Unsupervised Learning for Tertiary Structure Prediction of Protein Molecules: Systematic Review

Kazi Lutful Kabir

December 12, 2024

Unsupervised Learning for Tertiary Structure Prediction of Protein Molecules: Systematic Review

Kazi Lutful Kabir[0000−0003−4021−6618]

George Mason University, Fairfax VA 22030, USA
kkabir@gmu.edu

Abstract. Tertiary structures of molecules represent high-dimensional data containing spatial information of hundreds (even thousands) of atoms. Unsupervised learning techniques can be applied to such spatial data to uncover hidden organizations that can be subjected to further evaluation. Such techniques have already been employed in a number of relevant applications e.g., tracking the conformational changes in a set of biomolecular structures, detecting biologically active tertiary structures from computed structures of proteins, analyzing molecular dynamics simulation of peptides, and so on. This paper presents a comprehensive review of clustering techniques for tertiary (3D) molecular structure data focusing on protein molecules. In fact, the article systematically organizes as well as analyzes the existing approaches in terms of data representation, methodology, proximity measure, and evaluation metric. Besides, it highlights key open challenges and proposes future research directions to advance this domain.

Keywords: Clustering · Protein Tertiary Structure · Proximity Measure · Unsupervised Learning

1 Introduction

The tertiary structures of molecules represent the three-dimensional arrangement of atoms, which are highly complex and dynamic, particularly in molecules like proteins they take on different configurations under physiological conditions. Understanding their dynamic behavior requires organizing these structures into structural states, which can be addressed using unsupervised learning, specifically clustering. Proteins exhibit fast transitions within the same state and slower transitions between different states, making clustering a suitable tool to summarize their behavior and identify states relevant to cellular interactions. Clustering, as an optimization problem, lacks universal evaluation metrics, with methods differing based on data representation, proximity measures, and optimization processes. The fundamental steps include selecting appropriate data representation, proximity measures, techniques, and evaluation metrics. Applications in computational biology range from capturing conformational changes in protein structures [13] to detecting macrostates in molecular dynamics simulations [29]. This article reviews and categorizes the existing research, highlighting findings and limitations in clustering methods for protein tertiary structures.

The rest of this paper is organized as follows. Firstly, the key concepts are briefly summarized in Section 2. Section 3 presents an area taxonomy that has been identified and then summarizes existing methods along the identified taxonomy. The article concludes with a summary of future directions (in Section 4) and concluding remarks in Section 5.

Accepted in ICCABS 2025 (13th International Conference on Computational Advances in Bio and Medical Sciences)

2 Preliminaries

2.1 Representation of Protein Molecular Structures

Cartesian Coordinates Tertiary structures of protein molecules are generally represented as ordered sequences of 3D coordinates for their constituent amino acids. For a molecule with N atoms, a naive representation places it as a point S in a $3N$-dimensional space with $S = (x_1, y_1, z_1, \ldots, x_N, y_N, z_N)$. Structural data, such as that stored in the Protein Data Bank (PDB) [11], includes lists of atoms along with their 3D spatial coordinates. To reduce dimensionality, representations often focus on specific subsets of atoms, such as using only the alpha-carbon (C_α) atoms or the backbone atoms (C_α, C, N, and O).

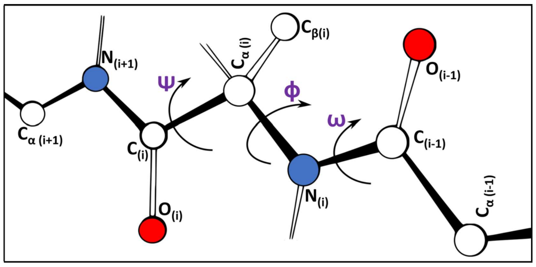

Fig. 1. Backbone Dihedral Angles

Dihedral Angles Instead of Cartesian coordinates, one can use the backbone dihedral/torsion angles (ϕ and ψ angles per amino acid) as features. A dihedral angle is the angle between two planes; the plane formed by the atoms $i-2, i-1, i$ and the plane formed by the atoms $i-1, i, i+1$ where $i-2, i-1, i, i+1$ are four sequentially bonded atoms (Figure 1). The backbone of a protein (which links the backbone atoms) has 3 different torsion angles- phi (ϕ): rotation around N–C$_\alpha$ bond in an amino acid, psi (ψ): rotation around C$_\alpha$–C bond in an amino acid, omega (ω): rotation around C–N bond linking two consecutive amino acids.

Shape-based features Ultrafast Shape Recognition (USR) metrics are used to characterize the 3D shapes of ligands and can be applied to featurize tertiary molecular structures. These metrics rely on moments of the distance distribution of atoms to compare molecular shapes. USR identifies four reference points from a structure: the molecular centroid (ctd), the closest atom to the centroid (cst), the farthest atom from the centroid (fct), and the farthest atom from fct (ftf). The geometry and shape are captured through the mean, variance, and skewness of the distance distributions for these points, resulting in a set of 12 features per structure [45].

Contact Map A contact map is a binary two-dimensional $L \times L$ square matrix, M that represents the distance between all possible residue pairs of a tertiary structure of a protein (L denotes the residue length). The element $M(i, j)$ is 1 if the distance between the residues i and j is less than a predefined threshold and 0 otherwise.

