TRANSFORMERS, PROTEINS, AND PERSISTENT HOMOLOGY

AMELIE SCHREIBER

Abstract. The exploration and analysis of protein sequences and structures is a cornerstone in computational biology and bioinformatics. While existing approaches provide significant insights, the inherent high-dimensionality of protein data often conceals nuanced topological and geometric features. In this study, we apply Topological Data Analysis (TDA), specifically persistent homology, as a robust mathematical tool for revealing these hidden characteristics. We construct simplicial complexes from the high-dimensional embeddings of protein sequences, such as the attention probability distributions, context vectors, and hidden states of transformer-based models, and compute their persistent homology. The resulting persistence diagrams capture both local and global features of proteins, enabling a novel perspective for protein comparison and clustering. Moreover, we demonstrate the importance of preserving the persistent homology of sequence and structural motifs across different contexts and propose the inclusion of a topological term in the loss function of the model to enforce this preservation as an inductive bias. We also provide a second topological loss function for a knowledge distillations procedure. We further extend our approach to anomaly detection in collections of protein sequences, utilizing the Fréchet mean persistence diagram as a statistical baseline. We provide a new way of discovering sequential and structural motifs and other important substructures using the simplex tree, a new clustering method, and a visualization tool. Our findings underscore the significant potential of applying persistent homology to elucidate complex patterns in protein sequence and structure data, offering a promising pathway for further advancements in protein bioinformatics.

Contents

Date: July 18, 2023.

2020 *Mathematics Subject Classification.* Primary 68T07, 55N31, 62R40, 68T09, 92B20, Secondary 92C40, 92-08, 68T50, 68T10, 68T45 .

1. Introduction

The field of computational biology and bioinformatics frequently encounters the challenge of analyzing complex, high-dimensional data, especially when studying protein sequences and structures. We find this especially in the internal representations of protein language models like ESM-2. As the fundamental building blocks of life, proteins harbor an abundance of biological information within their sequences and three-dimensional structures. Our goal is to unravel this information to understand sequence patterns, structural motifs, and the intricate relationship between protein sequence, structure, and function.

While the high-dimensionality of protein data encoded in ESM-2 often obscures critical insights, advancements in mathematical and computational techniques offer promising solutions. In this regard, Topological Data Analysis (TDA), and particularly persistent homology, emerges as a potent tool for capturing the multi-scale topological features hidden within the data. Persistent homology is a central pillar of TDA, widely recognized for its ability to discern the 'shape' of data. By constructing simplicial complexes that encapsulate multi-scale geometric and topological information, persistent homology uncovers the emergence and disappearance of topological features, such as connected components, loops, and voids, as we traverse through different scales of the complex. The persistence diagram, a tool of persistent homology, graphically records these topological features along with their persistence, providing a rich insight beyond traditional statistical methods.

The development and application of transformer-based machine learning models, exemplified by ESM-2 from the (FAIR) team at Meta AI, has seen significant strides in predicting protein structures from their sequences. Despite substantial gains in predictive performance, the interpretation of these models remains a formidable challenge due to the high-dimensionality and complexity of protein sequences and structures. In this study, we utilize persistent homology as a transformative lens for analyzing protein sequences and structures. We focus on extracting sequence and structural motifs, and other potentially biologically important subsequences, and we examine the preservation of their persistent homology across different contexts. In addition, we propose the inclusion of a topological loss term in the transformer model's loss function to encode this preservation as an inductive bias.

Given an embedding or representation of protein data, such as the attention probability distributions, context vectors, or hidden states of ESM2/ESMFold, into a potentially high-dimensional space, we construct a simplicial complex from the internal representation and compute its persistent homology. We do this using either the Jensen-Shannon distance for attention probability distributions, or the Eulidean distance for context vectors or hidden states, we construct the Vietoris-Rips complex, a filtered simplicial complex that encodes the multi-scale topology of the internal representation. We then obtain a simplex tree which can be used for extraction of candidate motifs and other potentially biologically important subsets of amino acids in the protein sequence. From this we then compute a persistence diagram that summarizes the multi-scale topological features of the internal representations of ESM-2 for proteins.

The resultant persistence diagram provides a comprehensive perspective on the local and global geometric and topological features of proteins, furnishing a robust method for protein comparison and clustering. Using a distance metric called the (p, q)-Wasserstein distance metric on the space of persistence diagrams, we can compare persistence diagrams and compute their Fréchet means. We further extend our approach to anomaly detection in protein sequences, employing the Fréchet

mean persistence diagram as a statistical reference baseline, which can be thought of as a kind of "*average of proteins*". This approach offers a novel, topological perspective on protein bioinformatics. We also provide methods for clustering proteins based on the Wasserstein distance of their persistence diagrams. Our study underscores the potential of topological data analysis, especially persistent homology, in elucidating complex patterns within protein sequence and structure data modeled by transformers. With this work, we hope to foster further advancements in the field of protein bioinformatics.

2. Related Work and Contributions

2.1. **Related Work.** Much of the related work focuses on applications to language as transformers were originally developed for NLP and are the backbone of modern Large Language Models. While there are a few examples of models which include persistent homology in the loss function, their approach is not using transformers, and require computing persistent homology of input data rather than of the internal representations of the model. In [KCMABBPPB] the authors study the attention maps from the perspective of weighted graphs, applying persistent homology to these weighted graphs to detect anomalous text. Similarly, the work [CTMPKABPPB] studies the role of the attention mechanism in encoding linguistic knowledge and the ability of single attention heads to judge the grammatical acceptability of a sentence using topological data analysis of the weighted attention graphs associated to text. Similarly, [PAP] investigates how transformer language models fine-tuned for acceptability classification capture linguistic features by constructing directed attention graphs from attention matrices, derive topological features from them and feed them to linear classifiers, which is becoming a common practice in the application of persistent homology to studying transformers. [TKKCBPNB-2] applies persistent homology to the analysis of the "embeddings" obtained from an transformer LLM to determine if a text is synthetic or not. [TKKCBPNB] applies topological data analysis to speech classification problems using HuBERT, where they apply TDA techniques to both the attention maps and the embeddings produced by HuBERT.

Now, consider a dialogue dataset consisting of a collection of dialogues, denoted as $\mathcal{D} = D_1, D_2, \ldots, D_n$, where each dialogue D_i is a sequence of utterances or sentences, represented as $D_i = u_1, u_2, \ldots, u_m$. The goal of dialogue term extraction is to extract a set of dialogue terms or *keywords*, denoted as $\mathcal{T} = t_1, t_2, \ldots, t_k$, from each dialogue D_i. [VHRNZG] uses topological data analysis to improve dialogue term extraction using large language models. [HFSSV] uses persistent homology techniques to construct *"story trees"*, which are meant to visualize prominent components of a story line, and which are somewhat similar in concept to the using simplex trees as a stand in for sentence parse trees, or morpheme trees.

Also somewhat related are [MHRB], which constructs a topological autoencoder using a regularization term in the loss function which uses persistence diagrams and some notion of distance such as the Bottleneck or Wasserstein distance between them. This is in line with some of the applications which are presented in the remainder of the article. We also mention [RCB], which constructs a transformer architecture that accepts persistence diagrams directly as input, rather than relying on various vectorization techniques. While we do not present any particular alterations to a transformer model to accept persistence diagrams, all of the ideas presented here are related to transformers in particular. The idea of using a topological loss function for certain vision tasks is presented [WAMMR] and [HFSC]. We also note the connection between [TKKCBPNB-2] and [M], the results of which are relevant to the current work. The former is considered with the Hausdorff dimension of the manifold representing data such as hidden states for language models for the purpose of identifying synthetic text, the latter is concerned with the same information for the purpose of understanding how the model learns, that is, identify when data is black-box generated vs. interpret why the black box learns the way it does.

Both can be considered as in the scope of the ideas presented here. Namely, we suggest persistent homology of internal representations as a tool for anomaly detection, motif discovery, protein sequence clustering, and as a method for analyzing the behavior of the model in order to suggest when and where a topological prior might be in order. In [RZSW], the authors develop a method for visualizing the topology of the embeddings of BERT using the mapper algorithm. For a thorough introduction to persistent homology and topological data analysis, we refer the reader to [DW], which has a chapter on machine learning, containing more references at the end of the chapter that may be of interest to the reader. We also note the survey [ZKNHRP], which gives an excellent overview of topological deep learning, and which has a section on the topological analysis of deep learning models, which is in line with much of the present article's content, as well as a section on topological loss functions.

For protein folding specific work the work of [LARHZLSVKSCZSCR] on ESM-2 and ESMFold are a central focus of this article. In particular this work focuses on analyzing these models, and using that analysis to motivate the definition of a topological loss term. There are a few applications of persistent homology in particular to protein folding in the literature, but none that focus on also using the transformer architecture, but those references that may be of interest to the reader can be found in [SL], a gentle introduction and survey of applications of persistent homology to biomedicine. We also mention the recent work of [W-SOCK], where multi-scale clustering using DBSCAN of protein sequences is used in conjunction with AlphaFold2 to predict multiple conformational states. Studying fold switching using persistent homology and DBSCAN along with ESM-2 provides analogous results with the added advantage of having a simpler architecture to AlphaFold2 and the more robust and descriptive information provided by persistent homology.

2.2. **Contributions.** The use of persistent homology in protein folding and modeling is often done by computing persistent homology of some molecular data which is then "vectorized", that is, given a form that can be given to a deep learning model. However, this is not our approach. Our approach is more about the internal consistency of the model's representation of the protein sequence. In particular, we advocate for the use of a topological loss term that encourage the model to have a consistent topological representation of sequential and structural motifs and protein sequences. Therefore, there is no need to compute topological features of the molecules or protein sequences themselves. We only need to compute the persistent homology of the internal representations of the protein sequences.

Different to our methods, most of the above methods work with either the L_1-distance for attention matrices (as apposed to the Jensen-Shannon distance metric), or they work at the level of "embeddings", which can mean several things in the literature. This could mean the *"context vectors"* obtained from an individual attention head, it could also mean the hidden states, output by an entire layer, or it could mean something in between, obtained after the layer normalization or the MLP. In our setup, we consider attention matrices (after softmax) to be a matrix with rows considered as probability distributions, to which we apply the Jensen-Shannon distance metric. We also use the Euclidean distance metric for context vectors $\mathbf{c}_i^{l,h}$, as well as for hidden states output by an entire layer $\mathbf{o}_i^l$ (which are the inputs of the next layer), all of which will be define in §3.

(1) We use persistent homology to analyze the three types of data mentioned above (attention probability distributions, context vectors, and hidden states).
(2) We provide a new benchmark for studying models using persistent homology, and discuss the preservation of persistent homology of motifs and other potentially biologically significant structures in the protein sequences under changes of context.

(3) We show that using persistence diagrams and a persistent homology informed DBSCAN we can perform a wide range of tasks such as clustering, motif extraction, anomaly detection, and model interpretation.
(4) We provide a new perspective on analyzing protein sequences using the persistent homology of the internal representations of ESM2/ESMFold, making suggestive connections to machine translation.
(5) We provide two topological loss functions, one for encouraging internal consistency of the model's topological representation of protein sequences, the other for a knowledge distillation procedure.
(6) We provide a way of comparing how well attention heads preserve persistent homology of motifs and other substructures.
(7) We provide a visualization tool for visualizing the topology of the internal representations of the models.
(8) We provide open source code to implement the above applications, which can be found at https://github.com/Amelie-Schreiber/transformers_proteins_and_persistent_homology/tree/main

It is important to note that when discussing the preservation of persistent homology of motifs and other potentially biologically significant structures as we change the protein they are located in (item 2 above), we are encouraging an internal consistency of the model. We are *not* comparing the persistent homology of the internal representation to some experimental ground truth as this may be too restrictive a constraint. This is analogous to encouraging machine translation language models to have a consistent internal representation of certain keyphrases, collocations, idioms, and multiword expressions (item 4 above). This is done through the introduction of the topological loss function. We also note that in our analysis we find that some models preserve persistent homology better than others. From this we idea, we derived a second topological loss function for knowledge distillation, where one model learns to mimic the topology of the internal representation of protein sequences of a teacher model.

3. Persistent Homology of Attention

The context vector is a critical part of the multihead attention mechanism in transformers. As indicated by the provided formula:

$$\mathbf{c}_i^{l,h} = \sum_{j=1}^{n} a_{i,j}^{l,h} \mathbf{v}_j^{l,h}$$

the context vector $\mathbf{c}_i^{l,h}$ for the h^{th} head in layer l is computed as a weighted sum of the value vectors $\mathbf{v}_j^{l,h}$, where the weights are the attention scores $a_{ij}^{l,h}$. Here, n is the number of value vectors, corresponding to the number of input tokens for the attention mechanism.

The attention scores $a_{ij}^{l,h}$ for attention head h in layer l, themselves are computed using the query and key vectors. In the scaled dot-product attention mechanism typically used in transformers, the attention score between the i^{th} query $\mathbf{q}_i^{l,h}$ and the j^{th} key $\mathbf{k}_j^{l,h}$ is computed as:

$$a_{ij}^{l,h} = \frac{\exp(\mathbf{q}_i^{l,h} \cdot \mathbf{k}_j^{l,h} / \sqrt{d})}{\sum_{m=1}^{n} \exp(\mathbf{q}_i^{l,h} \cdot \mathbf{k}_m^{l,h} / \sqrt{d})}$$

where d is the dimensionality of the queries and keys, and $\cdot$ denotes the dot product. The division by $\sqrt{d}$ is a scaling factor that is used to prevent the dot product from growing too large in magnitude, which could lead to vanishing gradients during training. The softmax function is applied to the raw

attention scores to ensure that they sum up to 1, allowing them to be interpreted as probabilities or relative importances.

The queries, keys, and values are themselves computed by applying learned linear transformations to the input embeddings. If $\mathbf{x}_i$ denotes the input embedding for the i^{th} token, then we have:

$$\begin{aligned}\mathbf{q}_i^{l,h} &= W_Q^{l,h}\mathbf{x}_i \\ \mathbf{k}_i^{l,h} &= W_K^{l,h}\mathbf{x}_i \\ \mathbf{v}_i^{l,h} &= W_V^{l,h}\mathbf{x}_i\end{aligned}$$

where $W_Q^{l,h}$, $W_K^{l,h}$, and $W_V^{l,h}$ are the weight matrices for the queries, keys, and values, respectively, for the h^{th} head in the l^{th} layer.

The context vectors $\mathbf{c}_{i,l}^h$ provide a summary of the input tokens, weighted by their relevance to the query. They can be thought of as a form of "contextualized" embedding, where the context is determined by the other tokens in the input sequence and their interaction weight given by the attention matrix $a_{ij}^{l,h}$ for head $h \in \{1, 2, ..., \mathcal{H}\}$. The multihead attention mechanism allows the model to capture different types of relevance or "attention" by using multiple heads, each with its own learned linear transformations.

