doi: DOI HERE
Advance Access Publication Date: Day Month Year
Paper

PAPER

Decoding protein language models: insights from embedding space analysis

Pia Francesca Rissom,[1, 2] * Paulo Yanez Sarmiento,[1, 2] Jordan Safer,[2]
Connor W. Coley,[3, 4] Bernhard Y. Renard,[1, 5] Henrike O. Heyne[1, 5, 6, 7] †
and Sumaiya Iqbal[2, 8] †

[1]Hasso Plattner Institute, Digital Engineering Faculty, University of Potsdam, Potsdam, 14482, Germany, [2]Bioinformatics and Machine Learning, Broad Institute of MIT and Harvard, Cambridge, 02142, MA, USA, [3]Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139, USA, [4]Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139, USA, [5]Hasso Plattner Institute, Mount Sinai School of Medicine, New York, NY 10029, USA, [6]Finnish Institute for Molecular Medicine (FIMM), University of Helsinki, Helsinki, Finland, [7]Program in Medical and Population Genetics, Broad Institute of MIT and Harvard, Cambridge, MA 02142, USA and [8]Cancer Data Sciences, Dana-Farber/Harvard Cancer Center, Boston, MA 02215, USA

†Co-supervision*Corresponding author. francesca.rissom@hpi.de

FOR PUBLISHER ONLY Received on Date Month Year; revised on Date Month Year; accepted on Date Month Year

Abstract

Foundation models, which encode patterns in large, high-dimensional data as embeddings, show promise in many machine learning related applications in molecular biology. Embeddings learned by the models provide informative features for downstream prediction tasks, however, the information captured by the model is often not interpretable. One approach to understanding the captured information is through the analysis of their learned embeddings, which in molecular biology so far has mainly focused on visualizing individual embedding spaces. This study introduces a quantitative framework for cross-space comparison, enabling intuitive exploration and comparison of embedding spaces in molecular biology. The framework emphasizes analyzing the distribution of known biological information within embedding space neighborhoods and provides insights into relationships between multiple embedding spaces. Comparison techniques include global pairwise distance measurements as well as local nearest neighbor analyses. By applying our framework to embeddings from protein language models, we demonstrate how embedding space analysis can serve as a valuable pre-filtering step for task-specific supervised machine learning applications and for the recognition of differential patterns in data encoded within and across different embedding spaces. To support a wide usability, we provide a Python library that implements all analysis methods, available at https://github.com/broadinstitute/EmmaEmb.

Key words: foundation models, protein language models, molecular biology, embedding spaces, vector space analysis, embedding interpretability, data exploration

Introduction

With the growing availability of data and computing power, foundation models have become increasingly relevant in molecular biology for encoding the high-dimensional data generated by laboratory experiments. Foundation models are trained to identify the inherent relationships within the training data, uncovering complex patterns. By using large datasets as input, foundation models place individual data points into a broader context, generating representations that are informative for a range of applications (22; 10).

Protein language models (PLMs) are a specific type of foundation model. These models are based on the architectures of Large Language Models, which have revolutionized text processing in Natural Language Processing. PLMs, such as ESM2 (22) and ProtT5 (10), are trained on large amounts of protein sequence data from various species and generate representations that capture the evolutionary relationships of proteins. The learned representations, also referred to as embeddings, can be leveraged for a variety of downstream tasks, including protein structure prediction (22), function prediction (20), and variant effect prediction (7).

As more foundation models, such as PLMs, based on varying architectures, training data, and data modalities emerge, it becomes increasingly important and interesting to understand what they have learned. This understanding can guide the appropriate application of the models but also help to detect biases in the models' representations (8). Further, understanding the models' learned patterns enables researchers to leverage them to offer new biological insights (4). However, due to their complex architecture and high

Table 1. Comparison of published tools for the simultaneous analysis of multiple embedding spaces.

		Feature distribution			Space comparison					
Tool Name	Visualisation of embedding space *	Vizualisation	Quantification	Neighbors	Vizualisation	Global analysis	Neighborhood analysis	Feature integration	Code available	Tool Description
EmmaEmb	PCA, t-SNE, UMAP	✔	✔	✔	✔	✔	✔	✔	✔	The proposed tool in this paper
repcomp (28)	-	-	-	-	-	✔	✔	-	✔	Computation of similarities between two embedding spaces; Nearest neighbours, canonical correlation, unit match; No visualisation of results
embComp (15)	IsoMap, LLE, MDS, PCA, t-SNE, UMAP	-	-	-	✔	-	✔	-	-	Visual local neighbourhood analysis, based on KNN; Distribution of similarities; Cosine or Euclidean distance
Embedding Comparator (5)	PCA, t-SNE, UMAP	-	-	-	✔	-	✔	-	✔	Visual local neighbourhood analysis, based on KNN; Distribution of similarities; Highlighting least similar samples and their nearest neighbours; Cosine or Euclidean distance
Vectory (24)	PCA, UMAP	-	-	-	✔	✔	-	-	✔	Aggregated similarity of local neighbourhoods based on KNN; Cosine or Euclidean distance
Emblaze (29)	PCA, t-SNE, UMAP	✔	-	-	✔	-	✔	-	✔	Visual local neighbourhood analysis, based on KNN; Visualising trajectories of points between embedding spaces; Highlighting nearest neighbours of individual embeddings and user-selected sets of embeddings; Cosine or Euclidean distance
EmbCompare (11)	PCA	✔	-	-	✔	✔	✔	-	✔	Visual local neighbourhood analysis, based on KNN; Distribution of similarities; Highlighting least similar samples; Cosine distance

* IsoMap (32); LLE: local linear embedding (25); MDS: multi-dimensional scaling (19); PCA: principal component analysis; t-SNE: t-stochastic neighborhood embedding; UMAP: uniform manifold approximation and projection

number of parameters, understanding what these models have learned is not straightforward. Therefore, it is crucial to develop methods for understanding the underlying patterns and features captured by the models.

In Natural Language Processing, embedding space analysis is a key approach for interpreting model behavior. Getting a better understanding of the distribution of points within one embedding space has helped to reveal semantic relationships (33) and biases (6) in the embeddings of language models. Additionally, identifying similarities between embedding spaces from different languages has facilitated the transfer of information from high-resource languages to a low-resource language, which has limited training data (13).