Furthermore, dimensionality reduction techniques are often employed to simplify the representation of molecular structures. Principal Component Analysis (PCA) is utilized [8] to sample protein structure coordinates by capturing correlations among atomic coordinates, particularly in low-energy regions, and to represent dihedral angle distributions [41]. Isometric feature mapping (ISOMAP) is applied to analyze protein trajectories, while Time-lagged Independent Component Analysis (TICA) is used for molecular dynamics data to identify coordinates with maximal auto-correlation over a specific lag time [25], in contrast to PCA's focus on maximal variance.

2.2 Proximity/Distance Measures

To compare the tertiary structures of proteins, a number of similarity/dissimilarity measures are available. Among them, the most prominent ones include:

Root mean square deviation (RMSD): The RMSD between pairs of equivalent atoms is widely used to capture the degree of similarity between two optimally superimposed tertiary structures of a protein. $RMSD = \sqrt{\frac{1}{N}\sum_{1}^{N} \mid S_A^i - S_B^i \mid^2}$ where N is the number of atoms, S_A^i and S_B^i represent the coordinate vectors for i-th atom of the structure A and structure B respectively (after optimal superimposition). One can consider a number of settings to compute RMSD such as: only over C_α atoms [34] or backbone atoms [13, 35] or over all atomic coordinates ($cRMSD$ [24]) of the structures.

Global Distance Test-Total Score (GDT-TS): It determines the similarity between two structures with their corresponding superimposed residues. $GDT - TS(S_A, S_B) = \frac{P_1+P_2+P_4+P_8}{4}$ where P_t denotes the percentage of residues from structure S_A to be superimposed with the corresponding residues from structure S_B having chosen distance threshold, t ($t \in \{1, 2, 4, 8\}$). GDT-TS value ranges from 0 to 1. The larger score indicates better similarity.

Template-Modeling (TM) Score: TM-Score determines the global structural similarity of a structure with respect to the reference structure in terms of the distances of each pair of residues. $TM - Score = max\left[\frac{1}{N}\sum_{i=1}^{N}\frac{1}{1+(\frac{d_i}{d_0})^2}\right]$ where N denotes the number of residues, d_i represents the distance of the i-th pair of residues after alignment. The range of TM-Score values is $(0, 1]$, with a higher value indicating better similarity.

Besides, metrics like contact map overlap(CMO) [24], Tanimoto coefficient (T_c) [21], Local-Global Alignment (LGA) Score [46], and C-score [47] have also been used as proximity measures in different occasions.

3 Systematic Review

Taxonomy-based Survey

Taxonomy provides a systematic way to organize the methods and tools developed in a particular area and most importantly it helps to identify the research gaps in the area. However, no such effort has been found in the literature for unsupervised learning of molecular structures of protein molecules. Hence an attempt is made here based on the comprehension of the research landscape in this domain. Table 1 represents an overview of the methods categorized under four major heads: representation, proximity measure/ distance function used, the types of clustering techniques employed, and the evaluation metrics applied.

Table 1. An Overview of the Available Methods that Employ Unsupervised Learning of Protein Tertiary Structures

Based on	**Categories**	**Instances from Literature**
Representation	Cartesian Coordinates	C_α **atoms**: $Calibur$ [34], $ONION$ [33], $SCUD$ [31]; **Backbone atoms**: Boutonnet et al. [13]; **All-atoms**: $ClusCo$ [24]
	Dihedral Angles	$ICON$ [40]
	Shape-based	Zaman et al. [45], Kabir et al. [29]
	Contact Map	Han et al. [22]
Proximity	Similarity	**TM-Score**: $ONION$ [33], $Calibur$ [34]
	Distance	**RMSD**: $SCAR$ [12], $Durandal$ [9], SCUD [31], $SPICKER$ [48]
Clustering Type	Partitional	$Pleiades$ [23], $SCAR$ [12], $ONION$ [33], $ClusCo$ [24]
	Hierarchical	$HCPM$ [20], Estrada et al. [17], $bcl-Cluster$ [7]
	Graph-based	Akhter et al. [4], Kabir et al. [28], Zhou et al. [49]
	Factorization-based	SNMF-DS [27], NMF-MAD and NMF-Rank [2], NTF-REL [26]
Evaluation	Application-specific	$SCAR$ [12], $SCUD$ [31], $ONION$ [33], $Calibur$ [34]
	Application-agnostic	Li et al. [32], Zhou et al. [49], ClusPro [15]

3.1 Representation

Tertiary structures of proteins are three-dimensional objects, they have shape and occupy volume in space as they are composed of atoms occupying positions. The atoms are not free-floating and connect to each other with links/bonds. Hence the obvious key question one would have to answer first is that if we want to cluster three-dimensional objects, how do we represent them? What do we encode that will allow us to recognize any inherent organization?

The most popular and intuitive way is to consider the structures as ordered sequences of 3D coordinates and represent them using Cartesian coordinates. One has to make a decision regarding whether to keep all of the atoms (all-atoms [24]) or a specific type of atom (C_α) [31,33,34] or a group of atoms (backbone atoms [13]). While suitable for classic distance metrics like Euclidean distance, this representation is extremely high-dimensional, leading to issues such as the curse of dimensionality and reduced performance for clustering algorithms.

On the other hand, the changes in molecular structures can be considered as the outcome of rotations around bonds that connect atoms. In fact, the comparison of structures (at room temperature) reveals that changes in angles are due to some specific ones (dihedral angles) [40]. Dihedral angle-based representation ensures dimensionality reduction by a factor of 7 over the Cartesian coordinates [36]. The most intuitive distance function for this representation would be L1-norm. However, it is necessary to go beyond the L1-norm to design more meaningful distance functions as all angles are not equally important. Because if we interpret angles as rotations, changes in angles at the beginning of the chain of atoms bring a larger impact (in terms of the swept volume in 3D) than changes in angles at the end of the chain.