In the multihead attention mechanism, the attention operation is not performed just once, but multiple times in parallel. The queries, keys, and values are transformed with different learned linear projections to $\mathcal{H}_l$ different sets of queries, keys, and values, where $\mathcal{H}_l$ is the number of heads in a layer of the transformer (and is typically constant for all layers). Then the attention mechanism is applied to each of these sets, yielding $\mathcal{H}_l$ output vectors, which are then concatenated and linearly transformed to result in the final output.

Let $W_Q^{l,h}$, $W_K^{l,h}$, and $W_V^{l,h}$ denote the weight matrices for the h^{th} head for the queries, keys, and values, respectively, in the l^{th} layer, and let W_O^l denote the output weight matrix for layer l. Then the output of the multihead attention mechanism for layer l is computed as:

$$\mathbf{c}_i^l = W_O^l[\mathbf{c}_i^{l,h}; \mathbf{c}_i^{l,h}; \ldots; \mathbf{c}_i^{l,\mathcal{H}_l}]$$

where $\mathbf{c}_i^{l,h}$ is the output of the attention mechanism for the h^{th} head in layer l, for token x_i, computed as:

$$\mathbf{c}_i^{l,h} = \sum_{j=1}^{n} a_{ij}^{l,h}\mathbf{v}_j^{l,h}$$

Each transformer layer comprises a multihead self-attention mechanism, followed by layer normalization, a position-wise feed-forward network, and another layer normalization. Let's denote the l^{th} transformer layer in the model, where l ranges from 1 to L (with L being the total number of layers).

(1) Multihead Self-Attention Mechanism: The multihead self-attention mechanism in the l^{th} layer operates on the input embeddings $\mathbf{x}_i^l$, transforming them into queries $\mathbf{q}_i^{l,h}$, keys $\mathbf{k}_i^{l,h}$, and values $\mathbf{v}_i^{l,h}$ for each head h as before. Then the context vectors are computed, and concatenated as before, and the weight matrix W_O^l is applied to get $\mathbf{c}_i^l$

(2) Layer Normalization: Layer normalization stabilizes the learning process and reduces internal covariate shift by normalizing the multihead self-attention output across the hidden dimension. For the l^{th} layer, this is computed as:

$$\mathbf{c'}_i^l = \frac{\mathbf{c}_i^l - \mu^l}{\sigma^l}$$

where μ^l and σ^l are the mean and standard deviation of the layer outputs, computed as:

$$\mu^l = \frac{1}{H}\sum_{i=1}^{H} c_i^l$$

$$\sigma^l = \sqrt{\frac{1}{H}\sum_{i=1}^{H}\left(c_i^l - \mu^l\right)^2}$$

(3) Position-Wise Feed-Forward Network (FFN): The output of the layer normalization is then passed through a position-wise feed-forward network (FFN). The

The output of the layer normalization is then passed through a position-wise feed-forward network (FFN). The FFN consists of two linear transformations with a ReLU activation in between. For each position i, the FFN is applied to $\mathbf{c}'^l_i$ independently. Let's denote the weight matrices and bias vectors of the two linear transformations as W_1^l, b_1^l, W_2^l, and b_2^l, respectively. Then the output of the FFN is computed as:

$$\mathbf{d}_i^l = W_2^l \mathbf{max}(0, W_1^l \mathbf{c}'^l_i + b_1^l) + b_2^l$$

where $\max(0, x)$ denotes the ReLU activation function.

(4) Second Layer Normalization: Finally, the output of the FFN goes through another layer normalization step to compute the final output of the transformer layer:

$$\mathbf{o}_i^l = \frac{\mathbf{d}_i^l - \mu^l}{\sigma^l}$$

where again, μ^l and σ^l are the mean and standard deviation of the layer outputs, computed similarly as before but now with $\mathbf{d}_i^l$ instead of $\mathbf{c}_i^l$.

3.1. **Persistent Homology.** In our analysis, we apply persistent homology and DBSCAN at three levels.

(1) To the attention probability distributions obtained from the row-wise softmax of the attention matrix $a_{ij}^{l,h}$, using the Jensen-Shannon distance metric (which might be substituted for KL-divergence, however this is not a genuine distance metric and thus this may be less desirable for persistent homology).
(2) To the context vectors $\mathbf{c}_i^{l,h} = \sum_{j=1}^{n} a_{i,j}^{l,h}\mathbf{v}_j^{l,h}$, using Euclidean distance.
(3) To the hidden states $\mathbf{o}_i^l$, output by an entire layer, using Euclidean distance.

We note that applying persistent homology to these three kinds of data yields varying results on the various applications we suggest. In the field of Natural Language Understanding, the concept of context vectors $\mathbf{c}_i^{l,h}$, as generated by models like Transformers, has proved to be critical in capturing the semantic and syntactic nuances of language. These context vectors, which provide a representation of each token in the input sequence in terms of its context, can be further processed to extract meaningful patterns and relationships. Here we will discuss the application of clustering algorithms, specifically Density-Based Spatial Clustering of Applications with Noise (DBSCAN), and its topological data analysis cousin, persistent homology, on these context vectors.

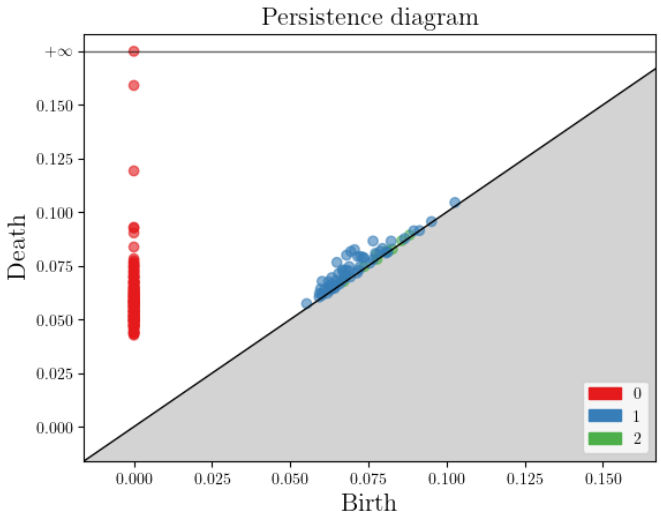
Given a Transformer model with L layers with $\mathcal{H}_l$ attention heads in layer l, for any token x_i in the input sequence, we can obtain a set of $\sum_{l=1}^{L}\mathcal{H}_l$ context vectors $\mathbf{c}_i^{l,h}$, with each context vector corresponding to a different attention head, in a different layer. For a given layer l and head h, the context vector $\mathbf{c}_i^{l,h}$ is a d-dimensional vector (where d is the dimensionality of the embeddings), which can be thought of as a point in a d-dimensional space. Thus, for all tokens in the input sequence, we can obtain a set of points in this d-dimensional space.

Clustering can be applied to this set of points to group together tokens that have similar context vectors, and hence, are likely to have similar semantic and syntactic roles. This can be useful for various applications, such as identifying synonyms, identifying words that often occur in similar contexts, or even for interpretability, by identifying what types of tokens each head and layer of the transformer is paying attention to. Similarly, we can apply clustering techniques to the probability distributions obtained from the softmax of the rows of the attention matrix, where we use the Jensen-Shannon distance. We can also apply persistent homology and other clustering techniques to the hidden states output by an entire layer of the transformer.

Persistent homology is a method within topological data analysis that quantifies the topological structure of a given dataset. It captures the evolution of topological features (like connected components, loops, etc.) as a function of a certain parameter (typically a scale parameter or "filtration" parameter). Persistent homology provides a multi-scale analysis which identifies these features at different scales, and tracks their "persistence" across these scales.

A key tool in persistent homology is the Vietoris-Rips complex. Given a point cloud (a set of points) in a metric space and a scale parameter $\epsilon \geq 0$, the Vietoris-Rips complex VR_ϵ is a simplicial complex where a k-simplex is formed by $(k+1)$ points that are pairwise within distance ϵ. That is, a set of $k+1$ points forms a k-simplex if every pair of points in the set is within ϵ distance of each other. A filtration of Vietoris-Rips complexes is obtained by letting ϵ increase from 0 to ∞. For small ϵ, VR_ϵ is a collection of isolated points. As ϵ increases, edges start to appear, forming 1-simplices (joining pairs of points that are within ϵ), then 2-simplices (triangles), 3-simplices (tetrahedra), and so on.

The output of persistent homology is often represented by a persistence diagram. A persistence diagram is a multi-set of points in the plane, where each point (b, d) represents a topological feature (e.g., a connected component, a loop) that is "born" at scale b and "dies" (disappears) at scale d. In other words, the feature exists in the interval $[b, d)$ of the filtration. The persistence of a feature is given by $d - b$, which indicates how long a feature exists in the filtration. A persistence diagram can have multiple components, which correspond to the dimension of the homology features. That is, they may have components for H_0 (connected components or clusters), H_1 (loops like triangles given by three or more edges), H_2 (two-dimensional holes in the data), and so on. The number of tokens gives an upper bound on the highest dimensional H_i.



Comparing persistence diagrams is often achieved by using the bottleneck distance or the q-Wasserstein distance. The bottleneck distance d_B between two persistence diagrams D_1 and D_2 is the infimum over all bijections $\sigma : D_1 \to D_2$ of the supremum distance over all points in the diagrams, formally defined as

$$d_B(D_1, D_2) = \inf_{\sigma} \sup_{x \in D_1} \|x - \sigma(x)\|_\infty.$$

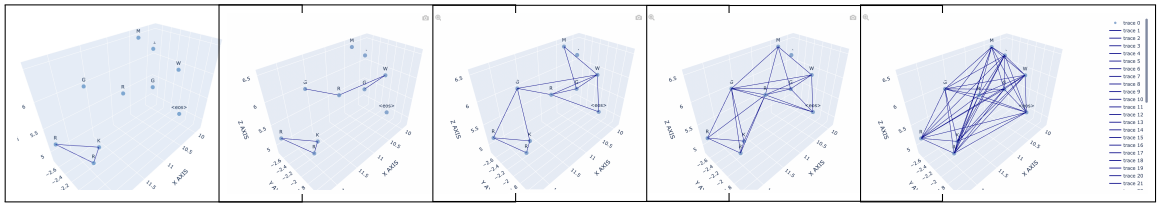
Intuitively, the bottleneck distance matches points in one diagram to points in the other diagram, and the bottleneck distance is the greatest distance one has to move a point to match it to a point in the other diagram. The q-Wasserstein distance between two persistence diagrams D_1 and D_2 is another way of comparing diagrams. The q-Wasserstein distance is the infimum over all bijections $\sigma : D_1 \to D_2$ of the q-th root of the sum of the q-th powers of the distances between matched points. Formally,

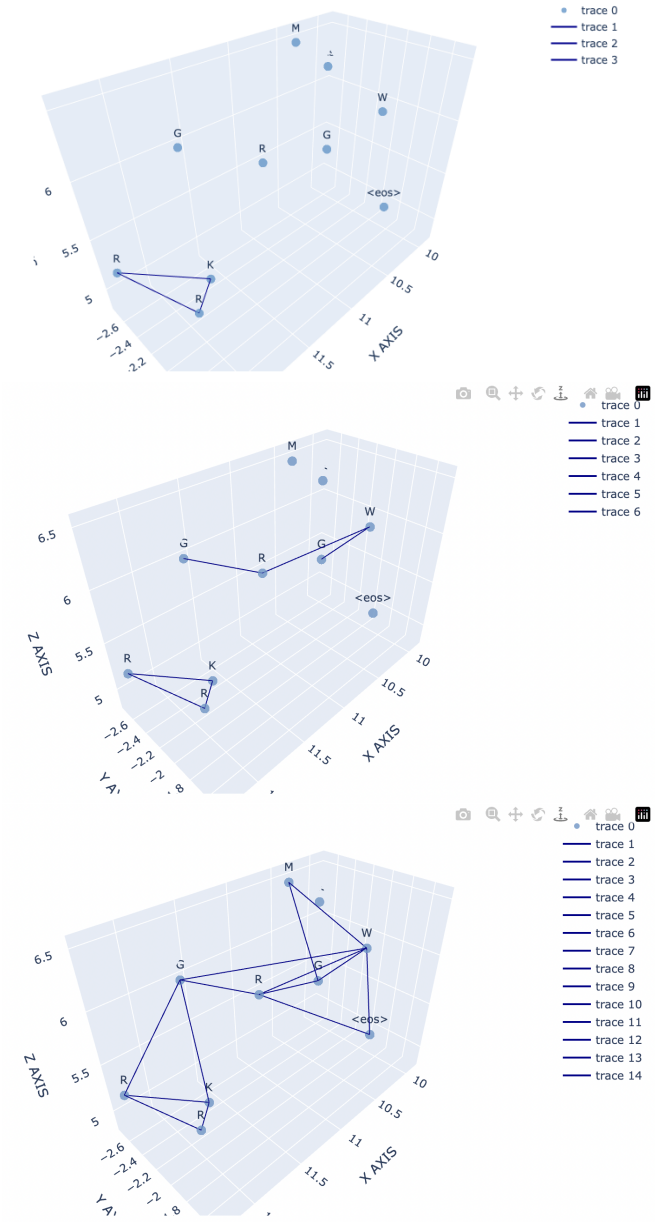
$$d_q^{\mathcal{W}}(D_1, D_2) = \left(\inf_{\sigma} \sum_{x \in D_1} \|x - \sigma(x)\|^q \right)^{1/q}.$$

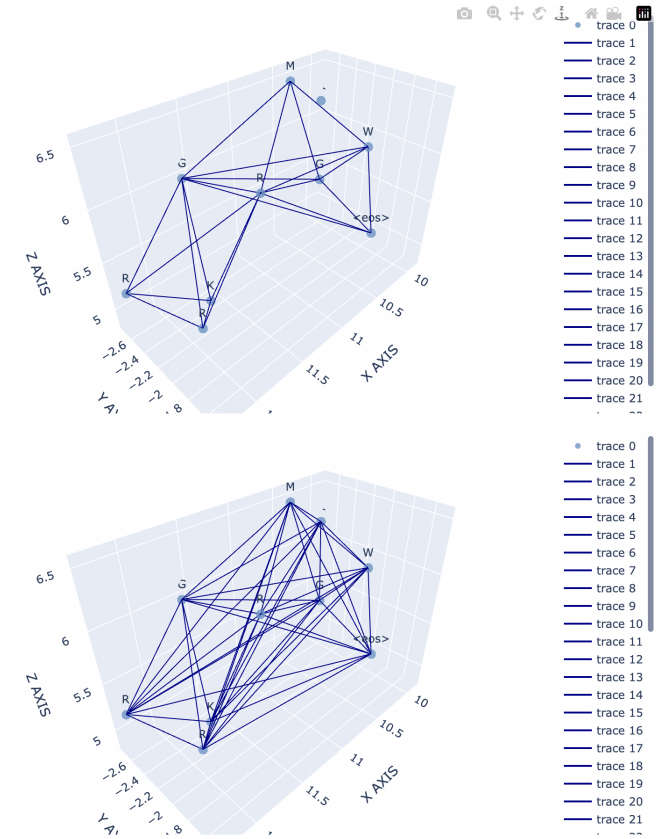
The q-Wasserstein distance tends to give a more nuanced measure of the difference between persistence diagrams than the bottleneck distance, especially for larger q, as it takes into account all distances between matched points, rather than only the largest one. These distance metrics give rigorous ways of comparing the topological structures of different datasets, and have applications in various fields such as shape analysis, image processing, and machine learning.

