frontiers | Frontiers in Bioinformatics

TYPE Original Research
PUBLISHED 17 November 2022
DOI 10.3389/fbinf.2022.1033775

OPEN ACCESS

EDITED BY
Rachel Kolodny,
University of Haifa, Israel

REVIEWED BY
Castrense Savojardo,
University of Bologna, Italy
Arun Prasad Pandurangan,
University of Cambridge,
United Kingdom

*CORRESPONDENCE
Konstantin Schütze,
schuetze@in.tum.de

SPECIALTY SECTION
This article was submitted to Protein Bioinformatics,
a section of the journal
Frontiers in Bioinformatics

RECEIVED 31 August 2022
ACCEPTED 31 October 2022
PUBLISHED 17 November 2022

CITATION
Schütze K, Heinzinger M, Steinegger M and Rost B (2022), Nearest neighbor search on embeddings rapidly identifies distant protein relations.
Front. Bioinform. 2:1033775.
doi: 10.3389/fbinf.2022.1033775

Nearest neighbor search on embeddings rapidly identifies distant protein relations

Konstantin Schütze[1]*, Michael Heinzinger[1,2], Martin Steinegger[3,4] and Burkhard Rost[1,5]

[1]TUM (Technical University of Munich) Department of Informatics, Bioinformatics & Computational Biology—i12, Munich, Germany, [2]TUM Graduate School, Center of Doctoral Studies in Informatics and its Applications (CeDoSIA), Garching, Germany, [3]School of Biological Sciences, Seoul National University, Seoul, South Korea, [4]Artificial Intelligence Institute, Seoul National University, Seoul, South Korea, [5]Institute for Advanced Study (TUM-IAS), Germany & TUM School of Life Sciences Weihenstephan (WZW), Freising, Germany

Since 1992, all state-of-the-art methods for fast and sensitive identification of evolutionary, structural, and functional relations between proteins (also referred to as "homology detection") use sequences and sequence-profiles (PSSMs). Protein Language Models (pLMs) generalize sequences, possibly capturing the same constraints as PSSMs, e.g., through embeddings. Here, we explored how to use such embeddings for nearest neighbor searches to identify relations between protein pairs with diverged sequences (remote homology detection for levels of <20% pairwise sequence identity, PIDE). While this approach excelled for proteins with single domains, we demonstrated the current challenges applying this to multi-domain proteins and presented some ideas how to overcome existing limitations, in principle. We observed that sufficiently challenging data set separations were crucial to provide deeply relevant insights into the behavior of nearest neighbor search when applied to the protein embedding space, and made all our methods readily available for others.

KEYWORDS

homology search, protein embeddings, language models, nearest neighbor search, remote homology detection

Abbreviations: 3D, three-dimensional; AI, Artificial Intelligence; BFD, Big Fantastic Database (); CATH, hierarchical classification of protein 3D structures in Class; Architecture, Topology and Homologous superfamily (); DL, Deep Learning; EAT, Embedding-based Annotation Transfer; EI, evolutionary information; embeddings, fixed-size vectors derived from pre-trained pLMs; ESM-1b, pLM from Facebook dubbed Evolutionary Scale Modeling (); FNN, Feed-forward Neural Network; HMM, Hidden Markov Model; HMMer, particular method for HMM-profile alignments (); LM, Language Model; MMseqs2, fast database search and multiple sequence alignment method (); MSA, Multiple Sequence Alignment; NLP, Natural Language Processing; PDB, Protein Data Bank (Burley et al., 2017); Pfam, Protein family database (); PIDE, percentage pairwise sequence identity; pLM, protein Language Model; ProtBERT, pLM () based on the LM BERT; ProtT5, pLM () based on the LM T5; PSSM, position-specific scoring matrix (also dubbed profile); SOTA, state-of-the-art.