Besides, changes in atomic positions or angles ultimately result in changes to the shape of the structure. And it is possible to come up with the coarse representation of shape. USR metrics summarize the distance distribution of atoms from four reference points via mean, variance, and skewness [45]. This mechanism ensures dimensionality reduction by a good margin but fails to capture subtle structural changes.

Contact maps capture the internal geometry of the structure and bring forth a more reduced representation of protein tertiary structures (in comparison to their entire three-dimensional atomic coordinates) and are also invariant to rotations and translations. Moreover, a contact map can also be represented by an ordered graph [22]. However, this representation throws out subtle information regarding structures.

3.2 Proximity/Distance Measure

Proximity measures or distance functions (for comparing protein tertiary structures) are primarily focused on Cartesian coordinate-based representation. The most popular and intuitive distance function is RMSD which is a variant of Euclidean distance. The RMSD value is dependent on the size of the molecule (number of atoms). The computation is time-demanding as it requires that the structures be aligned first to remove differences due to rigid-body motions (whole-body translation and whole-body rotation in 3D). Similarity metric GDT-TS aims to address the dependency of RMSD on the number of atoms. It first finds the subsets of atoms within certain thresholds of RMSD (1Å, 2Å, 4Å, 8Å) and then reports an average of these percentages over the thresholds. GDT-TS is more accurate than RMSD at capturing structural differences but it is also computationally demanding. On the other hand, TM-score weights shorter distances between corresponding atoms more strongly than longer distances. It ensures more sensitivity to global topology rather than local structure deviations. The magnitude of the TM-score is length-independent for random structure pairs.

3.3 Clustering Techniques

Clustering algorithms identify groups of observations that are more similar to each other than to the observations of other groups. For the clustering of protein tertiary structures, the strategy followed by most systems (e.g., *ROSETTA* [39], *I-TASSER* [44], *SPICKER* [48], *SCUD* [31], *Calibur* [34]) can be summarized by the following steps:

1. For a given set of structures, choose a threshold, t_h for the proximity measure
2. The structure with the most neighbors within distance t_h from it is extracted and is reported as the structure with the highest rank (choose arbitrarily in case of ties)
3. This structure and all of its neighbors form the first cluster and are removed
4. Repeat steps (3) and (4) until no further clusters are found

Techniques for clustering protein tertiary structure fall into four major branches: partitioning methods, hierarchical methods, graph-based methods, and factorization-based methods. Partitional clustering divides the set of observations into non-overlapping subsets (clusters) in such a way that each observation is in exactly one cluster. Hierarchical methods start with each observation forming a separate cluster and eventually construct a set of nested clusters that can be represented by a tree.

Partition-based Algorithms: `k-means` is the most popular partition-based clustering method. While dealing with the structure data, `k-means` tries to solve the following problem: Given n structures $S_1, S_2, S_3, ..., S_n$, `k-means` attempts to group the structures into k clusters $(A_1, A_2, A_3, ..., A_k)$ to optimize the objective function of k-means. In fact, `k-means` is considered as the baseline in [3, 45].

Pleiades is a k-means-based method for clustering protein structures that uses a 31-dimensional tuned Gauss integral (GIT) vector representation of the structures to approximate the RMSD. The Euclidean distance (having a fair correlation with the RMSD) measures the proximity between two structures [23]. While faster than RMSD-based k-means, GIT representation may map different structures to the same vector. RMSD-based k-means, though slower, produce more accurate clusters, showing that k-means can offset GIT's limitations. Pleiades' key advantage is its computational speed [23].

SCAR is a k-means-inspired clustering method that uses Relative RMSD (RRMSD) as a universal proximity measure and structure packing number for adaptive local cluster cutoff that considers the variability in the cluster's internal dispersion. It includes a refinement step with centroid realignment via singular value decomposition (SVD). Overlapping clusters are resolved by removing the one with the lower packing density (a measure of cluster cohesion). RRMSD follows a universal distribution (mean: 1.0, standard deviation: 0.09), enabling robust clustering, while centroids represent global consensus topology. The method minimizes inter-cluster correlation by ensuring centroid distances exceed individual cluster dispersion. Adjustable cutoffs enhance flexibility, and packing density captures cluster cohesion and dispersion errors [12].

ONION operates similarly to k-means, with a slightly modified objective function minimizing RMSD between structures and optimal centroids obtained via superimposition, $arg\ min \sum_{i=1}^{k} \sum_{S_j \in A_i} RMSD^2(O_i, S_j)$ where O_i denotes the optimal centroid of the set of structures A_i approximated via multiple structure superimposition [33]. It determines the optimal k (number of clusters) using a Gaussian mixture model with Schwartz's Bayesian information criterion (BIC). Centroids are estimated with random sampling, and pruning rotation spaces for globular proteins avoid pairwise RMSD calculations. ONION outperforms Pleiades with a polynomial-time approximation scheme, is faster than Calibur [34], and is comparable to SPICKER [48] in performance.

ClusCo The main motivation behind the development of *ClusCo* software is to build a high-throughput tool for all-versus-all comparison of protein structures with different proximity measures using parallel `k-means` clustering. In terms of execution time, *Durandal* [9] is faster than *ClusCo* [24], *Calibur* or *SPICKER. Calibur* uses heuristics-based preprocessing to speed-up clustering in three ways: grouping of structures into proximity groups to avoid pairwise RMSD computation (triangular inequality), utilizing efficiently computable upper and lower bounds to skip RMSD calculation whenever possible, discarding structures with low similarity to other structures (before clustering).