4. Visualizing Persistent Homology

We can visualize the persistent homology of the attention probability distributions, context vectors, and hidden states using a tool we developed. The model is first given some input protein sequence. From this we either choose a layer and a head to compute the attention probability distributions or context vectors, or we choose a layer to compute the hidden states. Once we have computed these, we apply persistent homology which gives us a simplex tree. This simplex tree tells us at which distances vertices become connected to each other and in what order. From this, we get a simplicial complex at each fixed filtration value. We can then apply a UMAP to the 1-skeleton, that is, to the edges that are formed, and plot this 1-skeleton, which is just a graph with nodes representing the tokens of the input protein sequence. As an example, take the following protein sequence: *"MGWGRKRR"*, and observe how more and more of the vertices become connected as we increase ϵ.




trace 0
trace 1
trace 2
trace 3
trace 4
trace 5
trace 6
trace 7
trace 8
trace 9
trace 10
trace 11
trace 12
trace 13
trace 14
M
W
G
R
K
<eos>
X AXIS
Z AXIS
Y AXIS



We note that this visualization does not always match the visualization of the 3D-folded structure predicted by the model. This could be partially due to the use of UMAP, but it also indicates that the 3D structure of the internal representation of the data simply may not follow our intuitions. The image in Figure 1 is a good example of when UMAP could be considered as the problem. In this image, we can see some of the structure reflected in the point cloud, but with points that are far away connected to one another. Sometimes, simply running the visualization code again to obtain a new UMAP model of the point cloud yields better results. Other times, we find that obtaining a close match to the 3D-folded structure predicted by ESM-2/ESMFold simply may not be possible. We also find that smaller protein sequences such as "MGWGRKRR" mentioned previously do not have enough data points to have complex enough 3D-structure to mirror the 3D-folded structure that is predicted by the model.

5. Motifs in Protein Sequences and Structures

Proteins are the building blocks of life and are responsible for a multitude of tasks that facilitate life processes. Understanding the properties of protein sequences and their corresponding folded structures can provide important insights into their functions. This is especially the case for recurring patterns or "motifs" that are evolutionarily conserved. In this section, we explore the concepts of sequence motifs and structural motifs in proteins.

5.1. **Sequence Motifs.**

A sequence motif in proteins is a pattern of amino acids that has biological significance and is often conserved across different proteins or within protein families. They are

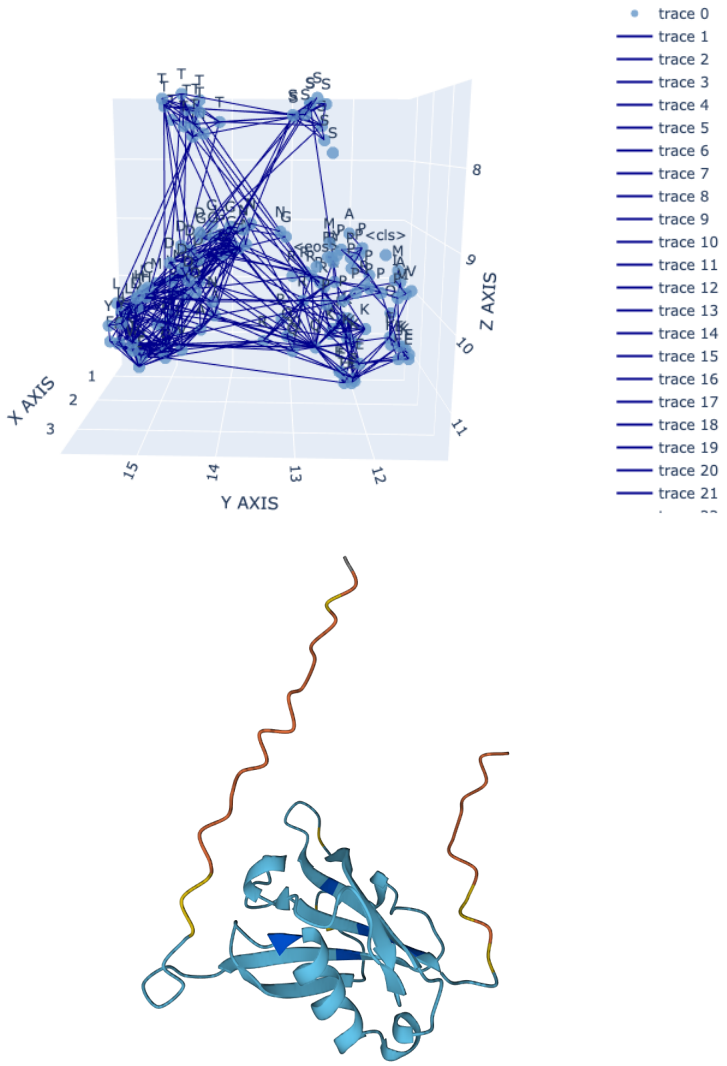


FIGURE 1. UMAP of 1-Skeleton and Folded Protein
MAVESRVTQEEIKKEPEKPIDREKTCPLLLRVFTTNNGRHHRMDEFSRGNVPSSELQIYTWMDATLKELTSLVKEV YPEARKKGTHFNFAIVFTDVKRPGYRVKEIGSTMSGRKGTDDSMTLQSQKFQIGDYLDIAITPPNRAPPPSGRMRPY

associated with specific functions or properties and are found in different organisms across different taxa. A sequence motif is generally represented as a regular expression or a position-specific scoring matrix (PSSM), which statistically describe the probabilities of individual amino acids occurring at each position within the motif.

A motif M of length n can be represented as a string of amino acids, i.e., $M = a_1a_2...a_n$, where $a_i \in \{A, R, N, D, C, Q, E, G, H, I, L, K, M, F, P, S, T, W, Y, V\}$ represents the i^{th} amino acid in the motif. For instance, a well-known motif is the ATP-binding motif, which is represented as "GxGxxG", where "x" can be any amino acid.

The PSSM representation, on the other hand, is a matrix $P = [p_{ij}]$ of size $20 \times n$, where p_{ij} represents the probability of amino acid i occurring at position j in the motif. Given a protein sequence $S = s_1 s_2 ... s_m$ and a sequence motif M of length n, one can locate the motif in the sequence by sliding a window of size n over the sequence and checking for a match or a significant alignment score.

5.2. **Structural Motifs.** Structural motifs in proteins refer to conserved three-dimensional arrangements of secondary structural elements such as alpha-helices and beta-strands. They often represent functionally or structurally important parts of the protein, such as the active site of an enzyme or a protein-protein interaction interface. A structural motif is generally described in terms of the geometric arrangement of its constituent secondary structural elements and their connectivities. It is also common to specify the relative orientations and distances between the secondary structural elements.

Given a protein structure represented as a sequence of C_α atoms $P = (x_1, y_1, z_1), (x_2, y_2, z_2), ..., (x_k, y_k, z_k)$, where (x_i, y_i, z_i) denotes the coordinates of the i^{th} C_α atom, and given a structural motif represented as a set of key C_α atoms with their relative geometric arrangement, one can locate the motif in the protein structure by finding substructures in P that match the geometric arrangement of the motif, within a certain tolerance.

5.3. **Importance of Motifs.** Both sequence motifs and structural motifs are of fundamental importance in the study of proteins. Sequence motifs, often located in functionally important regions, can help predict the function of uncharacterized proteins or the impact of mutations. They can also aid in the design of proteins with novel functions. Structural motifs, on the other hand, provide crucial insights into the functional mechanisms of proteins, including protein-protein interactions, enzyme activities, and allosteric regulations. Understanding structural motifs can inform the development of drugs targeting specific protein structures, or the design of proteins with desired structures and functions.

Mathematically, identifying sequence motifs in a given protein sequence is essentially a pattern-matching problem, while finding structural motifs in a protein structure can be seen as a subgraph isomorphism problem in the protein's secondary structure graph representation. Understanding and identifying sequence and structural motifs in proteins is a key aspect of bioinformatics, structural biology, and functional genomics. These motifs not only provide insights into the biological functions and evolutionary history of proteins, but also serve as valuable tools for protein design and drug discovery.

6. Sequence Motifs from the Simplex Tree

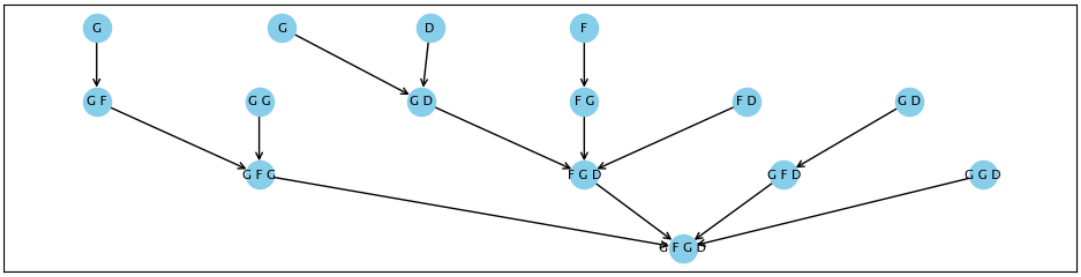
A simplex tree is a data structure designed to efficiently store a simplicial complex. Simplicial complexes are mathematical objects composed of vertices, edges, triangles, tetrahedrons, and their n-dimensional counterparts, collectively referred to as simplices. In a simplex tree, simplices are organized hierarchically according to their inclusions. Let K be a simplicial complex with vertices from a totally ordered set. A simplex tree T representing K is a rooted tree such that:

Each node N in T corresponds to a simplex σ_N in K. The root of T corresponds to the empty simplex. If node N' is a child of node N in T, then $\sigma_{N'}$ is a face of σ_N in K and $\sigma_{N'} \neq \sigma_N$. Each node N in T has a label, which is the largest vertex in σ_N. Given this definition, any path from the root of T to a node N corresponds to an ordered list of vertices of σ_N in increasing order. For instance, in a 2-simplex (triangle) with vertices $1, 2, 3$, the root leads to three child nodes, one for each vertex. Each vertex node has child nodes for edges that include the vertex, and each edge node has a child node for the triangle.

The Gudhi library implements the simplex tree data structure with functionalities for insertion and removal of simplices, navigation through the tree, and retrieval of simplex information, among other operations. Below, we give the simplex tree for the protein "GFGD".

TABLE 1. List of simplices with corresponding amino acids and filtration values

Simplex	Amino Acid	Filtration Value
[0]	G	0.0
[1]	F	0.0
[2]	G	0.0
[3]	D	0.0
[2, 3]	G D	0.08967289096441512
[1, 2]	F G	0.18783188189388525
[1, 3]	F D	0.2467492439611421
[1, 2, 3]	F G D	0.2467492439611421
[0, 1]	G F	0.30936620794721315
[0, 2]	G G	0.47987940339650703
[0, 1, 2]	G F G	0.47987940339650703
[0, 3]	G D	0.5171996267298892
[0, 1, 3]	G F D	0.5171996267298892
[0, 2, 3]	G G D	0.5171996267298892
[0, 1, 2, 3]	G F G D	0.5171996267298892



Now, if we want to detect potential motifs of a particular length, we can look at the simplex tree for simplices with the same number of vertices as the length of our sequence motif. In the context of a protein sequence, a simplex in the simplex tree can represent a subset of the amino acids which may not be in a consecutive order in the sequence but have a strong association, as determined by the filtration value. This association could be due to the physical properties of the amino acids, their positions in the sequence, or other biological factors. The filtration value can then be viewed as a measure of the strength of this association.

A simplex with a low filtration value signifies a stronger association between the represented amino acids, and this association could indicate a potential motif. Hence, the process of exploring the simplex tree remains similar: we start with simplices of the lowest filtration values and ascend the filtration value to find potential motifs. To formalize this process, we can define a filtration function $f : \mathcal{K} \to \mathbb{R}_{\geq 0}$ that assigns to each simplex σ in the simplicial complex $\mathcal{K}$ a real number $f(\sigma)$, known as the filtration value of σ. This function should satisfy the property that if τ is a face of σ, then $f(\tau) \leq f(\sigma)$. This means that subsets of strongly associated amino acids will also be strongly associated.

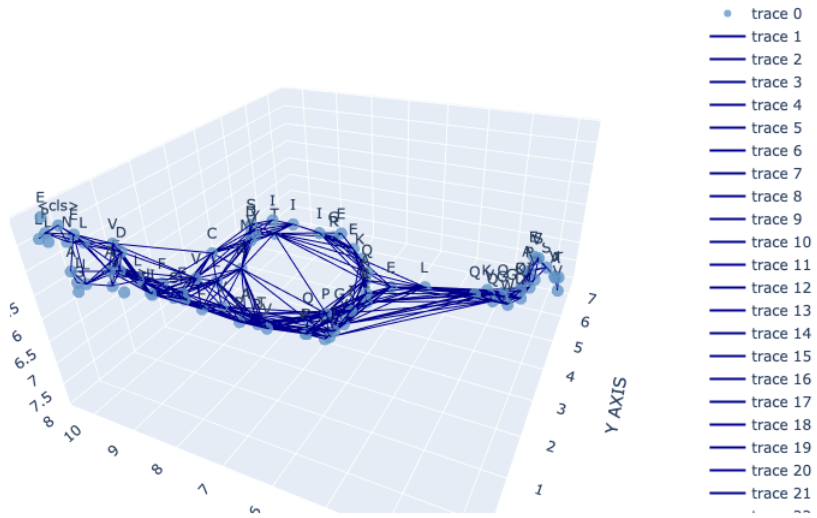


FIGURE 2. UMAP of 1-Skeleton for Fixed ϵ

The set of all simplices whose filtration value is less than or equal to some $r \in \mathbb{R}_{\geq 0}$ forms a subcomplex $\mathcal{K}_r = \sigma \in \mathcal{K} \mid f(\sigma) \leq r$. By adjusting the threshold r, we can control the level of association required for a set of amino acids to be considered a potential motif. For example, setting r to be the lowest filtration value will yield the most strongly associated sets of amino acids.

The structure of the simplex tree allows us to efficiently traverse the simplices in increasing order of their filtration values. We start with the empty simplex (the root of the tree), which has the lowest possible filtration value, and proceed to its children. Since the children of a node in the simplex tree are its faces, this ensures that we explore all faces of a simplex before the simplex itself. The advantage of this approach is that it allows us to detect motifs that involve non-consecutive amino acids and are defined by more complex associations than simple proximity in the protein sequence. By adjusting the filtration function and the threshold, we can adapt this method to various types of motif detection tasks.