Introduction

Homology detection

Any investigation of a query protein, Q, beginning with its sequence starts by the identification of evolutionary, structural, and functional relations between Q and all other proteins for which relevant experimental annotations exist. This investigation is often, and slightly misleadingly, labelled as "homology detection" inspired by the terminology introduced to describe the analogy of organs between different species (Owen, 1848). Homology as a term describing the similarity between proteins typically inherits the concept of "related by evolution" from the original comparison of species (Darwin, 1859). In practice, "related by evolution" is often replaced by "similar (or identical) structure (or function)" as "structural similarity" is less ambiguous to quantify than "evolutionary relation." This ambivalence also pertains to our view of "protein space": Maps showing relations between proteins are specific to particular definitions, e.g., the map of protein structure domains, or of functional units, or of evolutionary relations. The reproducibility of such maps increases with quantifiability, and is highest for protein structure because the knowledge of structure enables parsing proteins into their compact, independently foldable constituents, namely structural domains. Searches for relations in protein space root almost any drug development and are crucial for the success of the breakthrough prediction of protein structure prediction by AlphaFold2 (Jumper et al., 2021; Mirdita et al., 2021; Marx, 2022).

From sequence to feature similarity

With growing databases, speed has become THE major challenge for methods detecting homology by comparing the sequence of a query protein Q to all sequences in a database DB. All successful fast solutions from the classic BLAST/PSI-BLAST, over to MMseqs2 and Diamond2, (Altschul et al., 1997; Steinegger and Söding, 2017; Buchfink et al., 2021) essentially follow three steps. 1) Fast: Initialize search using sequence fragments with typically 3–10 consecutive residues (k-mer), i.e., by finding database sequences with k-mers identical or near-identical to the query. 2) Slower: Expand short k-mer matches (hits) through un-gapped alignment. 3) Slowest: Refine alignment for subset of Q-DB pairs with un-gapped scores above a predefined threshold (Step #2) through resource-intensive Smith-Waterman (Smith and Waterman, 1981). Ultimately, the first two steps pre-filter the finding of homologs, while the third generates the actual alignment yielding an E-value for each hit. This E-value estimates how many hits with identical are expected by chance (Karlin and Altschul, 1990). Following others, MMseqs2 [(Steinegger and Söding, 2017), Supplementary Online Material, SOM "Design of sensitivity benchmark"] measured the success in detecting homologs through a score referred to as AUC1, namely the fraction of annotated homologs found until the first non-homolog. Similar to other measures scoring search success, AUC1 depends heavily on the size of DB and the particular relation equated with homology, i.e., results differ between aiming at identifying pairs with similar structure, or similar function, or related in evolution, and given the immense diversity in definitions for function, AUC1-like measures can easily differ by an order of magnitude depending on the precise definition for function (Rost, 2002).

Alignments between pairs of sequences (also referred to as pairwise or sequence-sequence) unravel only simple connections, in evolutionary terms, the homology between proteins from closely related organisms. In order to intrude deeper into the twilight zone of sequence comparisons (Doolittle, 1986; Rost, 1999; Yona and Levitt, 2002), we need to find a family of related proteins through pairwise alignments, compile a profile or position-specific scoring matrix (PSSM) from this family, and then use the profile for more fine grained sequence-profile alignments (e.g. Clustal (Higgins and Sharp, 1989) or PSI-BLAST (Altschul et al., 1997)). The signal distinguishing between related and unrelated proteins becomes obfuscated upon entering the twilight zone; in fact, the transition from what we may call "daylight" to twilight zone is described by a rapid order-of-magnitude signal loss akin of a phase-transition (Rost, 1999). Profile-based searches intrude into the twilight zone, in particular, methods based on Hidden Markov Models (HMMs) as introduced for protein comparisons almost 30 years ago (Haussler et al., 1993; Krogh et al., 1994) and perfected through HHblits (Remmert et al., 2012) and Jackhmmr (Johnson et al., 2010).

Even more powerful than sequence-profile are profile-profile comparisons using profiles for query and database (Remmert et al., 2012). While some relations in this realm may no longer be indicative of evolutionary connections, at least many of the relations obtained by comparing the three-dimensional (3D) structures of proteins reveal that many relations are likely indicative of evolutionary connections so distant that even advanced sequence-based alignment methods fail to unravel those (Orengo et al., 2001; Nepomnyachiy et al., 2017). Thus, the identification of relations in the lower end of the twilight zone is often referred to as remote homology detection. Even profile-profile comparisons usually fail to intrude even further, namely the midnight zone in which sequences have diverged to random levels (5–10% pairwise sequence identity, PIDE) (Rost, 1997; Friedberg et al., 2000; Nepomnyachiy et al., 2017). In fact, most pairs of proteins with similar structures populate this realm (Rost, 1997; Friedberg et al., 2000).