Calibur is faster than SPICKER, but SPICKER produces better structures by focusing on low-energy regions and self-adjusting RMSD cutoffs. SPICKER outperforms SCAR in terms of the RMSD for top clusters. In [42], *SPICKER* is used along with additional steps (filtering and cluster reduction based on multidimensional scaling). SCUD avoids pairwise RMSD by using a random reference and sets protein-dependent cutoffs to balance cluster sizes [31]. Durandal [9] accelerates clustering using triangular inequality and initializes distances with a random reference, incorporating quaternion-based characteristic polynomial (QCP)-oriented RMSD for efficiency via information gain-based approach [10]. MUFOLD-CL [47], using the D-Score metric, is the fastest and produces superior structures compared to SPICKER, Pleiades, and Calibur.

SK-means [43] combines *SPICKER* and k-means to improve the selection of initial cluster centers, addressing a key limitation of basic k-means. This method outperforms SPICKER in terms of average TM-score for the reported best structure. SK-means has quadratic polynomial time complexity.

Medoids-based Li et al. [32] propose an ensemble clustering method based on k-medoids for protein structures. Unlike k-means, k-medoids selects actual data points as centroids. The method generates multiple clustering outcomes through repeated k-medoid runs with random initializations, then combines them using a voting-based approach. It uses the TM score to build a distance matrix and identify cluster centers. It outperforms SPICKER in final structure selection. The method is also utilized in [22].

Ferone and Marateas [18] propose graded Possibilistic c-Medoids, a medoid-based variation of graded possibilistic clustering combining fuzzy c-means (FCM) and possibilistic c-means (PCM). Unlike standard fuzzy clustering, GPCMdd allows membership sums greater than 1, enabling a soft transition between probabilistic and possibilistic distance functions. This makes it more robust to outliers and noise, effectively handling loosely related data. While outperforming other medoid-based methods and SPICKER, its high computational cost limits its feasibility for large structure datasets.

Hierarchical Clustering Methods: Strategies for hierarchical clustering generally fall into two major types: agglomerative (a bottom-up strategy starting with each observation as its own cluster, and consequent merging of pairs of clusters) and divisive (a top-down approach starting with all observations as a single cluster, and recursive splits of the clusters)

Hierarchical Clustering of Protein Models (HCPM) [20] uses an agglomerative clustering strategy with an average-link measure to calculate distances between clusters, measured by cRMSD as the average inter-atomic distances. The cluster representative is the structure nearest to the average distance map. The merging distance cut-off is determined using the plateau-center finding approach in a sigmoid plot or by evaluating several probe values and their cluster parameters. HCPM can also explore local energy minima in a protein energy landscape [19], often applied with the CABS method, which models proteins using four interaction centers per amino acid: C_α, C_β, the peptide bond center, and the side-chain center of mass.

UQlust [1] combines structural profiles with consensus ranking and profile hashing for efficient hierarchical agglomerative clustering of protein structures. It projects 3D coordinates into single-dimensional structural profiles by assigning each residue to a specific state, facilitating the comparison of structures to detect common substructures. UQlust employs two heuristics: profile hashing-based clustering and reference-based partitioning, offering better time and space efficiency compared to ClusCo. Additionally, Boutonnet et al. [13] came up with a multiple linkage hierarchical clustering algorithm to analyze protein structural changes.

bcl-Cluster [7] employs an agglomerative hierarchical clustering algorithm using pre-calculated pairwise distances between structures. It supports various proximity measures, including GDT, longest continuous segment, MaxSub, RMSD, largest common substructure, and Tanimoto coefficient. A pre-clustering step combines structures with specified similarity into clusters in a single pass. Integrated with PyMOL, bcl-Cluster provides detailed outputs, including dendrograms, molecular structures, cluster sizes, and color-coded results based on numerical descriptors.

Probabilistic Hierarchical Clustering [17] combines fuzzy c-means clustering (FCM) and a divisive hierarchical algorithm and identifies the number of clusters dynamically. The clustering cutoff is probabilistically determined based on variability. FCM starts with randomly chosen centroids and iteratively computes the degree of belonging for each structure using a normalized inverse distance from the centroid. New centroids are calculated as the weighted mean of structures, and the process repeats until convergence. Then divisive hierarchical clustering splits the structures into two subsets, temporarily removing redundant structures. It selects partitions based on a probability proportional to size and inversely proportional to internal variance, continuing subdivision until partition means are statistically significant.

Parallel Ward Clustering [16] employs a massively parallel CUDA implementation of the nearest neighbor chain algorithm for hierarchical Ward clustering of protein structures, using atom-based RMSD and rigid-body RMSD. For atom-based RMSD clustering, three strategies are analyzed: Threads per Atomic Coordinate (TAC), Thread Blocks per Centroid (TBC), and Threads per Cluster Centroid (TCC). Among these, the TCC approach demonstrates significant speedup over multi-threaded CPU implementations and surpasses ClusCo in performance, while also enabling the computation of the full hierarchical tree. Additionally, ClusPro [15] applies a pairwise RMSD-based hierarchical algorithm to cluster protein structures after an initial filtering step based on desolvation and electrostatic properties.

Graph-based Clustering Methods: For graph-based techniques, the first step is to represent the given set of structures in terms of a graph. An intuitive way is to employ the nearest-neighbor graph.