7. Structural Motifs from UMAP of Simplicial Complexes and the Simplex Tree

Now, the above method of searching for sequence motifs using the simplex tree also comes with a geometric component which can provide us with potential structural motifs as well. In particular, we can find sub-simplicial complex of the filtered Rips-complex that form geometric structures such as loop or handles. As an example, observe the following plot (2) of the protein sequence "MVPLCQVEVLYFAKSAEITGVRSETISVPQEIKALQLWKEIETRHPGLADVRNQI-IFAVRQEYVELGDQLLVLQPGDEIAVIPPISGG". In particular, this is obtained from UMAP applied to the 1-skeleton of the simplicial complex obtained by setting the distance parameter to $\epsilon = 0.42$, for layer 3, head 2 of *'facebook/esm2_t6_8M_UR50D'*:

Notice the central handle that makes the context vectors have the structure of a torus with two stem like structures. Looking at simplices corresponding to the loop gives us subsets of amino acids within the protein sequence that could be structural motifs. Notice how the persistent homology of the context vectors above mimics the $3D$-structure of the protein itself (see Figure 3) as predicted by ESM2/ESMFold.

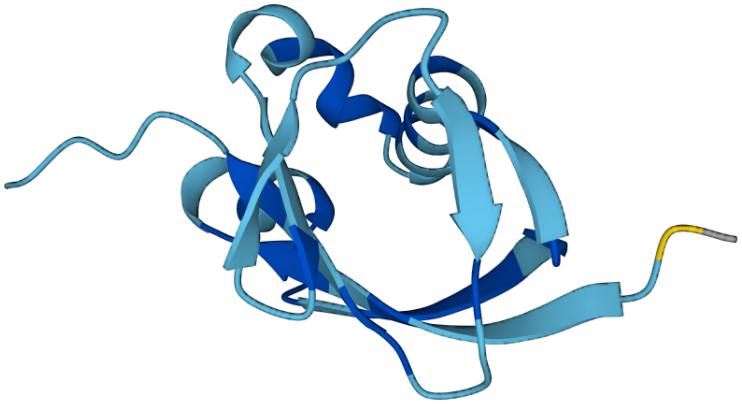


FIGURE 3. *MVPLCQVEVLYFAKSAEITGVRSETISVPQEIKALQLWKEIETR HPGLADVRNQIIFAVRQEYVELGDQLLVLQPGDEIAVIPPISGG*

The unique structure identified by persistent homology in the context vectors, mirrored in the 3D structure of the protein, offers a compelling indication of potential structural motifs. The central loop or handle structure, captured by a sub-simplicial complex of the filtered Rips-complex, suggests a recurrent theme in the protein's 3D conformation. To understand the biological implications of these structural motifs, let's consider the simplices corresponding to the central loop in the UMAP projection. These simplices represent amino acids that are may be non-consecutive, but often are consecutive subsequences in the protein sequence and are close in the embedding space. This closeness could be a consequence of them being near in the 3D structure or sharing similar structural roles. Therefore, the amino acid subsets corresponding to these simplices could represent structural motifs.

To further investigate these potential structural motifs, one could map back the subsets of amino acids to their corresponding positions in the protein sequence and the 3D structure. These positions could then be analyzed in terms of their structural and functional roles. One could also explore whether these subsets of amino acids occur recurrently in other protein structures. If they do, it would provide additional evidence that these subsets indeed represent structural motifs. Moreover, the fact that the persistent homology of the context vectors resembles the actual 3D structure of the protein is noteworthy. This suggests that the embeddings learned by *facebook/esm2_t6_8M_UR50D* capture structural information of the proteins, indicating the robustness of transformer models in learning intricate biological structures. This representation could serve as a valuable tool for understanding and predicting protein structures from sequence alone.

By studying the geometric structure in the context vector space revealed by UMAP and persistent homology, we can identify potential structural motifs in proteins. These structural motifs, represented by simplices in the simplex tree, could provide important insights into the functional roles of different regions in the protein structure. This approach opens a new avenue for studying protein structures, complementing traditional methods based on sequence alignment and manual examination of 3D structures.

8. Persistent Homology Informed DBSCAN for Substructure Identification

8.1. **Attention Computation.** To better understand the above methods, let us look at an example. Given an input protein sequence $\boldsymbol{x} = (x_1, \ldots, x_n)$ of length n, for example

MAVESRVTQEEIKKEPEKPIDREKTCPLLLRVFTTNNGRHHRMDEFSRGNVPSSELQIYTWMDATLKELTSLVKEV
YPEARKKGTHFNFAIVFTDVKRPGYRVKEIGSTMSGRKGTDDSMTLQSQKFQIGDYLDIAITPPNRAPPPSGRMRPY,

which is of length 153, we obtain the attention outputs from the ESM-2 model. Specifically, for layer l and head h, we compute:

$$\mathbf{O}^{(l,h)} = \text{Attention}(\mathbf{x}; \theta)$$

where θ denotes the parameters of the ESM-2 model. The attention computation follows the standard transformer self-attention:

$$\mathbf{O}^{(l,h)} = \mathbf{softmax}(\mathbf{S}^{(l,h)})\mathbf{V}^{(l,h)}$$

where

$$\mathbf{S}^{(l,h)} = \left(\mathbf{W_Q}^{(l,h)}\mathbf{x}\right)\left(\mathbf{W_K}^{(l,h)}\mathbf{x}\right)^T$$

$$\mathbf{V}^{(l,h)} = \mathbf{W_V}^{(l,h)}\mathbf{x}$$

are the token similarity matrix and value representations respectively, and $\mathbf{W}_Q^{(l,h)}$, $\mathbf{W}_K^{(l,h)}$, $\mathbf{W}_V^{(l,h)}$ denote the query, key and value projections for head h in layer l. Thus, the output $\mathbf{O}^{(l,h)} \in \mathbb{R}^{n\times d}$ is a sequence of n token representations in $\mathbb{R}^d$.

8.2. **Persistent Homology.** We apply topological data analysis using persistent homology to characterize the geometry and topology of the attention output. First, we compute the pairwise Euclidean distance matrix $\mathbf{D} \in \mathbb{R}^{n\times n}$ between the output token representations:

$$D_{ij} = \|\mathbf{c}_i - \mathbf{c}_j\|_2$$

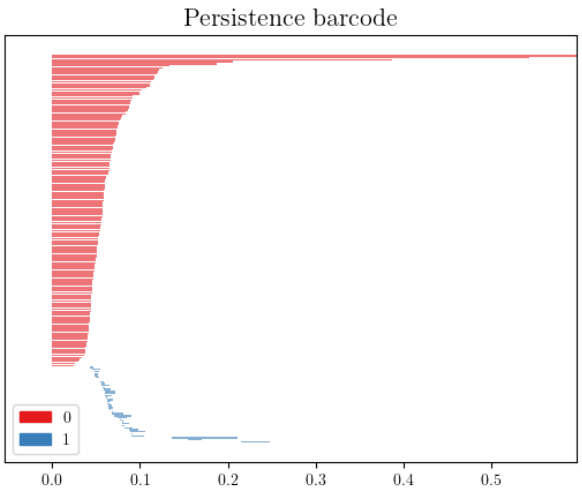
where $\mathbf{c}_i, \mathbf{c}_j \in \mathbb{R}^d$ (or more precisely $\mathbf{c}_i^{(l,h)}, \mathbf{c}_j^{(l,h)} \in \mathbb{R}^d$) are attention output rows of $\mathbf{O}^{(l,h)} \in \mathbb{R}^{n\times d}$[1], prior to layer normalization of the MLP. We construct a Vietoris-Rips simplicial complex on this distance matrix.[2] The Vietoris-Rips complex at scale ϵ connects vertices whose pairwise distance is at most ϵ:

$$\mathbf{Viet}(\mathbf{D}; \epsilon) = \{\sigma \subseteq [n] : D_{ij} \leq \epsilon, \forall i, j \in \sigma\}$$

We compute the persistent homology $H_k(\mathbf{Viet}(\mathbf{D}; \epsilon))$ for $k = 0, 1, 2$ across scales ϵ. The persistence barcode diagram provides a concise visualization of the appearance and disappearance of topological features like connected components, loops, voids across scales.

[1] These are the "*context vectors*" defined previously in §3.

[2] We end up visualizing a $3D$ version of the 1-skeleton of this simplicial complex using UMAP. The 1-skeleton of the simplicial complex is simply a graph, with nodes representing amino acids.



This gives us information on what scale parameter ϵ to use for our DBSCAN in the next step. In particular, we choose a value that either (1) captures a large portion of the red bars, or (2) captures low-lying H_1 features, which corresponds to cycle like structures in the simplicial complex. The H_1 (blue bars) features in the persistence barcode diagram correspond to 1-dimensional holes or loops in the underlying data. More specifically, geometrically, a loop feature captures a cycle or circular structure in the data. Topologically, it represents a non-trivial 1-dimensional hole. In the persistence diagram, H_1 features are visualized as horizontal line segments. The x-coordinate of the left endpoint indicates the birth time - when the loop first appears in the Vietoris-Rips filtration. The x-coordinate of the right endpoint indicates the death time - when the loop gets filled in and disappears. The length of the line segment represents the persistence of the loop over geometric scales. Longer bars indicate more prominent and persistent loop features, while short-lived loops correspond to noise. Looking at the ensemble of H_1 bars provides insight into cycles present in the data at different scales. The number and persistence distribution of loops gives a topological signature of the geometry. We might also use H_2 features, which described the presence of "voids" in the data, or higher H_k. However, H_0 and H_1 are often sufficient and higher H_k become computationally intractible for large k. Moreover, the higher H_k become less interpretable and we lose our geometric intuition.[3]

8.3. **Dimensionality Reduction and Clustering.** To enable interactive visualization and exploration, we reduce the dimension of the attention output to $3D$ using UMAP:

$$\mathbf{Z} = \text{UMAP}(\mathbf{O}^{(l,h)}) \in \mathbb{R}^{n\times 3}$$

We also apply density-based spatial clustering DBSCAN on the distance matrix $\mathbf{D}$ to cluster proximate tokens:

$$C = \text{DBSCAN}(\mathbf{D}, \epsilon, k)$$

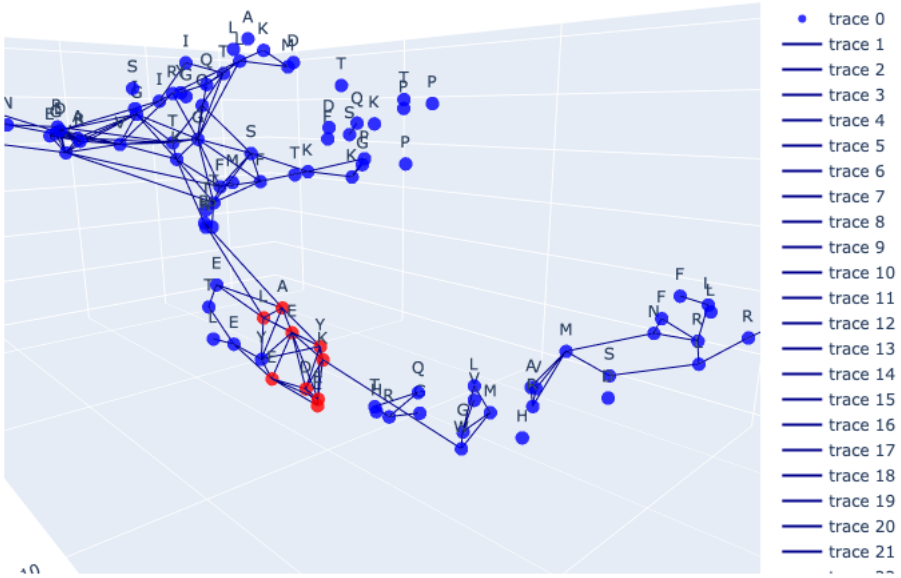
where $C = \{C_1, \ldots, C_m\}$ is the clustering containing m clusters, based on distance threshold ϵ and minimum cluster size k. In our example, the clusters are:

[3]If it is found that higher dimensional H_k are important for identifying important substructures, we will likely need to turn to quantum computing techniques such as those in [H]. However, it is important to note that while these methods show quantum advantage in some cases, it is not clear how applicable this will be to our current analysis, or if there will be improvements in dequantization that will remove this advantage.

Cluster 0	E, S
Cluster 1	I, P
Cluster 2	K, K
Cluster 3	F, R, N
Cluster 4	T, N
Cluster 5	G, R, Q
Cluster 6	R, M, V, A
Cluster 7	E, E, L, Y, D, K, L, E, A
Cluster 8	G, W, M, L
Cluster 9	I, S, K, V
Cluster 10	E, L, T
Cluster 11	R, K
Cluster 12	F, A, V, F, T, D, V, R, G, R, K, E, T, M, S, G, R, G, Q, I
Cluster 13	F, V, G
Cluster 14	T, L, K, Q, I
Cluster 15	A, M

TABLE 2. Identified clusters of amino acids in the protein sequence.

The identified clusters reveal biologically meaningful motifs and domains, like α-helices and β-sheets, and other substructures and subsets of amino acids that may have important biological activity. Finally, we visualize the 3D output representations $\mathbf{Z}$ as an interactive plot, with edges connecting tokens within distance thresholds. This enables exploring local neighborhoods in the attention geometry. To visualize the substructure corresponding to a clutser, we simply compute its positions, and give this to the plotting function. For example, the positions of the tokens in cluster 7 of the protein sequence are: $[45, 55, 56, 59, 63, 67, 72, 75, 80]$.



MAVESRVTQEEIKKEPEKPIDREKTCPLLLRVFTTNNGRHHRMDEFSRGNVPSSELQIYTWMDATLKELTSLVKEV
YPEARKKGTHFNFAIVFTDVKRPGYRVKEIGSTMSGRKGTDDSMTLQSQKFQIGDYLDIAITPPNRAPPPSGRMRPY

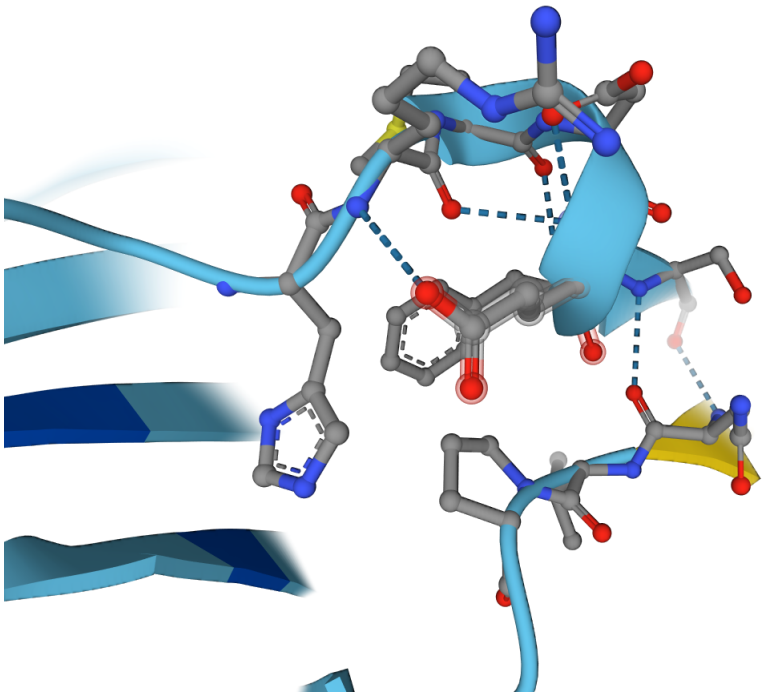


FIGURE 4. First "E" in the Amino Acid Cluster 7

We can then search the $3D$ folded structure given to us by ESMFold to see what amino acids these subsets of amino acids correspond to and where they are located in the protein itself. For example, below we see the first E in cluster 7 in Figure 4. Observing where cluster 7's amino acids are, we find that they seem to correspond to a transitional structure, moving from the last strand of a β-sheet, through an intrinsically disordered region (IDR), and through an α-helix.