In the field of molecular biology, embedding space analysis has also been leveraged, for example, to understand the properties of the embedding space and its impact on continuous prediction tasks (1; 12), as well as to analyze multiple modalities within a single embedding space (3). However, comparisons across embedding spaces have primarily been performed visually, often using dimensionality reduction techniques such as principal component analysis, t-distributed stochastic neighbor embedding, uniform manifold approximation and projection, pairwise controlled manifold approximation, and multidimensional scaling (27; 22) or by the use of tree-based hierarchical visualizations (35). We see a potential for advancing the use of embedding space analysis in the field of molecular biology through quantitative analysis across embedding space.

A common standard for understanding which features, such as the function of a protein, are captured by a model is to benchmark the model on downstream tasks using supervised machine learning (ML) (16). Small task-specific classifiers are trained on PLM embeddings. This approach requires task-specific models for each feature and each PLM of interest, which becomes increasingly resource-intensive as the number of models and benchmark datasets grows. Analysis of the embedding space provides a less resource-intensive initial evaluation. By examining the distribution of categorical features across various embedding spaces, we can gain insights into how these features are represented by assessing the proximity of samples from different classes. Although this analysis is less detailed and does not uncover complex relationships between the embedding dimensions, it offers valuable initial insights into the informativeness of the representations across different embedding spaces and contexts. This initial evaluation is particularly useful for determining which models warrant the application of more resource-intensive ML approaches.

Additionally, by directly comparing the distances between samples in different embedding spaces, we gain insights into similarities and differences between the models' learned information. Discrepancies in distances between the spaces can provide insights into complementary information that the models have acquired or highlight potential biases or noise the models have learned. One bottleneck hindering the cross-space embedding analysis of foundation models in molecular biology is the lack of frameworks and tools to facilitate

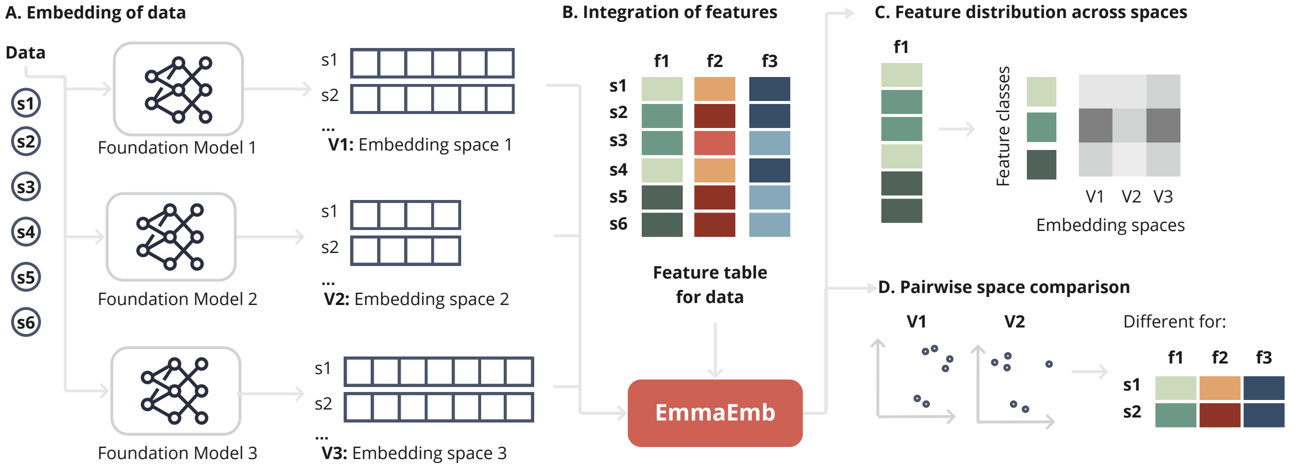


Fig. 1. Overview of the proposed analysis workflow. Our framework aims to analyse the information captured in different embedding spaces. **A**. Starting with a set of samples (e.g. genes or proteins, *here*: Data $\mathbf{s_1} - \mathbf{s_6}$), embeddings are derived from different foundation models (*here*: Foundation Models 1-3). In the example of proteins, embeddings could be derived from protein language models that vary in their architecture or training. **B** Categorical data about the samples (*here*: $\mathbf{f_1} - \mathbf{f_3}$) is integrated into the analysis, e.g. annotations of the function of proteins, protein families, etc. The embeddings, along with sample-specific categorical feature data, are then analyzed together using two main types of approaches. First, **C.** deriving the distribution of specific features, such as protein families, within neighborhoods of each embedding space, and comparing them across embedding spaces. This analysis is based on a k-nearest neighbors approach, quantifying how closely features from one class are in proximity and which classes are neighboring classes. Secondly, **D.** two embedding spaces can be compared to identify similarities and differences in the pairwise distances between embeddings or k-nearest neighbors. Regions of high difference can be further analyzed using the feature data to characterize the samples that were represented differently by the model. The introduced EmmaEmb python framework is a freely accessible implementation of the described approach.

rapid exploration. Although approaches for comparing multiple embedding spaces exist in other domains, particularly language (see Table 1), their value in molecular biology applications remains limited, and to our knowledge, the listed tools have not been applied in this context. Two key factors contribute to this gap. First, there has been a great emphasis on comparing embedding spaces visually after dimension reduction methods are applied, with little exploration of quantifying feature distributions. Second, there is often a focus on analyzing individual data points, a practice common in tools designed for Natural Language Processing, where entities such as words are easily interpretable by users. However, in molecular biology, context is essential, as the names of proteins and genes by themselves may not provide substantial meaning, even to experts. Instead, valuable insights are often gained by understanding the distribution of groups of data points with a specific biological meaning. Therefore, we emphasize the importance of incorporating the relationship between different natural groupings of the samples into the analysis, which is commonly lacking in Natural Language Processing tools.

To bridge this gap, we developed a comprehensive framework for the quantitative and cross-space analysis of embedding spaces in molecular biology. Our approach focuses on feature distributions within embedding space neighborhoods and enables comparisons across embedding spaces using methods such as global distance metrics and local neighborhood analyses. We evaluate insights gained from our analysis framework by comparing them to those derived from task-specific ML models. Additionally, we investigate how embedding space analysis can be leveraged to generate new hypotheses about the similarities between data representations across different models. To promote accessibility, we publish the Python library Embedding, Metadata, and Multimodel Analyzer for Exploration in Molecular Biology (EmmaEmb), which implements all framework components.