Nearest-neighbor Graph The proximity of the molecular structures under consideration can be encoded in the structure space via a nearest-neighbor graph (nngraph). Consider Ω, a set of structures that can be embedded in a nearest-neighbor graph (nngraph) $G = (V,\ E)$ where the vertex set V is populated with the structures, and the edge set E is populated by inferring a local neighborhood over each observation. The distance between two structures can be measured in terms of a suitable proximity measure (and an appropriate threshold for the same) after optimal superimposition with respect to a reference structure. In fact, such graph embedding of molecular structures of proteins has been considered in [4,28] under the context of template-free protein structure prediction.

Zhou et al. [49] employ techniques for constructing amino acid network-based graphs, also known as residue interaction graphs (RIG), where nodes represent amino acids through all atoms, side-chain atoms, or only C_α or C_β atoms. Edges in the graph are characterized by similarity measures such as physical distance or residue interaction energy between atomic units or side chains. Another work [28] utilizes nearest-neighbor graphs (nngraphs) and community detection algorithms, originally developed for social networks, to group protein structures. On the other hand, the `Basins-Select` [4] detects basins in the energy landscape by considering the potential energies of protein structures. It constructs a nearest-neighbor graph (nngraph) and extracts basins by first locating points of attraction or focal minima, which are vertices representing local energy minima. Each local minimum represents a basin. The other vertices are assigned to basins by following a negative gradient descent, determined by the edge (u, v) that maximizes the ratio $[e(u)–e(v)]/d(u, v)$, where $e(u)$ is the energy of structure represented by vertex u and $d(u, v)$ is the distance between vertices u and v. This process continues until a local minimum is reached, with all vertices converging to the same minimum assigned to the corresponding basin. It leverages the Structural Bioinformatics Library (SBL) [14] to decompose the structure nngraph into basins.

Factorization-based Methods: NMF-MAD [2] opens up a new avenue by exploring the

Table 2. Comparison of Different Methods

Name	Key Characteristics	Comparison found in literature	Evaluation Criteria
SPICKER [48]	– shrink the dataset (least energy structures from subsets) – pairwise RMSD cutoff to determine cluster membership – structure with the most neighbors as cluster center	better than SCAR by the quality of the top cluster	RMSD from an experimentally known structure
SCAR [12]	– relative RMSD (RRMSD) – global cluster cutoff to determine cluster membership – cluster-wise cutoff for refinement	demonstrates the effectiveness of RRMSD over RMSD	Avg. RMSD (cluster)
ONION [33]	– centroid approximation by MULTIPLE STRUCTURE SUPERIMPOSITION – similar to kmeans (by objective function)	– slightly better than SPICKER in single model selection – faster than Calibur and SPICKER	TM-Score
Calibur [34]	– auxiliary grouping of structures (threshold selection with C_α RMSD) – preliminary screening via lower and upper bounds – filtering of highly dissimilar structures from the dataset	– faster than SPICKER – slightly better than SPICKER	– Avg. TM-Score w.r.t. the experimentally known structure (cluster) – C_α RMSD w.r.t. the experimentally known structure (single structure)
ClusCo [24]	– high-throughput comparison of protein structures – different similarity measures (RMSD, GDT-TS, TM-Score, MaxSub)	– clustering results comparable to Calibur – faster than SPICKER and Calibur but slower than Durandal	comparison with ref. structure by different similarity measures (RMSD, GDT-TS, TM-Score, MaxSub)
Durandal [9]	– randomly chosen reference for distance matrix computation – lower and upper bound strategy to favor distance ranges over exact measures – uses the triangular inequality to accelerate exact clustering	faster than Calibur and SCUD	RMSD
Pleiades [23]	– Gauss integral representation for the tertiary structures of proteins – approximation of RMSD via Euclidean distance	– (slightly better) comparable to Calibur by all-atom RMSD – faster than Calibur	RMSD
SCUD [31]	– RMSD from randomly chosen reference as proximity measure – most neighboring structure: cluster representative – Representative structures are ranked by cluster size	slower than Durandal	RMSD
SK-means [43]	– integrates SPICKER with k-means – centroid of the largest cluster: best structure	better than SPICKER	RMSD
Ensemble method based on k-medoids [32]	– k-medoids algorithm is run several times (with different initializations) – voting to combine the clustering outcomes – largest cluster as the best cluster	– better than SPICKER – time-consuming	RMSD
Graph Clustering Methods [4, 28]	– encoding of proximity via nearest neighbor graph – community detection/ energy landscape analysis	landscape analysis-based methods perform better than community-based methods	Purity
MUFOLD-CL [47]	– D-Scores based measures having high correlation with RMSD and TM-Score – projection-based clustering – largest cluster as the best cluster	– faster than most of the approaches – better than SPICKER, Pleiades, Calibur and a bit worse than ONION (avg. RMSD of prototype of top five clusters)	– avg. RMSD from the experimentally known structure (cluster) – RMSD from the experimentally known structure (single structure)
HCPM [20]	– agglomerative strategy with the average link – cluster representative: the structure that is the closest to the the average distance map of the cluster – initial screening based on energy and gyration radius	slower than k-means-based methods but performs better than those	RMSD
UQlust [1]	– projection of 3D coordinates into a suitable 1D structural profile – geometric consensus ranking – largest cluster: best cluster	– time and memory efficient than ClusCo, Sk-means – better than ClusCo and Pleiades	MaxSub Score
bcl-Cluster [7]	– relies upon pre-calculated pairwise distances – accommodates a variety of proximity measures – offers a pre-clustering step	offers a wide range of proximity measures to choose from	several proximity measures
Parallel Ward Clustering [16]	the nearest neighbor chain algorithm for hierarchical Ward clustering of protein structures	– faster than ClusCo – clustering results comparable to Clusco	TM-Score (for single structure selection)
NMF-MAD [2]	domain-specific feature-based non-negative matrix factorization	better than the methods mentioned before this one in this table	RMSD (for single structure selection)
SNMF-DS [27]	symmetric non-negative matrix factorization on RMSD-based distance matrix	better than NMF-MAD	RMSD (for single structure selection)
NTF-REL [26]	utilizes tensors to capture multiview of protein tertiary structures	better than all of the other approaches mentioned in this table in terms of quality assessment	multiple proximity measures