Profile-based methods vary in speed, from the highly optimized HHblits (Remmert et al., 2012) averaging 2 min per query against UniRef30 (Remmert et al., 2012), to the lightning fast iterated MMseqs2 profile-aligning in sub-seconds on

UniRef90 (almost three orders faster than HHblits). Runtime details crucially depend on parameter choices such as "number of hits reported".

Protein Language Models capture crucial constraints

Faster computer hardware (largely GPUs and TPUs), better algorithms in Machine Learning, and big data combined to leap Natural Language Processing (NLP). In analogy, protein Language Models (pLMs) use large databases of raw protein sequences to implicitly understand the language of life (Alley et al., 2019; Bepler and Berger, 2019; Heinzinger et al., 2019; Elnaggar et al., 2021; Ofer et al., 2021; Rives et al., 2021). Indeed, many pLMs essentially needed access to sequence collections ten times larger than UniProt (Suzek et al., 2015; UniProt Consortium, 2017), namely BFD (Steinegger and Söding, 2018; Steinegger et al., 2019). Some pLMs additionally include supervised training (Bepler and Berger, 2019; El-Gebali et al., 2019). The values from the last hidden layers of the pLMs typically are extracted as "the embeddings of the pLM." For the pLM ProtT5 (Elnaggar et al., 2021), in particular, these embeddings have 1,024 dimensions for each residue in the protein (each protein position). The mean over all per-residue embeddings in a protein ($\frac{1}{L}\sum_{i}^{L}\text{embedding}_{di}$, with L as the protein length and embedding_{di} as the embedding of residue i in dimension d) yields a new per-protein embedding (global average pooling) of the same dimension (1024 days for ProtT5). We used this per-protein embedding as feature for our search.

Embeddings from pLMs capture information beyond sequence similarity and can help to detect close and remote homologs (Rao et al., 2019; Littmann et al., 2021a; Littmann et al., 2021b; Rives et al., 2021; Heinzinger et al., 2022). The similarity between protein-pairs in terms of embedding and sequence space are only weakly correlate which allows embedding-based annotation transfer (EAT) even for proteins with different sequences (PIDE<20%) (Littmann et al., 2020; Heinzinger et al., 2022). The per-residue embeddings as sole input to relatively shallow subsequent AI improve per-residue predictions of secondary structure (Elnaggar et al., 2021), inter-residue distance (Weissenow et al., 2022), 3D structure (Weissenow et al., 2022), and even residue-conservation and effects of sequence variation (Marquet et al., 2021; Dunham et al., 2022) beyond top prediction methods using evolutionary information from MSAs. Although falling substantially short of AlphaFold2 (Jumper et al., 2021). Per-protein embeddings outperform the best MSA-based methods in the prediction of sub-cellular location (Staerk et al., 2021), signal peptides (Teufel et al., 2021) and binding residues (Littmann et al., 2021c).

Nearest neighbor search through pLM embeddings

To search in embedding space, we want to find the k embeddings in a dataset most similar to our query given a distance metric. This is known as nearest neighbor search. As determining the exact nearest neighbors becomes intractable in high-dimensional spaces (Slaney and Casey, 2008), we applied approximate nearest neighbor search (k-nn) that is well established in domains including image recognition (Liu et al., 2007; Li et al., 2019), recommender systems (Bernhardsson, 2020) and NLP (Khandelwal et al., 2019). Modern indexing techniques such as Hierarchical Navigable Small World Graphs (Malkov and Yashunin, 2018) or Product Quantization (Jegou et al., 2010), as well as, approaches building upon those two (Babenko and Lempitsky, 2014; Baranchuk et al., 2018; Matsui et al., 2018) handle billion-scale datasets, suggesting the applicability to searching and/or clustering databases such as TrEMBL with 195 M sequences (11/2020 used here) (UniProt Consortium, 2017) or even the entirety of BFD (Steinegger and Söding, 2018; Steinegger et al., 2019).

Standard of truth: CATH and Pfam

We benchmarked on two databases: CATH (Orengo et al., 1997; Sillitoe et al., 2019) and Pfam (Bateman et al., 2000). CATH is created by three main steps: 1) parse all proteins of known 3D structure taken from the PDB (Protein Data Bank (Burley et al., 2017)) into compact domains. 2) Align all domains to each other by methods comparing 3D structures, i.e., structural alignment techniques (Orengo et al., 1992; Kolodny et al., 2005). 3) Proteins of unknown 3D structure are aligned by HMMer (Finn et al., 2011) to the 3D-aligned domain seeds forming the four classes of CATH: C (class), A (architecture), T (topology), H (homologous family). The Pfam database (El-Gebali et al., 2019) collects protein families without using 3D structure information. Consequently, Pfam family seeds are much shorter than structural domains (Liu and Rost, 2003), incidentally, also built using HMMer (Finn et al., 2011).