Methods and Materials

1. Embedding space analysis framework

Figure 1 provides an overview of our analysis framework. We consider a set of samples $s_1, \dots, s_{N_s}$, e.g. proteins, genes, etc., and $N_V \geq 2$ embedding spaces $V_h \subset \mathbb{R}^{d_h}$, $h = 1, \dots, N_V$. Every embedding space V_h corresponds to a different foundation model which embeds the input samples into numerical vectors, i.e., $s_i \mapsto v_i^{(h)} \in V_h$, $\forall\, h, i$. To analyze the information captured in the embedding spaces, a table of categorical feature vectors $\mathbf{f}_1, \dots, \mathbf{f}_{N_f}$, where each is a vector of length N_s containing the feature values of each sample. By $\mathbf{f}_p[i]$ we denote the value of the p-th feature of s_i. The feature data can for example include biological natural groupings, such as protein family information, or experimental data, such as functional annotations.

The embedding spaces may vary in dimensions, and there is no direct correspondence between dimensions across embedding spaces. To facilitate comparison across models, we focus our analysis on the distances between embeddings within each space. For each pair of samples s_i and s_j, the distance between their respective embeddings, $\mathrm{dist}(v_i^{(h)}, v_j^{(h)})$, is measured in each embedding space V_h. The pairwise distance measures form the foundation for the following analysis. Different distance measures, such as Euclidean, cosine, and Manhattan distance, can and should be used to inspect the proximity of data points from varying viewpoints (21). For parts of the analysis, only the k-nearest neighbors (KNN) are considered to focus on local parts of the embedding space. KNN based metrics have been tailored for various applications (34), here we adjust and leverage them for the needs of embedding comparisons. The neighbors are identified based on pairwise distance measures, with users defining the distance metric and the number of neighbors k. We suggest setting k to around 10% of the total number of samples, but it will not be larger than the size of the smallest feature class.

1.1 Analysis methods

We gathered a set of quantitative methods that are suitable to measure information across embedding spaces. We group the methods into feature distribution and pairwise space comparison methods (see Figure 2). The choice of analysis steps depends on the research goals, and we do not suggest that all methods must be applied or follow a particular sequence. Each method provides a unique perspective on what the foundation models have learned, enabling insights tailored to specific use cases.

1.1.1 Feature distribution across spaces

KNN feature alignment scores

We define scores that provide insight into how closely related a sample is to its neighbors in terms of the feature $\mathbf{f}_p$.

For a given feature $\mathbf{f}_p$ and embedding space V_h, the KNN feature alignment score for each sample is defined as the fraction of KNN $\mathcal{N}_{s_i}^{(h)}$ that share the same class label as the sample.

$$\text{Score}_p^{(h)}(s_i) := \frac{1}{k}\left|\left\{s_j \in \mathcal{N}_{s_i}^{(h)} : \mathbf{f}_p[j] = \mathbf{f}_p[i]\right\}\right|$$

The score ranges from 0 to 1, with higher values indicating a greater representation of the feature class in the sample's neighborhood.

KNN feature alignment scores can be stratified by feature labels to examine the extent to which each feature class is captured in an embedding space. Furthermore, KNN feature alignment scores of one embedding space can be directly compared to those from another embedding space, enabling an assessment of how well different models capture relationships for the same feature.

KNN class mixing matrix

We propose the inspection of how close individual feature classes are in each of the embedding spaces to gain insights into feature classes how similar different feature classes are encoded by the models. For each sample in a given embedding space and for a selected feature, we count how many times each feature class occurs within the KNN embeddings. For each feature class, we then sum the number of times the neighbors belong to that class for all samples that share the same class label. This process is repeated for every feature class, aggregating the class counts across all samples in the embedding space to get a comprehensive view of how feature classes are distributed within the space. The results are presented in a matrix format, where each column corresponds to a specific feature class, and each row represents the count of neighbors for samples labeled with that feature class, organized by their own feature class. High values in the matrix suggest that embeddings from the feature class in the column are frequently found close to samples from the feature class in the row, indicating that the model interprets them as more similar. However, it is important to note that a high degree of neighborhood coherence for a feature class may also imply the presence of multiple subclusters within that class.

1.1.2 Pairwise space comparison

Global comparison of pairwise distances

To perform a global comparison of pairwise distances across embeddings, we compute Spearman's rank correlation coefficient of pairwise distance values. This statistical measure allows us to evaluate how well the pairwise relationships between embeddings are preserved across different embedding spaces by comparing their ranking across different spaces.

To visualize, the pairwise distances in the two embedding spaces can be plotted against each other in a scatter plot.

Cross-space neighborhood similarity

To quantify and analyze local differences between two embedding spaces, we first identify the KNN for each sample in both spaces. As in previous approaches for embedding space comparison (15; 5; 29; 11), we compute the overlap of these KNN between the two spaces for each sample, enabling a comparison of how similar the local neighborhoods of each sample are in the two embedding spaces. We focus on the most and least similar sets of samples. In addition to previous work, we quantify the feature distribution within these subsets to characterise the differing samples and provide context.

1.2 Implementation in the EmmaEmb Python library

To make the proposed framework accessible, we developed a light-weight modular Python library called EmmaEmb. The library is centered around the `Emma` object, which handles embedding spaces, feature data, and intermediate computational results. Multiple functions on the `Emma` object enable analysis of feature distributions across embedding spaces, including feature alignment and neighborhood analysis, as well as pairwise space comparisons between different embedding spaces. EmmaEmb integrates functionalities from established libraries such as scikit-learn (23) and Plotly (17) to calculate the described analysis measures and offers intuitive options for visualizing the results. For distance measurement, EmmaEmb provides a variety of metrics, including Manhattan, Euclidean, and cosine distances, along with their normalized and scaled variants. The library is designed as a modular framework that can be extended for further types of analysis, such as clustering approaches or topological analysis. Additionally, the repository includes a script that facilitates the extraction of embeddings from the PLMs listed in Table 2, integrating sequence-chopping algorithms to process long sequences.

2. Experimental setup

2.1 Datasets

Two datasets were chosen to test the capabilities of our embedding space analysis framework.