route of matrix factorization to identify biologically active protein tertiary structures via utilization of the domain-specific features. SNMF-DS [27] dives in this direction further to demonstrate a feature-free and non-parametric method based on symmetric non-negative matrix factorization. Furthermore, NTF-REL [26] shifts from matrix-factorization to tensor-factorization to serve as a quality assessment method for protein tertiary structures in addition to grouping the same.

Other mentionable techniques for unsupervised learning of molecular structures include-uniform time clustering, regular space clustering along with Markov state modeling (MSM) offered by *PyEMMA* [37], *ICON* [40], granular clustering based on growing local similarities [21], clustering based on the distribution of local minima [30], conserved residue clustering [38], score-based clustering using ligand RMSD. Table 2 captures the key properties as well as provides a brief comparison of the discussed methods.

3.4 Evaluation Metrics

Two sets of metrics are typically employed in unsupervised learning literature. The first set (consisting of external metrics) is designated here as application-specific. The second set contains internal metrics, termed here as application-agnostic. It is worth noting that unsupervised learning research for molecular structure data presents several adaptations of both external and internal metrics as inspired by domain insights.

Application-specific Metrics
While most of these works assume that there is no external information about what states the clusters capture (that is, no ground truth), they often leverage the availability of known experimental structure(s) for a protein of interest in the PDB. Many external metrics take this into account.
Applying Similarity/Distance Metrics: SCAR computes the average RMSD of the structures (w.r.t. experimentally known structure) in a cluster to compare the quality of the cluster [12]. A similar process is followed in [9,23,31,47,48](with TM-score in [33,34]).
Purity: This measure is used with the graph-based techniques [4,28]. For a given cluster, it computes the fraction of structures that are very similar to the experimentally known structure in the cluster over the total number of structures contained by the cluster [4].

Application-agnostic Metrics
The works that employ these metrics don't take into account the availability of the experimentally known structure or other external information.
Utilization of Proximity/Distance measures: *ClusCo* selects the best cluster by $\texttt{min(}\frac{\langle R\rangle}{f}\texttt{)}$, where $\texttt{f}$ denotes the fraction of elements in a particular cluster and $\langle R\rangle$ denotes the average *RMSD* between cluster elements. And the cluster center of the best cluster is selected as the best structure. *ClusPro* [15] takes into account the center of the most populated cluster as the best structure. To determine the cluster quality, *SPICKER* considers normalized structure cluster density, D defined as, $D = \frac{M}{\langle RMSD\rangle M_t}$, where M is the multiplicity of structures in the cluster, M_t is the total number of structures to be clustered and $\langle RMSD\rangle$ denotes the average RMSD of the structures in the cluster. *ICON* [40] measures the cluster quality in terms of cluster concentration, $CC_j = \frac{N_j}{\overline{RMSD_j}}$; where N_j is the number of structures in the cluster and $\overline{RMSD_j}$ denotes the average RMSD of the cluster. Besides, *UQlust* [1] reports the centroids of the five largest clusters as the top scoring structures. *HCPM* [19,20] considers *RMSD* distribution of the intra-cluster structures and inter-cluster's centroids to

assess the clusters' quality.

Clustering Coefficient: Network properties e.g., average degree, clustering coefficient, and size of the communities are taken into account by Zhou et al. [49] to analyze the communities/clusters obtained from amino acid network-based graph representation of the structures. The Clustering Coefficient, C can be computed as, $C_i = \frac{2E_i}{m(m-1)}$ where m is the degree of vertex i and E_i denotes the number of connections from all neighbors of i. The clustering coefficient C of a cluster is the average of C_i.

Confidence score: To capture the size and internal similarity of a cluster, Li et al. [32] compute a confidence score for each cluster, $CS = \frac{\sum_{i \in c} \sum_{j \in c} sim(i,j)}{\sum_{i=1}^{n} \sum_{j=1}^{n} sim(i,j)}$ where n is the total number of structures, c is the cluster under observation and $sim(i,j)$ denotes the corresponding entry in the similarity matrix. The cluster center with the maximum confidence score contributes to the selection of the best structure.