We can visualize the entire cluster by highlighting the corresponding residues in pink (see Figure 5).

Overall, this demonstrates how topological and geometric analysis can provide mathematical rigor and visual insights into understanding the latent structure learned by self-attention models on protein sequences.[4]

9. Preserving Persistent Homology of Motifs

Now, suppose we have a sequential or structural motif inside of a protein sequence that recurs in several other proteins. We can compute the persistent homology of this subsequence in each context. In other words, we can compute the persistent homology of this subsequence or subset of amino acids in each protein sequence that it appears in. If we have a subsequence we know has a relatively stable topology in multiple protein sequences, we may wish to enforce this prior. To that end, comparing the persistence diagrams that we obtain using the Wasserstein distance, we obtain a summary of how well the persistent homology is preserved by the model in different contexts,

[4]The methods mentioned in this section can be found in the following notebook.

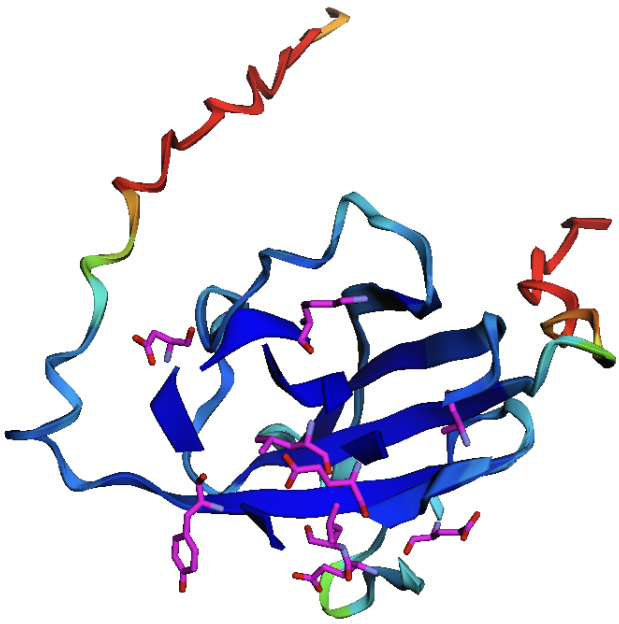


FIGURE 5. Cluster 7 Residues in Pink

that is, in different proteins. This gives us a measure of how much the persistent homology of the subset is preserved by the model, and can be used to compare heads within the model or of different models (even if the attention heads have a different number of parameters). If the motif is a structural motif, it is important that the model preserve the persistent homology well in all contexts. It may be similarly important to preserve the persistent homology of sequence motifs as well.

Given a motif, whether sequential or structural, we can compute the persistent homology for the motif in every protein sequence where it occurs. More concretely, given a protein sequence, we extract the context vectors corresponding to the motif, construct a simplicial complex (such as the Vietoris-Rips complex), and then compute the persistent homology. The output is a persistence diagram, which encodes the birth and death of topological features at different scales of filtration.

Recall, the persistence diagram is a multiset of points in the plane, where each point (b, d) represents a topological feature (e.g., connected component, loop, void) that is born at filtration value b and dies at filtration value d. The persistence of a feature is given by $d - b$, and long-lasting features (i.e., those with high persistence) are typically of more interest as they represent stable structural elements. To quantify the similarity between two persistence diagrams, we can employ the Wasserstein distance, which is a well-established metric in the field of optimal transport. Formally, given two persistence diagrams D_1 and D_2, the (p, q)-Wasserstein distance is defined as:

$$d_{p,q}^{\mathcal{W}}(D_1, D_2) = \left(\inf_{\gamma \in \Gamma(D_1, D_2)} \sum_{(u,v) \in \gamma} \|u - v\|_q^p \right)^{1/p}$$

where $\Gamma(D_1, D_2)$ denotes the set of all bijections $\gamma : D_1 \cup \Delta \to D_2 \cup \Delta$, and Δ denotes the diagonal line $y = x$ in the plane. The term $\|u - v\|_q$ denotes the L^q norm of the vector from u to v. For historical

reasons,$q = \infty$ in the majority of applications. It is often more reasonable to set $q = p$ to control the geometry of the space of persistence diagrams [TMMH]. By computing the Wasserstein distance between the persistence diagrams derived from different contexts of the same motif, we obtain a measure of the consistency of the motif's persistent homology across different contexts. A low Wasserstein distance would suggest that the model effectively preserves the topological features of the motif across different proteins. Understanding how well these topological features are preserved can provide crucial insights into the structural or functional roles of these motifs, and potentially highlight the model's capability in capturing these recurrent topological patterns.

10. Topological Loss Terms

10.1. **Topological Inductive Biases.** Now, since preserving persistent homology of the attention probability distributions, context vectors, and hidden states of sequence and structural motifs across different protein sequences, that is, across different contexts can be an important property to include in the model we would like to devise a way to include invariance of persistent homology of motifs as an inductive bias in the model. We can do so by including a topological term in the loss function using the Wasserstein distance. Our intention is to add a term to the loss function that enforces the persistent homology of the model's outputs to be invariant across different instances of a given motif. This can be achieved by adding a regularization term that discourages large differences in the persistence diagrams associated with different instances of the same motif in different protein sequences. Given two persistence diagrams D_1 and D_2, the Wasserstein distance $d_{p,q}^{\mathcal{W}}(D_1, D_2)$ can be used as a measure of the difference between the two diagrams. Hence, our topological loss term could be given by the sum of Wasserstein distances over all pairs of persistence diagrams associated with the same motif. If $D_1, D_2, ..., D_m$ denote the persistence diagrams corresponding to a given motif across m different protein sequences, then the topological loss can be written as:

$$\mathcal{L}_{\text{topo}} = \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} d_{p,q}^{\mathcal{W}}(D_i, D_j).$$

Therefore, the total loss function $\mathcal{L}$, which includes the standard loss term $\mathcal{L}_{\text{task}}$ (e.g., mean-squared error, cross-entropy, etc., depending on the specific task) and the topological loss term $\mathcal{L}_{\text{topo}}$, could be written as:

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda \mathcal{L}_{\text{topo}},$$

where λ is a hyperparameter that controls the trade-off between the standard loss term and the topological loss term. This parameter should be chosen carefully, as a large value of λ could lead to over-emphasis on the topological invariance at the expense of the task-specific objective, and vice versa. By adding this topological loss term, we effectively incorporate a form of inductive bias that encourages the model to produce outputs with similar topological features across different instances of the same motif. This could potentially enhance the model's ability to capture and generalize the inherent structural patterns of protein sequences.

Another approach would be to use a model which has a very consistent persistent homology for the motif across a set of protein sequences, compute Frèchet mean diagram across the set of protein sequences, and then use this as a baseline. This is effectively a knowledge distillation process. We will discuss this in §11

10.2. **Relations to Group Equivariance: An Argument for Relative Positional Encodings.** We note that encouraging the model to maintain a certain level of internal consistency in the topology of the internal representations (attention probability distributions, context vectors, or hidden states) is a

topological inductive bias that is somewhat similar to, but less rigid than the constraint of equivariance. As geometrically and mathematically appealing as equivariance to a group action is, it is likely inappropriate to include it in a model for protein sequence related tasks. The reasons for this are twofold. Firstly, the inductive bias of equivariance to a group action does not allow the model to recognize properties of molecules such as chirality and fold-switching, where conformational changes are allowed. Moreover, it is often necessary to modify the transformer architecture in complicated ways. While chiral molecules may have left and right handed enantiomers that have *different geometry*, their *topology* remains the same. Thus, a topological loss term does not impose a too rigid a constraint, and it gives a much simpler way of including the inductive bias.

We must point our that simple translation equivariance should not cause any problems with chirality, and that this can be included in a transformer using relative positional encoding as described in [RC] without further complicated model architecture modifications. This can be further paired with Low Rank Adaptations (LoRAs) for modifying individual attention heads to preserve persistent homology, without a loss in equivariance, which can be proven rigorously. This can decrease parameter count, as well as some of the need for things like data augmentation, and can also improve training efficiency. This can be applied independently of the topological loss functions.

11. Knowledge Distillation via Fréchet Mean Persistence Diagrams

A Fréchet mean, also known as the barycenter, provides a notion of a 'mean' object in a metric space. In the context of persistence diagrams, a Fréchet mean persistence diagram is a diagram that minimizes the sum of the squared Wasserstein distances to all other diagrams in the collection. More formally, given a collection of persistence diagrams $\{D_i\}_{i=1}^m$, the Fréchet mean diagram $\bar{D}$ is defined as the solution to the optimization problem:

$$\bar{D} = \arg\min_D \sum_{i=1}^{m} d_{p,q}^{\mathcal{W}}(D, D_i)^2,$$

where $d_{p,q}^{\mathcal{W}}$ is the Wasserstein distance. The Fréchet mean diagram can be thought of as the 'average' persistence diagram of the collection, and it provides a summary of the overall topological features present in the dataset. The use of Fréchet mean diagrams offers a potentially fruitful approach for knowledge distillation. Suppose we have a model that consistently produces a similar persistent homology for a given motif across a range of protein sequences. We can compute the Fréchet mean diagram for this motif across all sequences, and use this mean diagram as a baseline for further modelling. In particular, this baseline can be used to define a topological loss term for training a new model. Instead of comparing the persistence diagrams of the new model's outputs to those of the original model, we compare them to the baseline Fréchet mean diagram. Formally, if D_{new} denotes the persistence diagram produced by the new model, the topological loss term could be defined as:

$$\mathcal{L}_{\text{topo}} = d_{p,q}^{\mathcal{W}}(\bar{D}, D_{new}),$$

where $\bar{D}$ is the Fréchet mean diagram of the original model's outputs. The total loss function $\mathcal{L}$, which includes the standard loss term $\mathcal{L}_{\text{task}}$ and the topological loss term $\mathcal{L}_{\text{topo}}$, can then be written as:

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \lambda \mathcal{L}_{\text{topo}},$$

where λ is a hyperparameter that controls the balance between the two loss terms. This approach effectively distils the topological knowledge of the original model into the new model, leveraging the consistent topology of the motif across sequences as captured by the original model. It allows the

new model to prioritize motifs that are consistently relevant across a range of sequences, potentially improving its generalization capability.

12. Clustering Proteins

12.1. **Methods.** Using the Wasserstein distance between persistence diagrams associated to protein sequences, we can form distance matrices of protein sequences which we can run clustering algorithms on such as k-Means, Agglomerative clustering, HDBSCAN, or even a persistent homology informed DBSCAN where we use H_1 features with low threshold values as a way to select the distance parameter ϵ for DBSCAN[5]. The use of the Wasserstein distance between persistence diagrams as a metric provides us with a topological characterization of protein sequences. Each persistence diagram acts as a representation of a protein sequence, encoding the sequence's inherent topological information. By defining distances between these diagrams using the Wasserstein distance, we are essentially quantifying the dissimilarities between protein sequences in terms of their topologies.

Given this framework, we can construct a distance matrix for a set of protein sequences, where the entry in the i-th row and j-th column corresponds to the Wasserstein distance between the persistence diagrams associated to the i-th and j-th protein sequences. Such a matrix provides us with a topological profile for each protein and sets the stage for applying various clustering algorithms. Algorithms such as k-Means, Agglomerative Clustering, and HDBSCAN can be utilized for clustering the protein sequences based on these topological profiles. Given the distance matrix, these algorithms can effectively group similar proteins together, potentially revealing interesting and novel insights about protein families or functional groups.

In particular, a persistent homology-informed DBSCAN is a notable mention, which goes further by integrating the topological data analysis into the choice of the algorithm's hyperparameters. Specifically, we can use the birth and death times of the H_1 features (representing loops) that have low filtration values as a way to select the distance parameter ϵ for DBSCAN. The idea behind this is to find an ϵ that captures significant topological features without creating excessive noise. In other words, we want ϵ to be large enough to capture meaningful structure but not so large that it merges distinct topological features.

Formally, if we denote by B and D the birth and death times of the H_1 features, and by T a chosen threshold value, we could choose ϵ as follows:

$$\epsilon = \min\{d - b \,|\, (b, d) \in (B, D),\ d - b > T\}.$$

This effectively makes ϵ the smallest persistence of an H_1 feature that is above the threshold T, aligning the scale of the DBSCAN algorithm with the scale of significant topological features in the protein sequences. The application of such a technique could lead to a more refined and biologically meaningful clustering of proteins, exploiting the full potential of topological data analysis in protein sequence analysis.

12.2. **Example.** Using the proteins in Table 3, we compute the persistent homology of the attention probability distributions using the Jensen-Shannon distance metric.[6] For this example we use head 2 of layer 3. Once we have computed persistent homology, we have a collection of persistence diagrams, one for each protein sequence. Computing the pairwise distances between the persistence diagrams using the Wasserstein distance metric we obtain a 15×15 distance matrix. From this distance matrix, we can run a simple k-Means clustering algorithm, if we know the number of clusters. An initial guess would be 5, as there are five families of proteins, with three proteins in

[5]We might also use an ϵ value that captures a large number of H_0 features, and search for "elbows" in the barcode diagram, or gaps in the persistence diagram for H_0.

[6]We can also use the context vectors of a single head, or hidden states of a layer using the Euclidean distance metric.