DeepLoc

We utilized the training set of version 1.0 of the DeepLoc dataset, which consists of 11,231 protein sequences from eukaryotes annotated with subcellular localization information across 10 distinct classes (2). DeepLoc is a widely used benchmark dataset for evaluating protein subcellular localization prediction models (30; 16), offering a comprehensive resource for comparing the embedding space analysis framework's results to task-specific supervised ML model performance.

PLA2G2

The PLA2G2 dataset consists of protein sequences from the phospholipase A2 family, specifically the phospholipase A2 Isozyme Group II (PLA2G2) published in (18). The dataset includes protein sequences annotated across 7 different species and proteins are categorized into 11 different enzyme classes. To enable comprehensive data integration, only proteins with both

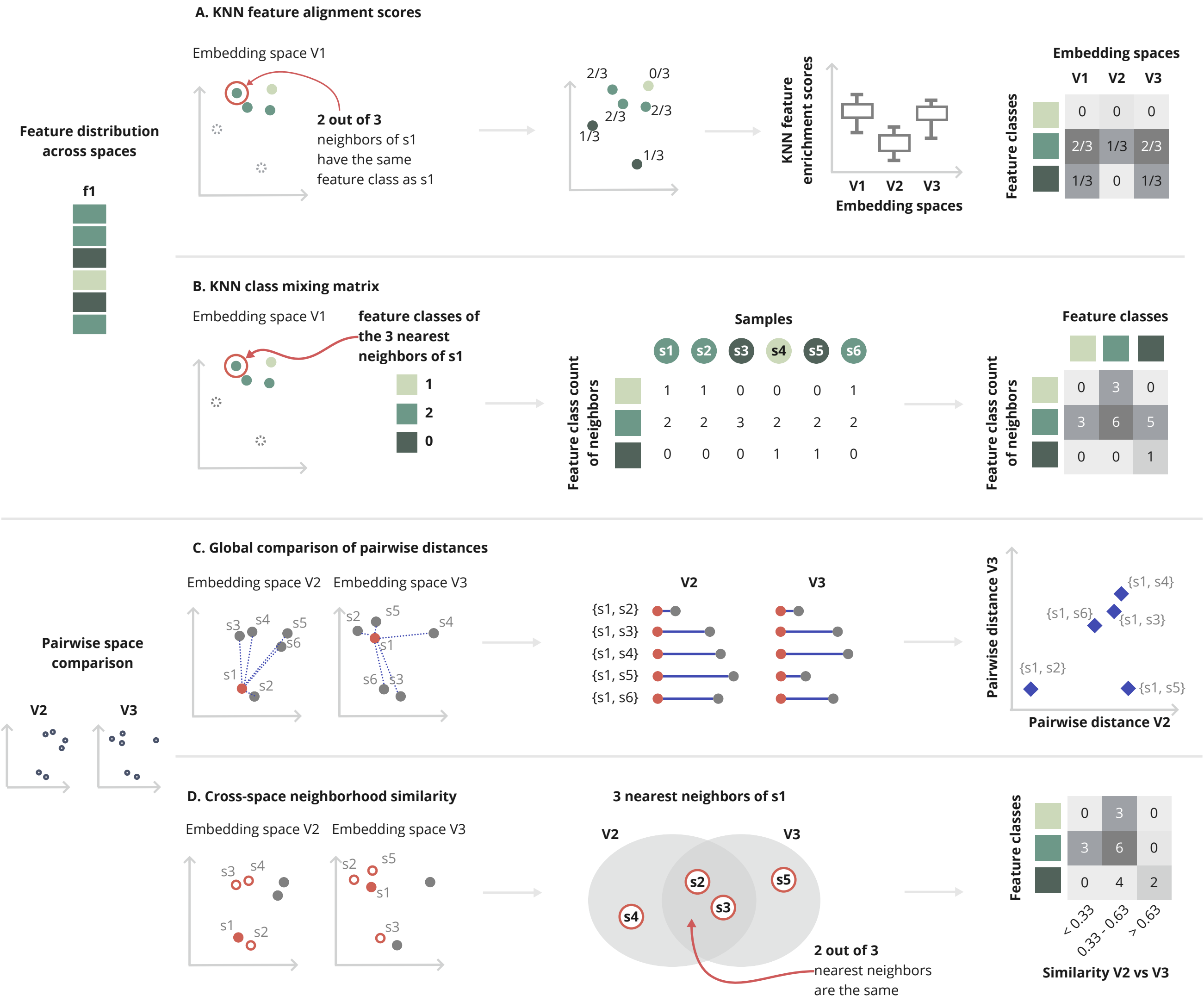


Fig. 2. Graphical overview of proposed methods for embedding space analysis. For the analysis of the feature distribution of a specific feature (*here*: feature $\mathbf{f_1}$) for a set of samples (*here*: $s_1 - s_6$) across embedding spaces (*here*: feature $\mathbf{V_1} - \mathbf{V_3}$) we propose two methods. Firstly, **A. KNN feature alignment scores**. The KNN feature alignment scores are calculated for each embedding within its embedding space, as a fraction of its k-nearest neighboring embeddings that share the same class label as the embedding itself. By comparing the distribution of KNN feature alignment scores across different embedding spaces, we can assess how closely embeddings of the same $\mathbf{f_1}$ class are to other embeddings of the same class. The score provides insights into how well an embedding model has learned to group embeddings with similar characteristics. Secondly, **B. KNN class mixing matrix**. The matrix is calculated separately for each embedding space (*here*: for V_1). For each embedding, the class distribution of class labels of its k-nearest neighbors is computed. The class distributions are then summed for all embeddings of the same class. The resulting matrix shows, for each class, the number of k-nearest neighbors of embeddings from this class for another class. This allows insights into which classes are close to each other and which are more differentiated within the embedding space. We also propose methods for pairwise embedding space comparison, such as **C. Global comparison of pairwise distances** (*here*: between embedding space V_2 and V_3). For this method, we first calculate all pairwise distances between embeddings within each embedding space. Then, we compare the correlation of pairwise distances, e.g. distance between s_1, s_2, across the two embedding spaces. This method provides an overview of the global similarity between embedding spaces and reveals groups of samples that are differently represented. **D. Cross-space neighborhood similarity**, in contrast, quantifies local differences between two embedding spaces. It is calculated for each sample as the number of k-nearest neighbors shared between V_2 and V_3. Identifying and characterizing samples with high divergence in their local neighborhood can help to uncover structural differences between the embedding spaces.

species and enzyme class information were included, resulting in a final set of 446 proteins. Compared to the DeepLoc dataset, the PLA2G2 dataset exhibits greater sequence similarity, making it more suitable for studying evolutionary changes.