4 Discussion

Even though significant works have been conducted (most being application-specific), there are ample open problems and ways forward such as:

i. How to pick a structure that represents a cluster (beyond traditional ways of doing that)? This is an important question to answer, particularly when the clustering is in support of a specific application or when the goal is data reduction. For instance, Akhter et al. [5] consider the potential energy of a structure and even employ a density-based structure weighting scheme to do so.

ii. The design of a proper distance function will remain an interesting direction of research. For instance, how to design a meaningful distance function (beyond L1-norm) for the dihedral angle representation as well as take it into account for clustering? The distance function should be robust in response to experimental and modeling errors and should have the capability to capture the proximity between structures at any level of resolution.

iii. Are the structures themselves sufficient for finding patterns or consideration of additional properties of molecular structures (e.g., energy-based features [2]) is needed to improve the clustering results? Moreover, representation learning via auto-encoders [6] may prove useful to highlight the main dimensions that possibly reveal interesting organizations.

iv. Subspace clustering presents an unleveraged direction at the moment. Furthermore, subspace clustering promises to find the subset of dimensions (all dimensions might not be relevant) that reveal an informative grouping of the clusters. One can also consider dimension weighting and reduce the problem to learning the weights of the different dimensions for optimizing an objective function that corresponds to the quality of the clustering.

5 Conclusion

This article presents an effort to organize the landscape of research on the clustering of molecular structures of proteins. As highlighted, direct comparison of existing methods is sometimes difficult due to the lack of standards with regard to benchmark datasets and metrics used. As identified above, there are several directions of research that may improve current efforts to reveal the underlying organization of molecular structures to elucidate structural states as an exploratory step toward understanding the behavior of a dynamic molecule.

References

1. Adamczak, R., Meller, J.: Uqlust: combining profile hashing with linear-time ranking for efficient clustering and analysis of big macromolecular data. BMC bioinformatics **17**(1), 546 (2016)

2. Akhter, N., et al.: Improved protein decoy selection via non-negative matrix factorization. IEEE/ACM Transactions on Computational Biology and Bioinformatics **19**, 1670–1682 (2021)
3. Akhter, N., Chennupati, G., Kabir, K.L., Djidjev, H., Shehu, A.: Unsupervised and supervised learning over the energy landscape for protein decoy selection. Biomolecules **9**(10), 607 (2019)
4. Akhter, N., Shehu, A.: From extraction of local structures of protein energy landscapes to improved decoy selection in template-free protein structure prediction. Molecules **23**(1) (2018)
5. Akhter, N., Vangara, R., Chennupati, G., Alexandrov, B.S., Djidjev, H., Shehu, A.: Non-negative matrix factorization for selection of near-native protein tertiary structures. In: Int'l Conf. on Bioinformatics and Biomedicine (BIBM). pp. 70–73. IEEE (2019)
6. Alam, F.F., et al.: Learning reduced latent representations of protein structure data. In: The 10th ACM Int'l Conf. on Bioinformatics, Computational Biology and Health Informatics. pp. 592–597 (2019)
7. Alexander, N., et al.: bcl:: Cluster: A method for clustering biological molecules coupled with visualization in the pymol molecular graphics system. In: The 1st Int'l Conf. on Computational Advances in Bio and Medical Sciences (ICCABS). pp. 13–18. IEEE (2011)
8. Álvarez, Ó., et al.: Principal component analysis in protein tertiary structure prediction. Journal of bioinformatics and computational biology **16**(02) (2018)
9. Berenger, F., Shrestha, R., Zhou, Y., Simoncini, D., Zhang, K.Y.: Durandal: fast exact clustering of protein decoys. Journal of computational chemistry **33**(4), 471–474 (2012)
10. Berenger, F., Zhou, Y., Shrestha, R., Zhang, K.Y.: Entropy-accelerated exact clustering of protein decoys. Bioinformatics **27**(7), 939–945 (2011)
11. Berman, H.M., Bourne, P.E., Westbrook, J., Zardecki, C.: The protein data bank. In: Protein Structure, pp. 394–410. CRC Press (2003)
12. Betancourt, M.R., Skolnick, J.: Finding the needle in a haystack: educing native folds from ambiguous ab initio protein structure predictions. Journal of Computational Chemistry **22**(3), 339–353 (2001)
13. Boutonnet, N.S., Rooman, M.J., Wodak, S.J.: Automatic analysis of protein conformational changes by multiple linkage clustering. Journal of molecular biology **253**(4), 633–647 (1995)
14. Cazals, F., Dreyfus, T.: The structural bioinformatics library: modeling in biomolecular science and beyond. Bioinformatics **33**(7), 997–1004 (2017)
15. Comeau, S.R., et al.: Cluspro: an automated docking and discrimination method for the prediction of protein complexes. Bioinformatics **20**(1), 45–50 (2004)
16. Dang, H.V., Schmidt, B., Hildebrandt, A., Tran, T.T., Hildebrandt, A.K.: Cuda-enabled hierarchical ward clustering of protein structures based on the nearest neighbour chain algorithm. The Int'l Journal of High Performance Computing Applications **30**(2), 200–211 (2016)
17. Estrada, T., Armen, R., Taufer, M.: Automatic selection of near-native protein-ligand conformations using a hierarchical clustering and volunteer computing. In: The 1st ACM Int'l Conf. on Bioinformatics and Computational Biology. pp. 204–213 (2010)
18. Ferone, A., Maratea, A.: Decoy clustering through graded possibilistic c-medoids. In: Int'l Conf. on Fuzzy Systems. pp. 1–6. IEEE (2017)
19. Gront, D., Hansmann, U.H., Kolinski, A.: Exploring protein energy landscapes with hierarchical clustering. Int'l journal of quantum chemistry **105**(6), 826–830 (2005)
20. Gront, D., Kolinski, A.: Hcpm—program for hierarchical clustering of protein models. Bioinformatics **21**(14), 3179–3180 (2005)
21. Guzenko, D., Strelkov, S.V.: Granular clustering of de novo protein models. Bioinformatics **33**(3), 390–396 (2017)
22. Han, X., Li, L., Lu, Y.: Selecting near-native protein structures from predicted decoy sets using ordered graphlet degree similarity. Genes **10**(2), 132 (2019)
23. Harder, T., Borg, M., Boomsma, W., Røgen, P., Hamelryck, T.: Fast large-scale clustering of protein structures using gauss integrals. Bioinformatics **28**(4), 510–515 (2012)
24. Jamroz, M., Kolinski, A.: Clusco: clustering and comparison of protein models. BMC Bioinformatics **14**(1), 62 (2013)
25. Kabir, K.L., Akhter, N., Shehu, A.: Connecting molecular energy landscape analysis with markov model-based analysis of equilibrium structural dynamics. In: The 11th Int'l Conf. on Bioinformatics and Computational Biology (BICOB). vol. 60, pp. 181–189 (2019)