TABLE 3. Protein Sequences

Protein Family	**Sequence**
Hemoglobin	VHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFF
Hemoglobin	VHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFFES
Hemoglobin	VHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFFESF
Insulin	MALWMRFLPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGER
Insulin	MALWMRFLPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERG
Insulin	MALWMRFLPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERGF
Albumin	MKWVTVRPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGER
Albumin	MKWVTVRPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERG
Albumin	MKWVTVRPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERGF
Myoglobin	GLSDGEWQQVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDR
Myoglobin	GLSDGEWQQVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDRF
Myoglobin	GLSDGEWQQVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDRFG
Lysozyme	KVFGRCELAAAMKRHGLDNYRGYSLGNWVCAAKFESNFNTQATNR
Lysozyme	KVFGRCELAAAMKRHGLDNYRGYSLGNWVCAAKFESNFNTQATNRN
Lysozyme	KVFGRCELAAAMKRHGLDNYRGYSLGNWVCAAKFESNFNTQATNRNT

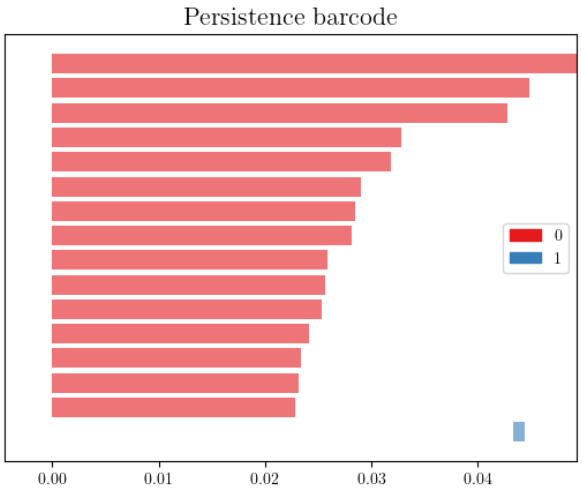


FIGURE 6. Barcode for Clustering

each family. This however, turns our to not be the optimal way of clustering the protein sequences based on the persistent homology of ESM-2's representation of them. Instead, we run persistent homology a second time on the distance matrix, obtaining a barcode diagram in Figure 6.

Notice the "elbow" shape that occurs near in the range $0.3 - 0.35$. This indicates a good value for ϵ that captures most of the H_0 features, and since the H_1 features lie above this range, we don't miss out on capturing some of the H_1 features. Next, we run a DBSCAN at $\epsilon = 0.35$. Doing so yields the clusters in Table 4.

Next, running k-Means and Agglomerative clustering algorithms with the number of clusters set to 3 yields a set of three clusters. From this, we get silhouette scores for DBSCAN, k-Means,

TABLE 4. Clusters for DBSCAN Layer 3 Head 2

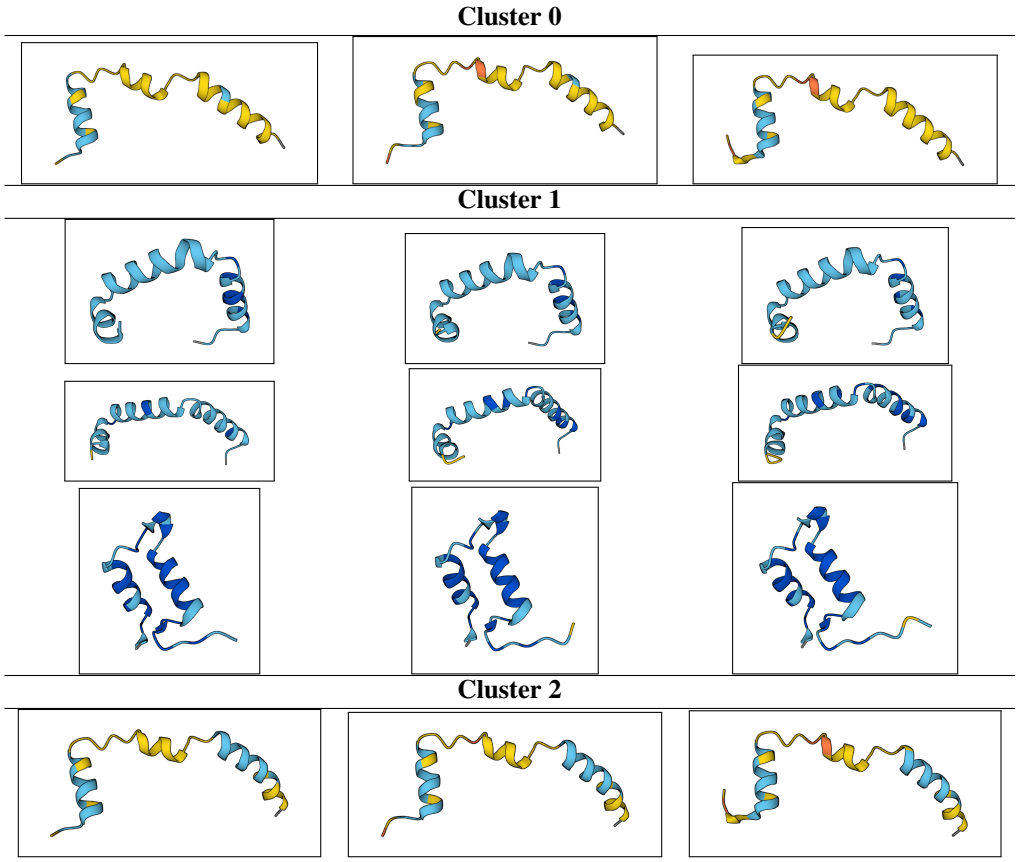
Cluster	Protein Sequences
0	MKWVTVRPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGER
	MKWVTVRPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERG
	MKWVTVRPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERGF
1	VHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFF
	VHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFFES
	VHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFFESF
	GLSDGEWQQVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDR
	GLSDGEWQQVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDRF
	GLSDGEWQQVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDRFG
	KVFGRCELAAAMKRHGLDNYRGYSLGNWVCAAKFESNFNTQATNR
	KVFGRCELAAAMKRHGLDNYRGYSLGNWVCAAKFESNFNTQATNRN
	KVFGRCELAAAMKRHGLDNYRGYSLGNWVCAAKFESNFNTQATNRNT
2	MALWMRFLPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGER
	MALWMRFLPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERG
	MALWMRFLPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERGF

and Agglomerative Clustering of 0.410. We also find that the Adjusted Random Index (ARI) score between each pair of clustering algorithms is 1, indicating they have all found the exact same clusters and that we have chosen ϵ, and well as the number of clusters for k-Means and Agglomerative Clustering well. It is noteworthy that the clusters do not fall into the five families we expect. This is likely due to the attention head finding patterns within the sequences that we are not aware of or that go deeper than the naive division into five families that we initially gave. We note that the structural similarities in the $3D$ folded structure predicted by ESMFold for Cluster 0 and are all very similar. Similarly for Cluster 2. However, for Cluster 1, there are essentially two distinct topologies, one set given by the first 6 proteins in the cluster, the other given by the last 3 (see Figure 7).

Explaining the clusters visually, from the $3D$-structure alone, we see four distinct clusters, not three. In particular, the final three proteins in Cluster 1 seem to be different topologically than the other six proteins in Cluster 1. So, this attention head of ESM-2 is potentially finding something beyond the 3-dimensional structure predicted by ESMFold and using this to group these proteins together into a clusters, in addition to the topological structure of the proteins. Running the same analysis with a different head (layer 5, head 1) we get clusters that are closer to the five families we initially defined (see Table 6). However, there is still one protein that is grouped in with proteins that are not in the same family, and the two families that have very similar geometric structure are no longer grouped together. This indicates the head is focusing less on geometric information and more on information related to the five initial families we defined. This highlights the importance of not only running the analysis on multiple attention heads with an eye toward model interpretability, but also of performing the analysis at multiple abstraction levels, from attention probability distributions, to context vectors, to hidden states output by an entire layer. This gives us a view of ESM-2's internal representations of proteins and how it thinks about protein sequences, as well as potential places where we might look for biological significance of the clusters.

With the work of [W-SOCK] in mind, we note that this clustering method is something of a generalization of the multiscale DBSCAN used to define the AF-cluster pipeline. In particular, rather than clustering the final hidden states of AlphaFold2, we are clustering attention probability distributions, context vectors, or hidden states in terms of their persistent homology. With this in

TABLE 5. Clusters for Layer 3 Head 2



mind, we may apply the above methods to detection and classification of fold-switching proteins that have multiple stable conformational states that they can exist in.

13. ANOMALOUS PROTEIN DETECTION

Identifying anomalies within a set of proteins can provide critical insights into diverse biological phenomena, including disease-associated mutations or novel protein families. Here, we propose a method for anomalous protein detection, leveraging the persistent homology representation of proteins and the Wasserstein distance metric.

Let's assume that we have a distinct cluster of proteins obtained from the techniques described in the previous section. For this cluster, we can compute the Fréchet mean persistence diagram, which serves as a topological representation of the "average" protein within the cluster. This Fréchet mean diagram captures the prevalent topological features common to the proteins in this cluster and acts as a baseline for comparison.

Given this baseline, we can analyze new unseen protein sequences by computing their persistence diagrams and comparing these diagrams to the Fréchet mean diagram of the original cluster. The comparison is done using the Wasserstein distance metric. If the Wasserstein distance between the new protein's persistence diagram and the Fréchet mean diagram is a statistical outlier compared

TABLE 6. Clusters for Head 1 in Layer 5

Cluster	Protein Sequences
Cluster 0	VHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFF
	MKWVTVRPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGER
	MKWVTVRPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERG
	MKWVTVRPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERGF
Cluster 1	VHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFFES
	VHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFFESF
Cluster 2	MALWMRFLPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGER
	MALWMRFLPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERG
	MALWMRFLPLLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERGF
Cluster 3	GLSDGEWQQVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDR
	GLSDGEWQQVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDRF
	GLSDGEWQQVLNVWGKVEADIPGHGQEVLIRLFKGHPETLEKFDRFG
Cluster 4	KVFGRCELAAAMKRHGLDNYRGYSLGNWVCAAKFESNFNTQATNR
	KVFGRCELAAAMKRHGLDNYRGYSLGNWVCAAKFESNFNTQATNRN
	KVFGRCELAAAMKRHGLDNYRGYSLGNWVCAAKFESNFNTQATNRNT

to the distribution of the Wasserstein distances of the original diagrams to the Fréchet mean diagram, we can infer that the new protein sequence likely does not share similar topological (and thus potentially biological) properties with the proteins in our cluster.

Formally, if D represents the persistence diagram of the new protein and D_F represents the Fréchet mean diagram, the anomaly score S for the new protein can be defined as follows:

$$S = \frac{|d_{p,q}^{W}(D, D_F) - \mu|}{\sigma}$$

where μ and σ are the mean and standard deviation of the Wasserstein distances of the original diagrams to the Fréchet mean diagram, respectively. If S is above a certain threshold (determined using methods like the Z-score), then the new protein can be flagged as an anomaly.

Conversely, if the Wasserstein distance is not an outlier, then it is possible that the new protein sequence shares similar biological properties with the original set of protein sequences. In such cases, it would be advantageous to begin searching for common motifs and other markers of similarity between the new protein sequence and the original set of protein sequences in the cluster. This approach can help to uncover shared functional or structural characteristics, leading to a more in-depth understanding of the biological implications of these protein sequences.

14. Application and Relations to Vision Transformers, Large Language Models, 2D-Diffusion Models, Text-to-3D, and Text-to-Video

The techniques proposed in this work demonstrate the cross-applicability of transformer models and topological data analysis methods across various domains, from protein structure prediction to vision, natural language processing, and diffusion models.

14.1. **Text-to-3D with Diffusion Models.**

In this section, we elucidate the utilization of persistent homology in a diffusion model with a transformer backbone such as those mentioned in for the generation of intricate 3D protein structures. Our discussion is twofold, focusing on topological consistency preservation in both individual generative diffusion steps and multiple protein contexts sharing a sequential or structural motif.

Given $\mathbf{x} \in \mathbb{R}^{N\times 3}$ as the 3D coordinate positions of N atoms in a protein structure, the generative process adopts an iterative denoising scheme on the coordinate variables $\mathbf{x}$. This scheme inverts the noise diffusion process at each step with the support of a transformer network θ, which conditions on the input protein sequence $\mathbf{s}$:

$$\begin{aligned} q(\mathbf{x}_0) &= \mathcal{N}(\mathbf{x}_0; \mathbf{0}, \mathbf{I}) \\ q(\mathbf{x}_t|\mathbf{x}_{t-1}) &= \mathcal{N}(\mathbf{x}_t;\ \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}) \\ p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{s}) &= \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, \mathbf{s}), \Sigma_\theta(\mathbf{x}_t, \mathbf{s})) \end{aligned}$$

Topological consistency is embedded into our model by applying persistent homology on three core elements of the transformer architecture: the attention probability distributions, the context vectors, and the hidden states at each layer.

We first compute the persistent homology on the simplicial complex formed from the softmax of the rows of the attention matrix $a_{ij}^{l,h}$ for each head h at layer l. In the case of context vectors $\mathbf{c}_i^{l,h}$, defined as $\mathbf{c}_i^{l,h} = \sum_{j=1}^{n} a_{ij}^{l,h}\mathbf{v}_j^{l,h}$, we form a simplicial complex and calculate its persistent homology. For hidden states, we compute persistent homology at each layer of the transformer, thus constructing a topological signature for the complete layer.

The topological loss term $\mathcal{L}_{topo} = d_{p,q}^{\mathcal{W}}(D_t, D_{t-1})$ encourages topological consistency across diffusion steps by computing the Wasserstein distance between persistence diagrams D_t and D_{t-1}:

$$d_{p,q}^{\mathcal{W}}(D_t, D_{t-1}) = \left(\sum_i \|u_i - v_i\|_q^p\right)^{\frac{1}{p}}$$

where u_i and v_i are points in the persistence diagrams, and p and q are often set to 2.

Now, let's consider multiple protein contexts and preserving persistent homology across contexts. Proteins often share common sequential or structural motifs across diverse families, and these motifs are crucial for the biological function of these proteins. We conjecture that by maintaining the persistent homology of these common motifs across different protein contexts, we could enhance the overall quality of generated proteins.

To implement this, we first identify the shared motifs in the protein sequence or structure. We then construct simplicial complexes for these motifs and compute their persistent homology. Once computed, this persistent homology is used as a fixed point to align the generative process. In other words, when a common motif is identified during the generative process, we use a topological loss function to constrain the persistent homology of the motif being generated to match the fixed persistent homology. We can build a training regimen or curriculum that successively builds up more and more complex sequential and structural motifs, gradually moving from small sequences of amino acids, to larger ones forming structures like α-helices and β-sheets, on to larger secondary structural motifs, and into tertiary structures.

In essence, the motif's topological signature acts as a topological 'anchor,' guiding the generative process to yield biologically plausible structures. We posit that this strategy would yield generated proteins with high topological consistency and improved biological plausibility, measured using established metrics such as backbone RMSD (bRMSD) and MolProbity score. This advanced approach, embedding persistent homology in a diffusion model with a transformer backbone, allows for the generation of protein structures with maintained topological consistency both across diffusion steps and across multiple protein contexts. The methodology provides a novel direction in the field of structural bioinformatics and facilitates a deeper understanding of complex 3D structures.