2.2 Embeddings

Table 2 provides an overview of the transformer-based protein language models evaluated in the experiments. As some of these models have restrictions on the maximum sequence length, protein sequences longer than 1022 amino acids were divided into smaller chunks. Each chunk was processed with a minimum overlap of 300 amino acids to maintain sequence continuity. The embeddings for each chunk were then aggregated using

Table 2. Overview of the protein language models used in this study.

Model name	Ref	Version	# parameters	Embedding length	Training
Ankh	(9)	large	1.15B	1536	Only trained on sequence data
ProtT5	(10)	prot_t5_xl_uniref50	3B	1024	Only trained on sequence data
ProstT5	(20)	-	3B	1024	ProtT5 fine-tuned for translation between sequence and structure
ESM2	(22)	esm2_t36_3B_UR50D	3B	2560	Only trained on sequence data
ESMC	(31)	esmc-300m-2024-12	300M	960	Related to the multimodal ESM3 model (14), but optimised for representation learning

the mean of all chunk embeddings, resulting in a single unified representation for the full-length protein. As proposed by Brandes et al. (7), to ensure that the entire sequence is effectively represented, chunks were taken with overlapping windows from both sides, and if necessary, an additional window was included in the middle to fill any gaps.

Results and discussion

Distribution of protein localization information

We evaluate our embedding space analysis approach by comparing its results with those derived from training ML models on embeddings, focusing on the subcellular localization task. In particular, we consider results from light-attention networks, which have demonstrated superior performance compared to other approaches on embeddings from PLMs for this task (30). The networks were trained specifically for the subcellular localization task using the DeepLoc training set (30; 16), and we compare our findings to the results of these task-specific models reported on test sets. KNN computation in the embedding space analysis was conducted using $k = 100$ and the cosine distance.

To compare how well models capture the subcellular localization information we calculated the KNN feature alignment scores. According to the estimated distribution, ProtT5 has the highest scores on average, followed by Ankh, ESM2 and ProstT5 (Figure 3A). This ranking was statistically validated using one-sided Wilcoxon signed-rank tests (n = 11,231): ProtT5 vs ESM2 (p-value < 0.0001, ranked biserial correlation coefficient = 0.20), ESM2 vs Ankh (p-value $<$ 0.0001, ranked biserial correlation coefficient = 0.12), and Ankh vs ProtT5 (p-value < 0.0001 , ranked biserial correlation coefficient = 0.46). This ranking aligns with previously reported performance when the embeddings from these models were used with a trained light-attention network for this task (16). The observed consistency between the mean KNN feature alignment scores and ML-based performance rankings shows that embedding-based KNN feature alignment analysis can serve as a proxy for evaluating the informativeness of embeddings in downstream tasks.

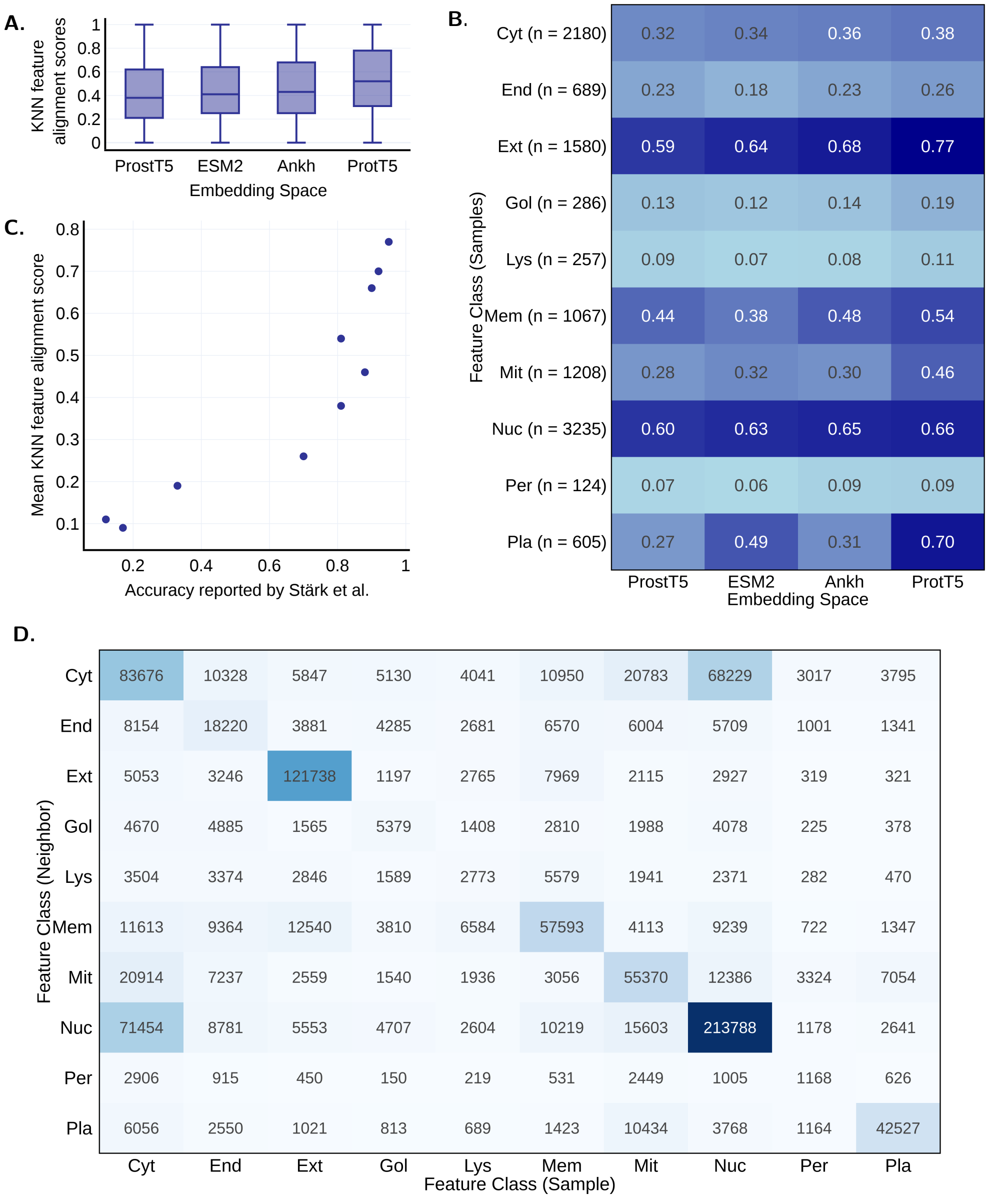
Next, we calculated mean aggregated KNN enrichment scores per embedding space, stratified by feature classes (Figure 3B). This approach highlights which feature classes are consistently well captured in the model's representations, such as "Extracellular"and "Nucleus", while also identifying models that capture specific classes more effectively, such as ProtT5 for "Plastid"and "Mitochondrion". Further analysis reveals a strong correlation between the mean KNN feature alignment scores of the ProtT5 embedding space for specific subcellular localization classes and the prediction accuracy achieved by a light-attention model trained on ProtT5 embeddings, as reported in (30). This relationship, illustrated in Figure 3C, is supported by a two-sided Spearman's rank correlation coefficient of 0.97 (p-value $<$ 0.0001). This analysis provides similar insights but at a more granular level, demonstrating that embedding space analysis can be used to approximate the models' overall understanding of not only of specific features but also for different feature classes.