26. Kabir, K.L., Bhattarai, M., Alexandrov, B.S., Shehu, A.: Single model quality estimation of protein structures via non-negative tensor factorization. In: The 11th Int'l Conf. on Computational Advances in Bio and Medical Sciences (ICCABS). pp. 3–15 (2021)
27. Kabir, K.L., Chennupati, G., Vangara, R., Djidjev, H., Alexandrov, B.S., Shehu, A.: Decoy selection in protein structure determination via symmetric non-negative matrix factorization. In: Int'l Conf. on Bioinformatics and Biomedicine (BIBM). pp. 23–28. IEEE (2020)
28. Kabir, K.L., Hassan, L., Rajabi, Z., Akhter, N., Shehu, A.: Graph-based community detection for decoy selection in template-free protein structure prediction. Molecules **24**(5), 854 (2019)
29. Kabir, K.L., Ma, B., Nussinov, R., Shehu, A.: Fewer dimensions, more structures for improved discrete models of dynamics of free versus antigen-bound antibody. Biomolecules **12**(7) (2022)
30. Li, H.: A model of local-minima distribution on conformational space and its application to PSP. Proteins: Structure, Function, and Bioinformatics **64**(4), 985–991 (2006)
31. Li, H., Zhou, Y.: Scud: Fast structure clustering of decoys using reference state to remove overall rotation. Journal of computational chemistry **26**(11), 1189–1192 (2005)
32. Li, L., Yan, H., Lu, Y.: Selecting near-native protein structures from ab initio models using ensemble clustering. Quantitative Biology **6**(4), 307–312 (2018)
33. Li, S.C., Bu, D., Li, M.: Clustering 100,000 protein structure decoys in minutes. IEEE/ACM Transactions on Computational Biology and Bioinformatics **9**(3), 765–773 (2011)
34. Li, S.C., Ng, Y.K.: Calibur: a tool for clustering large numbers of protein decoys. BMC Bioinformatics **11**(1), 25 (2010)
35. Mereghetti, P., Ganadu, M.L., Papaleo, E., Fantucci, P., De Gioia, L.: Validation of protein models by a neural network approach. BMC bioinformatics **9**(1), 66 (2008)
36. Moll, M., Schwarz, D., Kavraki, L.E.: Roadmap methods for protein folding. Protein Structure Prediction pp. 219–239 (2008)
37. Scherer, M.K., et al.: Pyemma 2: A software package for estimation, validation, and analysis of markov models. Journal of chemical theory and computation **11**(11), 5525–5542 (2015)
38. Schueler-Furman, O., Baker, D.: Conserved residue clustering and protein structure prediction. Proteins: Structure, Function, and Bioinformatics **52**(2), 225–235 (2003)
39. Shortle, D., Simons, K.T., Baker, D.: Clustering of low-energy conformations near the native structures of small proteins. National Academy of Sciences **95**(19), 11158–11162 (1998)
40. Subramani, A., DiMaggio Jr, P.A., Floudas, C.A.: Selecting high quality protein structures from diverse conformational ensembles. Biophysical journal **97**(6), 1728–1736 (2009)
41. Tribello, G.A., Gasparotto, P.: Using dimensionality reduction to analyze protein trajectories. Frontiers in Molecular Biosciences **6**, 46 (2019)
42. Wang, Q., Shang, Y., Xu, D.: A new clustering-based method for protein structure selection. In: Int'l Joint Conf. on Neural Networks. pp. 2891–2898. IEEE (2008)
43. Wu, H., Li, H., Jiang, M., Chen, C., Lv, Q., Wu, C.: Identify high-quality protein structural models by enhanced k-means. BioMed Research (2017)
44. Wu, S., Skolnick, J., Zhang, Y.: Ab initio modeling of small proteins by iterative tasser simulations. BMC biology **5**(1), 17 (2007)
45. Zaman, A.B., Kamranfar, P., Domeniconi, C., Shehu, A.: Reducing ensembles of protein tertiary structures generated de novo via clustering. Molecules **25**(9), 2228 (2020)
46. Zemla, A.: Lga: a method for finding 3d similarities in protein structures. Nucleic acids research **31**(13), 3370–3374 (2003)
47. Zhang, J., Xu, D.: Fast algorithm for clustering a large number of protein structural decoys. In: Int'l Conf. on Bioinformatics and Biomedicine. pp. 30–36. IEEE (2011)
48. Zhang, Y., Skolnick, J.: Spicker: a clustering approach to identify near-native protein folds. Journal of computational chemistry **25**(6), 865–871 (2004)
49. Zhou, J., Yan, W., Hu, G., Shen, B.: Amino acid network for the discrimination of native protein structures from decoys. Protein and Peptide Science **15**(6), 522–528 (2014)