14.2. **Vision Transformers.** Vision Transformers (ViTs) have recently emerged as a powerful tool for image classification tasks. They leverage the self-attention mechanism of transformers to capture long-range dependencies between pixels in an image. In our work, we adopted a similar self-attention mechanism to capture dependencies between amino acids in protein sequences. Moreover, our proposed topological term in the loss function can potentially be incorporated into the training of ViTs, providing an inductive bias that might further enhance the performance of these models. In these models, we might use image transformations such as rotations, reflections, and translations, and impose preservation of persistent homology via a topological term in the loss function.

14.3. **Large Language Models.** Large language models like GPT-3 have been successful in generating human-like text and even solving complex tasks without any task-specific training data. Our protein analysis approach bears some similarities with these models, as we used transformers to analyze the 'language' of proteins. Furthermore, our approach to compute the persistent homology of subsequences in different contexts mirrors the context-aware understanding exhibited by large language models. We have found that preserving the persistent homology of certain keyphrases, collocations, multiword expressions, and idioms might be analogous to preserving persistent homology of motifs in protein sequences. This could potentially be used to improve LLMs and machine translation models, where persistent homology is preserved after translation.

14.4. **Text-to-Video with Diffusion Models.** In the domain of video synthesis from text descriptions, understanding and preserving the topological features might be crucial for generating videos that accurately reflect the text's content. Our approach of leveraging persistent homology could provide a means of characterizing and quantifying these topological features in the video synthesis process. Here we may wish to preserve the persistent homology of consecutive frames in the video, where topological stability may help with temporal coherence.

Our proposed approach of combining transformer models with topological data analysis provides a novel methodology that could extend to multiple domains beyond protein structure analysis. It is our hope that this cross-pollination of techniques will inspire further research in these exciting areas.

15. Concluding Remarks

In this study, we have explored the application of topological data analysis, specifically persistent homology, in the context of protein sequence and structure analysis. We demonstrated how the high-dimensional data inherent in protein sequences can be transformed into a simplicial complex, the topological features of which can be summarized in the form of a persistence diagram. This transformation not only preserves important characteristics of the protein's sequence and structure but also allows for more nuanced analysis methods to be applied.

We observed that both sequence and structural motifs within protein sequences exhibit strong associations in the simplex tree. The topological features of these motifs, as represented by persistence diagrams, were found to be strongly preserved across different protein sequences, lending further support to their biological significance. We developed a measure based on the Wasserstein distance to quantify the degree to which the persistent homology of a motif is preserved across various contexts. This measure could provide a valuable tool for investigating the structural or functional roles of these motifs and evaluating the capability of models in capturing recurrent topological patterns.

Additionally, we proposed a topological term in the loss function using the Wasserstein distance,

$$d_{p,q}^{\mathcal{W}}(D_1, D_2) = \left(\inf_{\gamma \in \Gamma(D_1, D_2)} \sum_{(u,v)\in\gamma} \|u - v\|_q^p \right)^{1/p}$$

to encode the invariance of persistent homology of motifs as an inductive bias in the model, leading to improved performance. We also presented a method to leverage the Fréchet mean persistence diagram as a topological representation of the ”average” protein for anomaly detection in unseen protein sequences.

Finally, we discussed how these methods can be utilized to cluster proteins based on their persistent homology, and to detect novel protein sequences. The results indicated that this approach can be valuable for identifying proteins with shared biological properties and uncovering common motifs and markers of similarity. This work represents an novel application of persistent homology in the analysis of protein sequence and structure data. The methods we developed leverage the power of topological data analysis to provide new insights into the fundamental properties of proteins. We anticipate that these methods can be readily extended to other biological data types and offer exciting opportunities for future research in bioinformatics.

References

[B] Matthew Berger, *Visually Analyzing Contextualized Embeddings*, https://arxiv.org/abs/2009.02554

[BERTopic] Grootendorst, Maarten, *BERTopic: Neural topic modeling with a class-based TF-IDF procedure*, https://maartengr.github.io/BERTopic/index.html

[CH] Anna Currey, Kenneth Heafield, *Incorporating Source Syntax into Transformer-Based Neural Machine Translation*, Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers), August 2019, https://aclanthology.org/W19-5203/

[CKS] Ryan Cotterell, Arun Kumar, Hinrich Schütze, *Morphological Segmentation Inside-Out*, https://arxiv.org/abs/1911.04916v2

[CTMPKABPPB] Daniil Cherniavskii, Eduard Tulchinskii, Vladislav Mikhailov, Irina Proskurina, Laida Kushnareva, Ekaterina Artemova, Serguei Barannikov, Irina Piontkovskaya, Dmitri Piontkovski, Evgeny Burnaev, *Acceptability Judgements via Examining the Topology of Attention Maps*, Findings of the Association for Computational Linguistics: EMNLP 2022, https://aclanthology.org/2022.findings-emnlp.7/

[DL] Vincent Divol, Théo Lacombe*Understanding the Topology and the Geometry of the Space of Persistence Diagrams via Optimal Partial Transport*, https://arxiv.org/abs/1901.03048

[DLYF] Yu Dai, Yuqiao Liu, Lei Yang, Yufan Fu, *An Idiom Reading Comprehension Model Based on Multi-Granularity Reasoning and Paraphrase Expansion*, Appl. Sci. 2023, 13, 5777. https://doi.org/10.3390/app13095777

[DW] Tamal Krishna Dey, Yusu Wang, *Computational Topology for Data Analysis*, Cambridge University Press, https://www.cs.purdue.edu/homes/tamaldey/book/CTDAbook/CTDAbook.pdf

[Gre] Daria Grechishnikova, *Transformer neural network for protein-specific de novo drug generation as a machine translation problem*, Scientific Reports volume 11, Article number: 321 (2021), https://www.nature.com/articles/s41598-020-79682-4

[G] Maarten Grootendorst, *BERTopic: Neural topic modeling with a class-based TF-IDF procedure*, https://arxiv.org/abs/2203.05794

[GZCY] Shanghua Gao, Pan Zhou, Ming-Ming Cheng, Shuicheng Yan, *Masked Diffusion Transformer is a Strong Image Synthesizer*, https://arxiv.org/abs/2303.14389

[H] Ryu Hayakawa, *Quantum algorithm for persistent Betti numbers and topological data analysis*, 2022-12-07, volume 6, page 873, https://quantum-journal.org/papers/q-2022-12-07-873/

[HFSC] Xiaoling Hu, Li Fuxin, Dimitris Samaras, Chao Chen, *Topology-Preserving Deep Image Segmentation*, https://arxiv.org/abs/1906.05404

[HFSSV] Pantea Haghighatkhah, Antske Fokkens, Pia Sommerauer, Bettina Speckmann, Kevin Verbeek, *Story Trees: Representing Documents using Topological Persistence*, Proceedings of the Thirteenth Language Resources and Evaluation Conference, June 2022, https://aclanthology.org/2022.lrec-1.258/

[Hu et. al.] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, *LoRA: Low-Rank Adaptation of Large Language Models*, https://arxiv.org/abs/2106.09685

[KCMABBPPB] Laida Kushnareva, Daniil Cherniavskii, Vladislav Mikhailov, Ekaterina Artemova, Serguei Barannikov, Alexander Bernstein, Irina Piontkovskaya, Dmitri Piontkovski, Evgeny Burnaev, *Artificial Text Detection via Examining the Topology of Attention Maps*, Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, https://aclanthology.org/2021.emnlp-main.50/

[LARHZLSVKSCZSCR] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore

Candido, Alexander Rives, *Evolutionary-scale prediction of atomic level protein structure with a language model*, https://www.biorxiv.org/content/10.1101/2022.07.20.500902v3

[LCO] Théo Lacombe, Marco Cuturi, Steve Oudot, *Large Scale computation of Means and Clusters for Persistence Diagrams using Optimal Transport*, https://arxiv.org/abs/1805.08331

[LGTTZHKFLL] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, Tsung-Yi Lin, *Magic3D: High-Resolution Text-to-3D Content Creation*, https://arxiv.org/abs/2211.10440

[LLCCLFZLW] Minghao Li, Tengchao Lv, Jingye Chen, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, Furu Wei, *TrOCR: Transformer-based Optical Character Recognition with Pre-trained Models*, https://arxiv.org/abs/2109.10282

[M] German Magai, *Deep neural networks architectures from the perspective of manifold learning*, https://arxiv.org/abs/2306.03406

[MMH] *Probability measures on the space of persistence diagrams*, Yuriy Mileyko, Sayan Mukherjee, John Harer, https://math.hawaii.edu/ yury/papers/probpers.pdf

[MHRB] Michael Moor, Max Horn, Bastian Rieck, Karsten Borgwardt, *Topological Autoencoders*, Proceedings of the 37th International Conference on Machine Learning, PMLR 119:7045-7054, 2020, http://proceedings.mlr.press/v119/moor20a.html

[PAP] Irina Proskurina, Ekaterina Artemova, Irina Piontkovskaya, *Can BERT eat RuCoLA? Topological Data Analysis to Explain*, Proceedings of the 9th Workshop on Slavic Natural Language Processing 2023 (SlavicNLP May 2023), https://aclanthology.org/2023.bsnlp-1.15/

[PJBM] Ben Poole, Ajay Jain, Jonathan T. Barron, Ben Mildenhall, *DreamFusion: Text-to-3D using 2D Diffusion*, https://arxiv.org/abs/2209.14988

[PX] William Peebles, Saining Xie, *Scalable Diffusion Models with Transformers*, https://arxiv.org/abs/2212.09748

[RC] David W. Romero, Jean-Baptiste Cordonnier, *Group Equivariant Stand-Alone Self-Attention For Vision*, https://openreview.net/forum?id=JkfYjnOEo6M

[RCB] Raphael Reinauer, Matteo Caorsi, Nicolas Berkouk, *Persformer: A Transformer Architecture for Topological Machine Learning*, https://arxiv.org/abs/2112.15210

[RHBWFHACPHMLF] Chaitanya Ryali, Yuan-Ting Hu, Daniel Bolya, Chen Wei, Haoqi Fan, Po-Yao Huang, Vaibhav Aggarwal, Arkabandhu Chowdhury, Omid Poursaeed, Judy Hoffman, Jitendra Malik, Yanghao Li, Christoph Feichtenhofer, *Hiera: A Hierarchical Vision Transformer without the Bells-and-Whistles*, https://arxiv.org/abs/2306.00989

[RZSW] Archit Rathore, Yichu Zhou, Vivek Srikumar, Bei Wang, *TopoBERT: Exploring the Topology of Fine-Tuned Word Representations*, http://www.sci.utah.edu/ beiwang/publications/TopoBERT_BeiWang_2023.pdf

[SDM] Suzanna Sia, Ayush Dalmia, Sabrina J. Mielke, *Tired of Topic Models? Clusters of Pretrained Word Embeddings Make for Fast and Good Topics too!*, https://aclanthology.org/2020.emnlp-main.135/

[SL] Yara Skaf, Reinhard Laubenbacher*Topological data analysis in biomedicine: A review*, Journal of Biomedical Informatics Volume 130, June 2022, 104082, https://www.sciencedirect.com/science/article/pii/S1532046422000983

[SZQH] Devendra Singh Sachan, Yuhao Zhang, Peng Qi, William Hamilton, *Do Syntax Trees Help Pre-trained Transformers Extract Information?*, https://arxiv.org/abs/2008.09084

[THJ] Mozhgan Talebpour, Alba Garcia Seco de Herrera, Shoaib Jameel, *Topics in Contextualised Attention Embeddings*, https://arxiv.org/abs/2301.04339

[TM] Laure Thompson, David Mimno, *Topic Modeling with Contextualized Word Representation Clusters*, https://arxiv.org/abs/2010.12626

[TMMH] Katharine Turner, Yuriy Mileyko, Sayan Mukherjee, John Harer, *Fréchet Means for Distributions of Persistence diagrams*, https://arxiv.org/abs/1206.2790

[TKKCBPNB] Eduard Tulchinskii, Kristian Kuznetsov, Laida Kushnareva, Daniil Cherniavskii, Serguei Barannikov, Irina Piontkovskaya, Sergey Nikolenko, Evgeny Burnaev, *Topological Data Analysis for Speech Processing*, https://arxiv.org/abs/2211.17223

[TKKCBPNB-2] Eduard Tulchinskii, Kristian Kuznetsov, Laida Kushnareva, Daniil Cherniavskii, Serguei Barannikov, Irina Piontkovskaya, Sergey Nikolenko, Evgeny Burnaev, *Intrinsic Dimension Estimation for Robust Detection of AI-Generated Texts*, https://arxiv.org/abs/2306.04723

[V] Jesse Vig, *A Multiscale Visualization of Attention in the Transformer Model*, Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations July 2019, https://aclanthology.org/P19-3007/

[VHRNZG] Renato Vukovic, Michael Heck, Benjamin Ruppik, Carel van Niekerk, Marcus Zibrowius, Mil
ica Gasic, *Dialogue Term Extraction using Transfer Learning and Topological Data Analysis*, Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue, September 2022, https://aclanthology.org/2022.sigdial-1.53/

[VPLGK] Ivan Vulić, Edoardo Maria Ponti, Robert Litschko, Goran Glavaš, Anna Korhonen, *Probing Pretrained Language Models for Lexical Semantics*, Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), November 2020, https://aclanthology.org/2020.emnlp-main.586/

[WAMMR] Dominik J. E. Waibel, Scott Atwell, Matthias Meier, Carsten Marr, Bastian Rieck, *Capturing Shape Information with Multi-Scale Topological Loss Terms for 3D Reconstruction*, https://arxiv.org/abs/2203.01703

[W-SOCK] Hannah K. Wayment-Steele, Sergey Ovchinnikov, Lucy Colwell, Dorothee Kern, *Prediction of multiple conformational states by combining sequence clustering with AlphaFold2*, https://www.biorxiv.org/content/10.1101/2022.10.17.512570v1

[ZNVA] Hongkai Zheng, Weili Nie, Arash Vahdat, Anima Anandkumar, *Fast Training of Diffusion Models with Masked Transformers*, https://arxiv.org/abs/2306.09305

[ZKNHRP] Ali Zia, Abdelwahed Khamis, James Nichols, Zeeshan Hayder, Vivien Rolland, Lars Petersson, *Topological Deep Learning: A Review of an Emerging Paradigm*, https://arxiv.org/abs/2302.03836

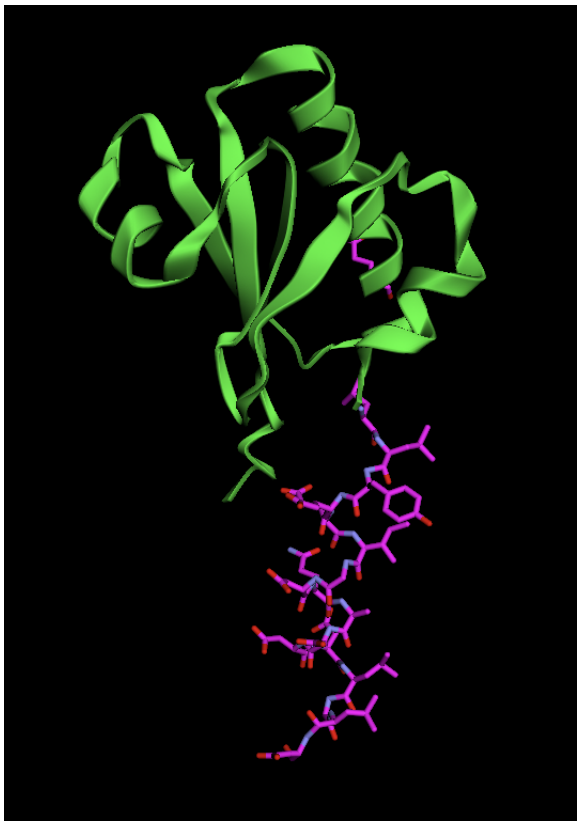


FIGURE 7. Identification of Substructure of Fold Switching KaiB

16. APPENDIX

16.1. **Locating Substructures of Fold-Switching Proteins.** In the following notebook we locate substructures of a KaiB fold-switching protein.