Even more fine-grained, the differentiation of different feature classes within the ProtT5 embedding space are also examined using the proposed KNN class mixing matrix (see Figure 3D). For classes that are not yet well differentiated in the space, such as "Cytoplasm", it is possible to dive deeper into feature classes that proteins from this class are located nearby and might be "mixed"with. For example, proteins classified as "Cytoplasm"are in proximity to those classified as "Nucleus". This information can aid in more targeted training of models to better differentiate specific features and improve class separation in the embedding space. Again we compare the results with the results from the light-attention approach. For each class, we examine the closest neighboring class in the KNN class matrix and then identify the classes to which the proteins were most frequently misclassified in task-specific ML classification. We found that for all 8 of 10 classes, the closest neighboring class corresponded to the class where proteins were most often misclassified as belonging to that class in the task-specific ML model.

Our results support the idea that, with an increasing number of models and benchmarking tasks, embedding space analysis could play a key role in pre-selecting models for further, more computationally intensive, task-specific, and embedding-specific analyses. It is important to recognize that task-specific ML models may uncover task-relevant complex relationships between dimensions that might not be apparent in a global analysis on the entire embedding space. Adding to this, recent work comparing the performance of task-specific ML models trained on embeddings versus fine-tuning PLMs suggests that the superior performance of one PLM over another in a task-specific embedding approach does not necessarily predict its superiority after fine-tuning (26). Therefore, benchmarking for specific tasks remains essential for evaluating task-specific models.

Comparison of ESMC and ProtT5 embeddings for PLA2G2 proteins

In this section, we apply our pairwise space comparison to the PLA2G2 dataset. We compare the ProtT5 model against the recently published ESMC model. The two models differ in their training, parameter sizes, as well as embedding dimensions (see



Feature Class (Samples)	ProstT5	ESM2	Ankh	ProtT5
Cyt (n = 2180)	0.32	0.34	0.36	0.38
End (n = 689)	0.23	0.18	0.23	0.26
Ext (n = 1580)	0.59	0.64	0.68	0.77
Gol (n = 286)	0.13	0.12	0.14	0.19
Lys (n = 257)	0.09	0.07	0.08	0.11
Mem (n = 1067)	0.44	0.38	0.48	0.54
Mit (n = 1208)	0.28	0.32	0.30	0.46
Nuc (n = 3235)	0.60	0.63	0.65	0.66
Per (n = 124)	0.07	0.06	0.09	0.09
Pla (n = 605)	0.27	0.49	0.31	0.70

	Cyt	End	Ext	Gol	Lys	Mem	Mit	Nuc	Per	Pla
Cyt	83676	10328	5847	5130	4041	10950	20783	68229	3017	3795
End	8154	18220	3881	4285	2681	6570	6004	5709	1001	1341
Ext	5053	3246	121738	1197	2765	7969	2115	2927	319	321
Gol	4670	4885	1565	5379	1408	2810	1988	4078	225	378
Lys	3504	3374	2846	1589	2773	5579	1941	2371	282	470
Mem	11613	9364	12540	3810	6584	57593	4113	9239	722	1347
Mit	20914	7237	2559	1540	1936	3056	55370	12386	3324	7054
Nuc	71454	8781	5553	4707	2604	10219	15603	213788	1178	2641
Per	2906	915	450	150	219	531	2449	1005	1168	626
Pla	6056	2550	1021	813	689	1423	10434	3768	1164	42527

Fig. 3. Feature distribution analysis of subcellular localization feature in the DeepLoc dataset across the ProstT5, ESM2, Ankh, and ProtT5 models. KNN analysis is performed with k = 100 and cosine distance. **A.** Boxplots displaying the KNN feature distribution scores per model. Protein localization information is most closely represented in ProtT5, followed by ESM2, Ankh, and ProstT5. **B.** Heatmap of the mean KNN feature distribution score per embedding space and feature class. Some classes are strongly represented across all models, while others are captured by only specific models. **C.** Scatter plot comparing the accuracy of a supervised machine learning model trained on the ProtT5 embeddings (as reported by Stärk et al. (30)) on the x-axis and the mean KNN feature alignment scores for the ProtT5 embeddings on the y-axis. Each dot represents a different localization class, and a strong correlation is observed between these two measures. **D.** KNN class mixing matrix for the ProtT5 embedding space. Some classes are represented in closer proximity to each other than others. Cyt: Cytoplasm; End: Endoplasmic reticulum; Ext: Extracellular; Gol: Golgi apparatus; Lys: Lysosome/vacuole; Mem: Cell membrane; Mit: Mitochondrion; Nuc: Nucleus; Per: Peroxisome; Pla: Plastid.

