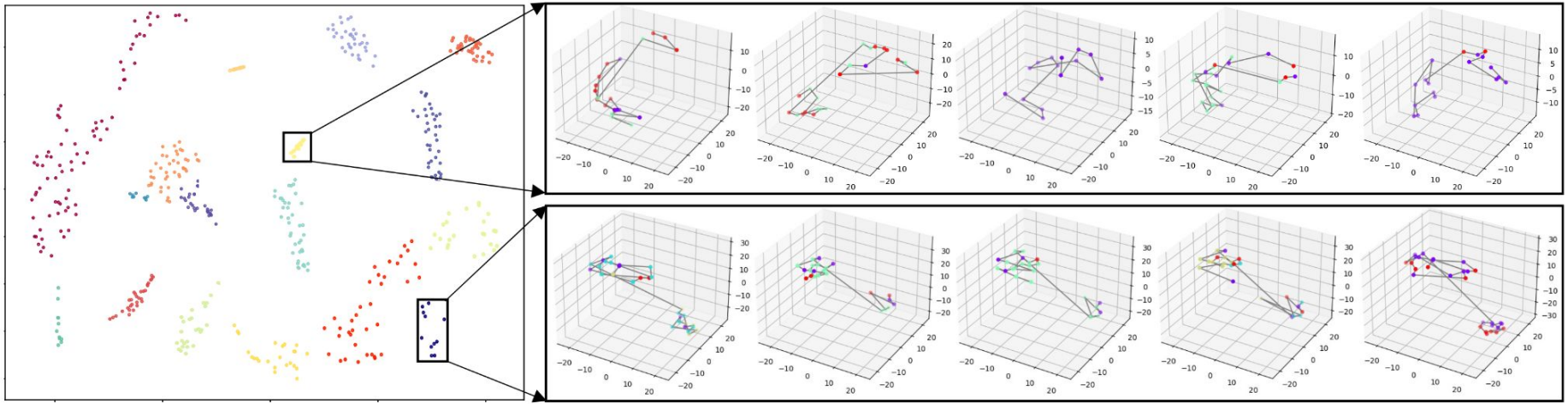


Figure 6. UMAP projection [58] of the learned representation. See related analysis in §4.6.

Conclusion

Effective Protein Representation:

- This research introduces a **neural clustering framework** that **prioritizes critical amino acids** in protein sequences and structures, addressing the **limitations** of traditional methods that treat all **amino acids equally**.
- By leveraging **iterative clustering** (SCI, CRE, CN), this approach refines protein representations, capturing **essential structural and functional motifs** for accurate classification.

State-of-the-Art Performance:

- Extensive evaluations across diverse protein tasks—**fold classification**, **enzyme reaction classification**, **gene ontology term prediction**, and **EC number prediction**—demonstrate this model's ability to deliver **highly accurate predictions**.
- Compared to prior models, it achieves notable accuracy improvements and is both **efficient and scalable**, ensuring suitability for real-world applications in bioinformatics.

Broader Implications:

- **Protein design**: By isolating functionally crucial amino acids, it provides valuable insights for designing novel proteins with targeted properties, advancing fields like drug discovery and enzyme engineering.
- **Structural biology**: Helps identify structural motifs directly linked to biological functions, which could further aid in understanding disease mechanisms at the molecular level.

Future Work

Integration with Protein Language Models:

- Combining this neural clustering method with **large pre-trained protein language models** (e.g., ESM) could yield even more powerful representations.
- This hybrid approach may leverage **clustering's focused feature selection** and **pre-trained models' broad structural knowledge**, improving both the depth and generalizability of protein embeddings.

Self-Supervised Learning Extensions:

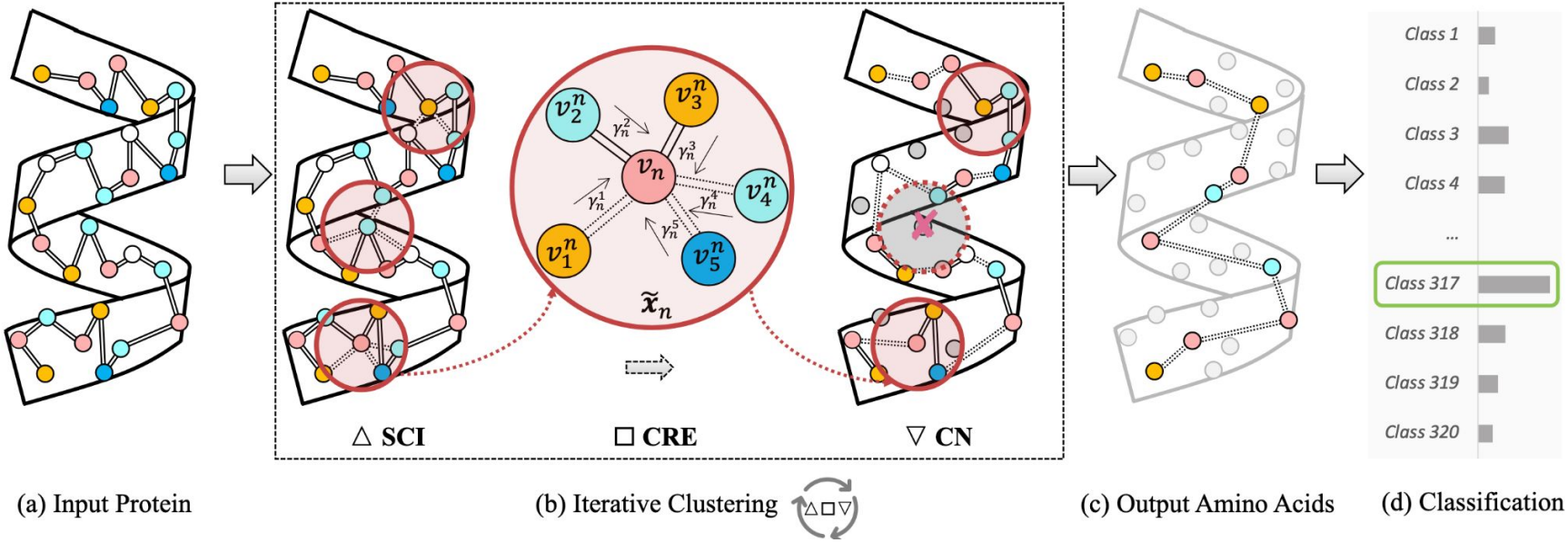
- Implementing **self-supervised learning tasks** within the clustering framework could further enhance its ability to learn meaningful representations from unannotated protein data.
- Future efforts could explore tasks like **masked amino acid prediction** or **contrastive learning** to refine representations without requiring extensive labeled data.

Expanding to Protein Complexes and Interactions:

- Currently focused on individual proteins, this model could be extended to **multi-protein complexes** and **protein-protein interaction** networks.
- Adapting the framework to recognize inter-protein dependencies would enhance its utility in predicting **binding sites** and **interaction dynamics**.

Thanks for listening!

Questions-Comments?



(a) Input Protein (b) Iterative Clustering (c) Output Amino Acids (d) Classification