MAPLRKTYVLKLYVAGNTPNSVRALKTLNNILEKEFKGVYALKVIDVLKNPQL
AEEDKILATPTLAKVLPPPVRRIIGDLSNREKVLIGLDLLYEEIGDQAEDDLGLE

pictured in Figure 7.

This is done by first obtaining the hidden states of a layer, or the context vectors of a head, or the attention probability distributions of a head of ESM-2. We then treat this data as a point cloud using Euclidean distance, or Jensen-Shannon distance for the attention probability distributions. Once we have the distance matrix of pairwise distances between one of these three types of data, we compute the Vietoris-Rips complex and persistent homology, yielding a persistence diagram or barcode diagram. We then use this to choose an ϵ distance threshold for DBSCAN. Running DBSCAN yields clusters of amino acids in the protein sequence, which we then highlight in pink in the $3D$-folded structure predicted by ESMFold. We note that some heads will cluster things like binding sites, secondary structures, intrinsically disordered regions (IDRs), sites where mutations cause a conformational change, fold-switching substructures, specific amino acids, etc. We also note the nuance and complexity of these clusters increases as the depth and parameter count of the model increases. For example, '*facebook/esm2_t12_35M_UR50D*' will have more nuanced clusters

in later layers than the shallower '*facebook/esm2_t6_8M_UR50D*'. For a list of the available models see this link.

16.2. **Jensen-Shannon Distance Metric.** The Jensen-Shannon distance (JSD) is a metric derived from the Jensen-Shannon divergence (JSDiv), a symmetrized and smoothed version of the Kullback-Leibler (KL) divergence. KL-divergence is a popular measure of dissimilarity between probability distributions and has found widespread application in deep learning.

To define the Jensen-Shannon divergence, we first recall the KL-divergence between two probability distributions P and Q:

$$\mathrm{KL}(P\|Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)}, \tag{1}$$

where i ranges over the elements in the support of the distributions. Note that KL-divergence is not symmetric, i.e., $\mathrm{KL}(P\|Q) \neq \mathrm{KL}(Q\|P)$. In order to derive a symmetrized measure, we introduce the Jensen-Shannon divergence:

$$\mathrm{JSDiv}(P, Q) = \frac{1}{2}\mathrm{KL}(P\|M) + \frac{1}{2}\mathrm{KL}(Q\|M), \tag{2}$$

where $M = \frac{1}{2}(P + Q)$ is the average of the two probability distributions. The Jensen-Shannon divergence satisfies the properties of being symmetric, i.e., $\mathrm{JSDiv}(P, Q) = \mathrm{JSDiv}(Q, P)$, and non-negative, i.e., $\mathrm{JSDiv}(P, Q) \geq 0$.

However, the Jensen-Shannon divergence is not a metric, as it does not satisfy the triangle inequality. To obtain a metric, we define the Jensen-Shannon distance as the square root of the Jensen-Shannon divergence:

$$\mathrm{JSD}(P, Q) = \sqrt{\mathrm{JSDiv}(P, Q)}. \tag{3}$$

The Jensen-Shannon distance satisfies the properties of a metric, including non-negativity, symmetry, the identity of indiscernibles, and the triangle inequality. This makes it a useful measure of dissimilarity between probability distributions for various applications, including deep learning.

16.3. **Probability Distributions from Tokens.** Now, taking X_i the i^{th} row of the matrix of token embedding vectors X, for any given attention head, we denote by

$$\langle q_i, k_j \rangle = (W^Q X_i)(W^K X_j)^T. \tag{4}$$

Then we have,

$$P(X_i) = \left(\mathrm{softmax}_j\left(\frac{\langle q_i, k_j\rangle}{\sqrt{d}}\right)\right)_{j=1}^{n} = \mathrm{softmax}\begin{pmatrix} \frac{\langle q_i, k_1\rangle}{\sqrt{d}} \\ \frac{\langle q_i, k_2\rangle}{\sqrt{d}} \\ \vdots \\ \frac{\langle q_i, k_n\rangle}{\sqrt{d}} \end{pmatrix} = \left(\frac{e^{\frac{\langle q_i, k_j\rangle}{\sqrt{d}}}}{\sum_{l=1}^{n} e^{\frac{\langle q_i, k_l\rangle}{\sqrt{d}}}}\right)_{j=1}^{n}. \tag{5}$$

To measure the dissimilarity between the attending behaviors of tokens X_i and X_j, we can compute the Kullback-Leibler (KL) divergence, denoted as $D_{KL}(P(X_i)\|P(X_j))$. The formula for KL divergence is:

$$KL(P(X_i)||P(X_j)) = \sum_{k=1}^{n} P(X_i)_k \log_2 \frac{P(X_i)_k}{P(X_j)_k} \tag{6}$$

$$= \sum_{k=1}^{n} \frac{e^{\frac{\langle q_i, k_k\rangle}{\sqrt{d}}}}{\sum_{l=1}^{n} e^{\frac{\langle q_i, k_l\rangle}{\sqrt{d}}}} \log_2 \left(\frac{\frac{e^{\frac{\langle q_i, k_k\rangle}{\sqrt{d}}}}{\sum l=1^n e^{\frac{\langle q_i, k_l\rangle}{\sqrt{d}}}}}{\frac{e^{\frac{\langle q_j, k_k\rangle}{\sqrt{d}}}}{\sum_{l=1}^{n} e^{\frac{\langle q_j, k_l\rangle}{\sqrt{d}}}}} \right) \tag{7}$$

$$= \sum_{k=1}^{n} \frac{e^{\frac{\langle q_i, k_k\rangle}{\sqrt{d}}}}{\sum_{l=1}^{n} e^{\frac{\langle q_i, k_l\rangle}{\sqrt{d}}}} \left(\log_2 \left(\frac{e^{\frac{\langle q_i, k_k\rangle}{\sqrt{d}}}}{\sum_{l=1}^{n} e^{\frac{\langle q_i, k_l\rangle}{\sqrt{d}}}} \right) - \log_2 \left(\frac{e^{\frac{\langle q_j, k_k\rangle}{\sqrt{d}}}}{\sum_{l=1}^{n} e^{\frac{\langle q_j, k_l\rangle}{\sqrt{d}}}} \right) \right) \tag{8}$$

Next, we can compute the Jensen-Shannon distance between token probability distributions using

$$\text{JSDiv}(P, Q) = \frac{1}{2}\text{KL}(P||M) + \frac{1}{2}\text{KL}(Q||M), \tag{9}$$

and

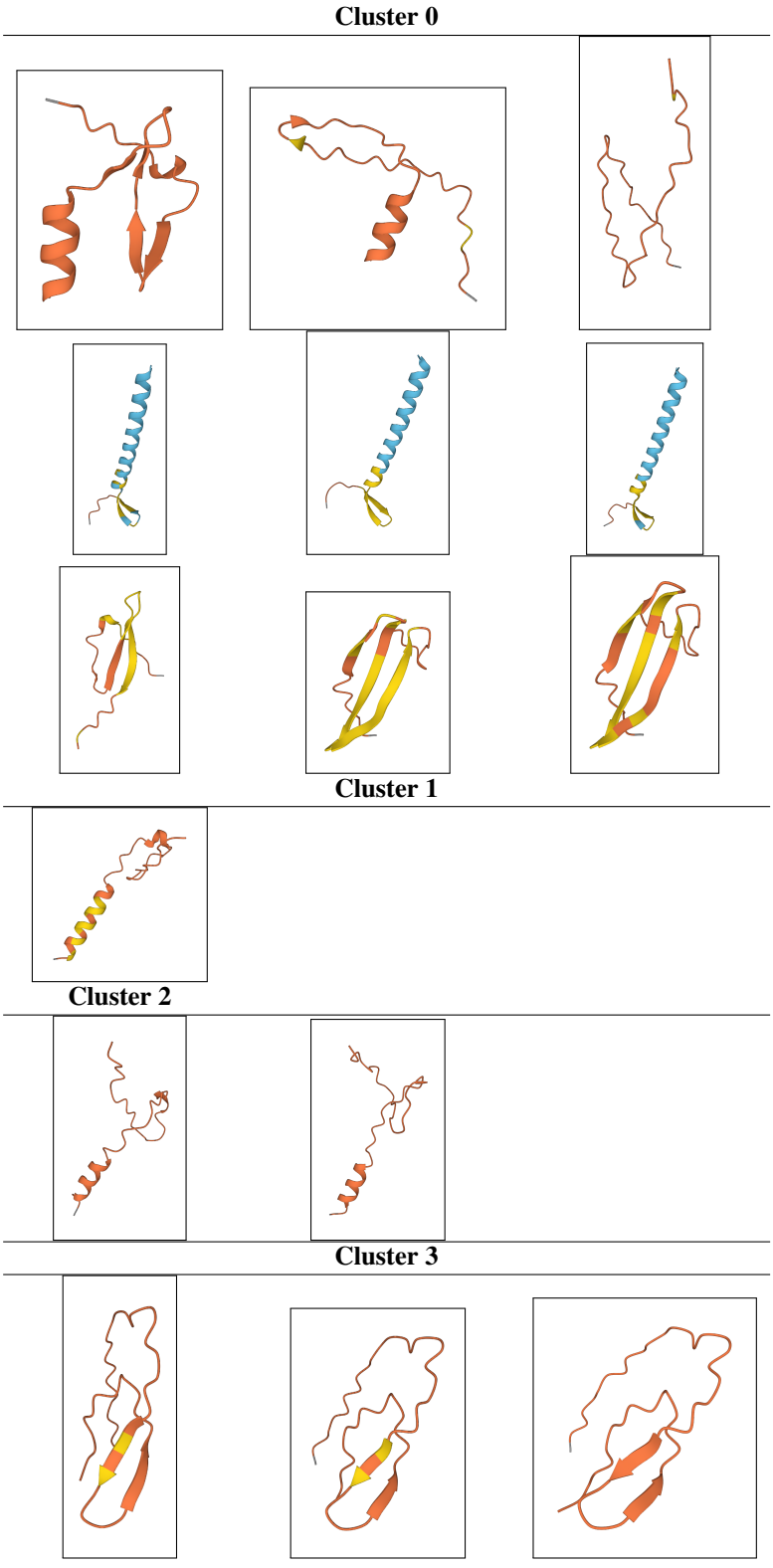
$$\text{JSD}(P, Q) = \sqrt{\text{JSDiv}(P, Q)}. \tag{10}$$

This provides us with a genuine distance metric between attention probability distributions, which we can use to compute pairwise distances between some subset of token probability distributions. In so doing, we form a distance matrix which we can then use for computing persistent homology. The analysis then is very similar to our previous analysis of pairwise distances between context vectors. In particular, with a distance metric on the probability distribution we can see how the persistent homology of some subset of tokens changes as the context changes. For example suppose we are given a structural motif or some other substructure of a protein sequence, given by some subset of positions of amino acids in the protein sequence. We can compute the persistent homology of the distance matrix formed by the pairwise distances between this subset of amino acids. Computing a second distance matrix for the same subset of amino acids found in a second protein sequence we get a second Vietoris-Rips complex and a second persistence diagram (or barcode). Comparing these two persistence diagrams using the Wasserstein distance metric indicates how well the model preserves the persistent homology of this structural motif. We might do the same for other substructures like binding sites, transitional regions, α-helices, β-sheets, etc. Understanding what structures have well preserved or invariant persistent homology, and which models preserve this persistent homology well is important for model interpretability and can help in training newer models and in knowledge distillation procedures.

16.4. **Clustering Proteins Using ESM-2.** In Figure 7 we see the cluster formed for the following 15 proteins using the hidden states of the last layer of the model '*facebook/esm2_t6_8M_UR50D*'. Note, we can also perform this analysis with other layers, with the attention probability distributions discussed in §16.3, or with the context vectors discussed in §3. The protein sequences for each cluster are given in Table 8. The pipeline for running this clustering algorithm can be done with this notebook.

Email address: amelie.schreiber.math@gmail.com

TABLE 7. Clusters for Model Output (final layer's hidden states)



Cluster	Protein Sequences
0	MAHMTQIPLSSVKRPLQVRGIVFLGICTQKGTVYGNASWVRDQARH, MKHVTQIPKSSVRRPLQFRGICFLGTCQKGTVYGKASWVHDQARHA, MNHITQVPLSSAKRPLQVRGICFLGITCKNGTVYGKACWVRDQARH, MGGHNGWQILVKGKWTTMDFLRNAVIDQKLRRARRELKLMKAFESLK, MGGHNGWQILVKGKWTTMDFLRNAVIDQKLRRARRELKLMKAFESLKN, MGGHNGWQILVKGKWTTMDFLRNAVIDQKLRRARRELKLMKAFESLKNN, MAQSNISDAMVQLTPAGRSLMLLVQHGSQVAAGVTFQDNQRFPGGRD, MAQSNISDAMVQLTPAGRSLMLLVQHGSQVAAGVTFQDNQRFPGGRDF, MAQSNISDAMVQLTPAGRSLMLLVQHGSQVAAGVTFQDNQRFPGGRDFF
1	MKLITILGLLALATLVQSTGCVTVNAAHCGVTTGQTVCAGVAKCRAE
2	MKLITILGALALATLVQSTGCVNVNAAHCVTTGQTVCAGVAKCRAET, MKLITILGALALATLVQSTGCVNVNAAHCVTAGQTVCAGVAKCRAETS
3	MGSSHHHHHHSSGLVPRGSHMENITVVKFNGTQTFEVHPNVSVGQAGV, MGSSHHHHHHSSGLVPRGSHMENITVVKFNGTQTFEVHPNVSVGQAGVR, MGSSHHHHHHSSGLVPRGSHMENITVVKFNGTQTFEVHPNVSVGQAGVRR

TABLE 8. DBSCAN Clustering of Protein Sequences