Table 2). Our aim is to investigate whether these models exhibit differences in how they represent this set of enzyme proteins.

Global correlation analysis of the pairwise cosine distances between the embeddings of the models shows a high Spearman's rank correlation coefficient of 0.75 (p-value < 0.0001) between ESMC and ProtT5. This indicates a substantial similarity in how these two models represent relationships among the sequences overall.

When examining the scatter plot of pairwise cosine distances for embeddings of the ESMC and ProtT5 models, we observe that many data points align along a linear trend, consistent with the high correlation value. However, a distinct cluster of proteins deviates profoundly, with much greater pairwise distances in ProtT5 compared to ESMC (see Figure 4A.1). We could identify this driven by a subset of bird proteins as illustrated in Figure 4A.2. Notably, this difference in representation is not observed when comparing pairwise distances for other species, such as crocodiles (see Figure 4A.3). This analysis raises new questions about the models and why they have learned to represent a subset of bird proteins

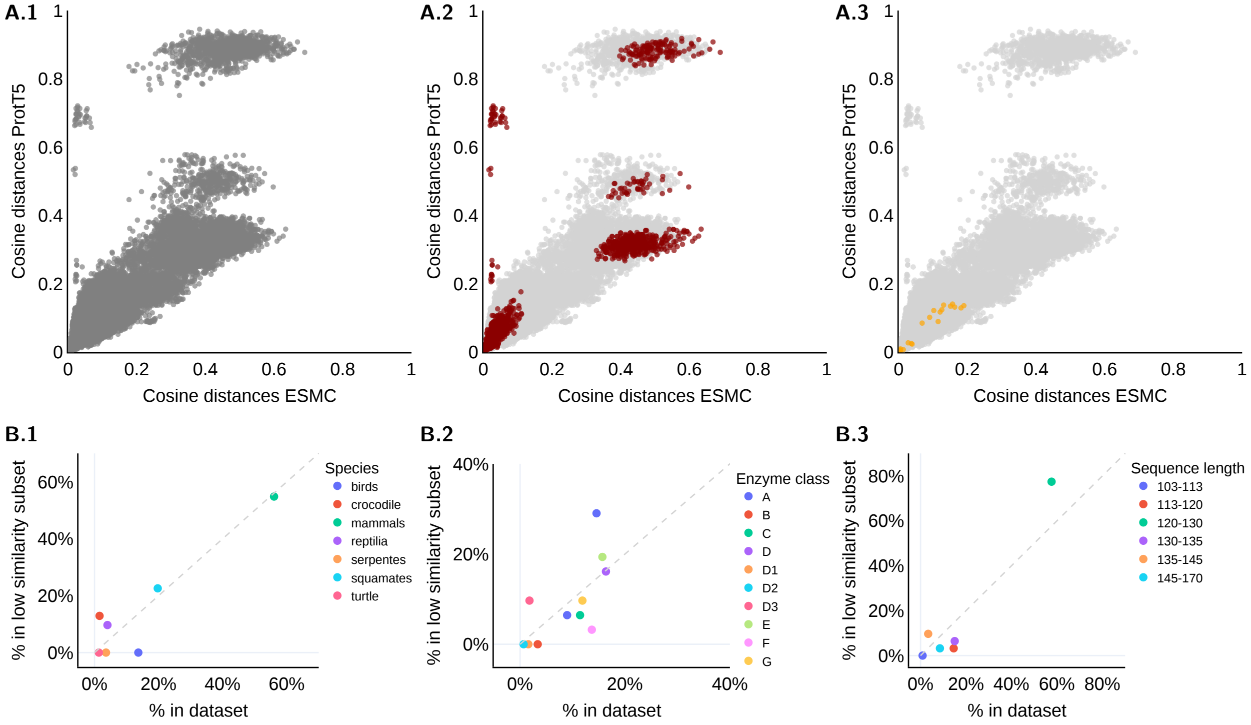


Fig. 4. A. The top row displays pairwise cosine distances between PLA2G2 embeddings using ESMC (x-axis) and ProtT5 (y-axis). **A.1** for all embeddings, **A.2** between embeddings of bird proteins (red), and **A.3** between embeddings of crocodile proteins (yellow). **B.** The bottom row illustrates the class distribution of proteins with 30% local neighborhood overlap (cosine distance, $k = 10$) between the ESMC and ProtT5 models, compared to overall dataset representation (x-axis) for the features: **B.1** species, **B.2** enzyme classes, and **B.3** sequence length.

differently. One possibility is that one of the models has learned finer distinctions between subgroups of bird species, potentially due to differences in training data.

As a final step, we compare local neighborhoods between the models and identified proteins with a local neighborhood similarity of less than 30% when comparing the ESMC and ProtT5 models. Analyzing the distribution of species, enzyme classes and amino acid sequence lengths (see Figure 4B). In comparison to the global analysis, local changes between the ESMC and ProtT5 models were not observed in bird proteins but rather in proteins from squamates and reptilia. This suggests that, even though the positions of points corresponding to bird species may have shifted within the embedding space, the local neighborhood information for bird sequences is preserved, indicating a consistent representation of their relationships within the space. This analysis highlights the importance of analyzing embedding spaces at both global and local levels, which will give partially complementary views.

Conclusion

In summary, we show that comparing embedding space analysis with supervised machine learning models trained on the embeddings provides similar insights into model learning, without requiring task-specific or model-specific training. While embedding space analysis is limited to the information captured in the overall embedding space without accounting for complex interactions or subspaces, it can play a key role in pre-selecting the growing number of models for further training.

Further, we explore the use of embedding space analysis to investigate discrepancies between PLMs on subsets of the data. This approach helps to identify discrepancies in the information captured by the models, which can reveal gaps in the training data or highlight complementary learned information.

Although our approach is demonstrated here with protein sequence embeddings, it is domain-agnostic. It can be applied to embedding spaces across foundation models in molecular biology and beyond, particularly in cases foundation models are combined with structured metadata.

Competing interests

No competing interest is declared.

Author contributions statement

P.F.R. conceived and designed the study with input from P.Y.S.. P.F.R. conducted the experiments and analyses. J.S. contributed to the development of the codebase used in the analysis. P.F.R. drafted the manuscript. P.Y.S., J.S., C.W.C., B.Y.R, H.O.H., S.I. provided critical feedback and input throughout the study. All author reviewed and approved the final version.

Acknowledgments

We would like to express our gratitude to Katharina Baum, Jakub Bartoszewicz, Melania Nowicka, Marta Lemanczyk, Theresa Hradilak, Eugenia Alleva, and Hilary Finucane for their valuable discussions and insightful input regarding this project. The project was supported by the Designing for

Sustainability research program, a joint initiative of the Morningside Academy for Design at MIT and the Hasso Plattner Institute, generously funded by the Hasso Plattner Foundation as well as the DFG, German Research Foundation (project number 459422098).

References

1. Matteo Aldeghi, David E. Graff, Nathan Frey, et al. Roughness of molecular property landscapes and its impact on modellability. *Journal of Chemical Information and Modeling*, 62(19):4660–4671, 2022.
2. José Juan Almagro Armenteros, Casper Kaae Sønderby, Søren Kaae Sønderby, et al. Deeploc: prediction of protein subcellular localization using deep learning. *Bioinformatics*, 33(21):3387–3395, 2017.
3. Tom Altenburg, Thilo Muth, Patrick van Zalm, et al. Foundation model enables interpretable open and error-tolerant searching for mass spectrometry-based proteomics. *bioRxiv*, 2025.
4. Žiga Avsec, Markus Weilert, Avanti Shrikumar, et al. Base-resolution models of transcription-factor binding reveal soft motif syntax. *Nature Genetics*, 53(3):354–366, 2021.
5. Angie Boggust, Brandon Carter, and Arvind Satyanarayan. Embedding comparator: Visualizing differences in global structure and local neighborhoods via small multiples. *Proceedings of the 27th International Conference on Intelligent User Interfaces*, 746–766, 2022.
6. Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, et al. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *Advances in Neural Information Processing Systems*, 29, 2016.
7. Nadav Brandes, Grant Goldman, Charlotte H. Wang, et al. Genome-wide prediction of disease variant effects with a deep protein language model. *Nature Genetics*, 55(9):1512–1522, 2023.
8. Frances Ding and Jacob Steinhardt. Protein language models are biased by unequal sequence sampling across the tree of life. *bioRxiv*, 2024.
9. Ahmed Elnaggar, Hazem Essam, Wafaa Salah-Eldin, et al. Ankh: Optimized protein language model unlocks general-purpose modelling. *arXiv*, 2023.
10. Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, et al. Prottrans: Towards cracking the language of life's code through self-supervised deep learning and high performance computing. *bioRxiv*, 2020.
11. France-Travail. embcompare. https://github.com/France-Travail/embcompare
12. David E. Graff, Edward O. Pyzer-Knapp, Kirk E. Jordan, et al. Evaluating the roughness of structure–property relationships using pretrained molecular representations. *Digital Discovery*, 2:1452–1460, 2023.
13. Viktor Hangya, Silvia Severini, Radoslav Ralev, et al. Multilingual word embeddings for low-resource languages using anchors and a chain of related languages. *arXiv*, 2023.
14. Thomas Hayes, Roshan Rao, Halil Akin, et al. Simulating 500 million years of evolution with a language model. *bioRxiv*, 2024.
15. Florian Heimerl, Christoph Kralj, Torsten Möller, et al. embcomp: Visual interactive comparison of vector embeddings. *IEEE Transactions on Visualization and Computer Graphics*, 28(8):2953–2969, 2022.
16. Michael Heinzinger, Konstantin Weissenow, Joaquin Gomez Sanchez, et al. Bilingual language model for protein sequence and structure. *NAR Genomics and Bioinformatics*, 6(4):lqae150, 11 2024.
17. Plotly Technologies Inc. Collaborative data science, 2015.
18. Ivan Koludarov, Timothy NW Jackson, Vivek Suranse, et al. Reconstructing the evolutionary history of a functionally diverse gene family reveals complexity at the genetic origins of novelty. *bioRxiv*, 2020.
19. Joseph B. Kruskal. Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29(2):115–129, 1964.
20. Maxat Kulmanov, Francisco Guzmán-Vega, Paula Duek Roggli, et al. Protein function prediction as approximate semantic entailment. *Nature Machine Intelligence*, 6:1–9, 02 2024.
21. Zhong Li, Qiaolin Ding, and Weihua Zhang. A comparative study of different distances for similarity estimation. *Intelligent Computing and Information Science*, 483–488, 2011.
22. Zeming Lin, Halil Akin, Roshan Rao, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.
23. Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(85):2825–2830, 2011.
24. PentoAI. Vectory. https://github.com/pentoai/vectory
25. Sam T. Roweis and Lawrence K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.
26. Robert Schmirler, Michael Heinzinger, and Burkhard Rost. Fine-tuning protein language models boosts predictions across diverse tasks. *Nature Communications*, 2024.
27. Tobias Senoner, Tobias Olenyi, Michael Heinzinger, et al. Protspace: a tool for visualizing protein space. *Journal of Molecular Biology*, 2025.
28. Dan Shiebler. repcomp. https://github.com/dshieble/RepresentationComparison
29. Venkatesh Sivaraman, Yiwei Wu, and Adam Perer. Emblaze: Illuminating machine learning representations through interactive comparison of embedding spaces. *Proceedings of the 27th International Conference on Intelligent User Interfaces*, 418–432, 2022.
30. Hannes Stärk, Christian Dallago, Michael Heinzinger, et al. Light attention predicts protein location from the language of life. *Bioinformatics Advances*, 1(1):vbab035, 2021.
31. ESM Team. Esm cambrian: Revealing the mysteries of proteins with unsupervised learning. 2024.
32. Joshua B. Tenenbaum, Vin de Silva, et al. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.
33. Ekaterina Vylomova, Laura Rimell, Trevor Cohn, et al. Take and took, gaggle and goose, book and read: Evaluating the utility of vector differences for lexical relation learning. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1671–1682, 2016.
34. F. Alexander Wolf, Philipp Angerer, and Fabian J. Theis. SCANPY: large-scale single-cell gene expression data analysis. *Genome Biology*, 19(1):15, 2018.
35. Wayland Yeung, Zhongliang Zhou, Liju Mathew, et al. Tree visualizations of protein sequence embedding space enable improved functional clustering of diverse protein superfamilies. *Briefings in Bioinformatics*, 24(1):bbac619, 2023.