Advantages of pLMs

The key advantage of pretrained pLMs is that might implicit capture the same constraints that shaped evolution. Could this same advantage be harnessed to also revolutionize sequence comparisons? Here we analyzed to which extent alignments using the generalized sequences as found in embeddings might be competitive with traditional sequence-based approaches.

Methods

Data set 3D: CATH20

We used a redundancy-reduced version of CATH v4.2.0 (Orengo et al., 1997; Sillitoe et al., 2019) provided by the CATH team, which was optimized to contain as many sequences as possible. It was created by eliminating pairs with ≥20% PIDE and ≥60% overlap with the longest protein and consists of 14,433 domain sequences in 5,125 families. We computed embeddings for both datasets with the python api of bio_embeddings v0.2.0 (Dallago et al., 2021). The full 14,433 domains served as target database (to search against with the query), and 10,874 domains from the subset of 1,566 families with more than one member served as queries (to search with). We deemed a result as correct if the top hit (excluding self-hits) belonged to the same Pfam/CATH family as the query.

Data set 1D: Pfam20

In practice, many users will either not know or use single domains for their searches because as many as 80% of all proteins may have several domains (Liu et al., 2004), and because for the target database users would be limited to domain-based resources such as CATH (Orengo et al., 1997; Sillitoe et al., 2019) or Pfam (Bateman et al., 2000; El-Gebali et al., 2019). To flip this perspective: most expert users will likely use one of those two at some point and will have some idea about the composition of structural domains in their protein, in particular, given the AlphaFold2 predictions for hundreds of millions of proteins (Tunyasuvunakool et al., 2021) that simplify separating proteins into structural domains.

We proxied searches with full-length proteins through the Pfam-based dataset. To have enough proteins with matching domains without over-representing large families, we picked 20 domains from each Pfam family with at least 20 members and accumulated all of those proteins into a set dubbed Pfam20. For these, we retrieved the full-length proteins for each Pfam-region. This provided 313,518 proteins and the set of all Pfam domain annotations for each protein. The task was to find all proteins that have a Pfam domain from the same family as any of the Pfam-domains annotated for the full-length query. This sampling ensured each query to have at least 20 correct hits. We searched this set in all-against-all fashion, considering a query-hit pair as correct if the two had at least one Pfam annotation in common. For k-nn, we retrieved the 300 nearest neighbors for each query. As most queries had 20 correct hits, the AUC1 (area under curve until first incorrect match) fell between 0 and 20 of 20.

Sequence alignment

MMseqs2 version 13 (Steinegger and Söding, 2017) served as state-of-the-art (SOTA) for combining speed and sensitivity. We searched with a sensitivity of 7.5 (−s 7.5) and accepted hits with E-values ≤ 10^{4} and a prefilter limit of 300 hits. For the CATH20 set, these settings found the correct hit in all but 11 of the 14,433 queries.

Protein Language Model embeddings

SeqVec (Heinzinger et al., 2019) applies the bidirectional Long Short-Term Memory layer (LSTM) architecture of ELMo (Peters et al., 2018) architecture from NLP to proteins, yielding embeddings that are 1,024 dimensional 32-bit float vectors. ProtTrans (Elnaggar et al., 2021) abbreviates a collection of pLMs all of which use a transformer architecture and have more free parameters than SeqVec (ProtBert 420 M—with M for million, ProtAlbert 224M, ProtT5 3B—with B for billion, vs. SeqVec 93 M) and were trained on the much larger BFD dataset (UniRef50 50 M) (Steinegger and Söding, 2018; Steinegger et al., 2019; Elnaggar et al., 2020). While ProtBert has the same embedding dimensionality as SeqVec (d = 1,024), ProtAlbert has 4096-dimensional embeddings. From the ESM-series of pLMs (Rives et al., 2021), we benchmarked ESM and ESM1b with 670M and 650 M parameters respectively and 1,280 embedding dimensions, which were trained on the 250 million sequences. ProtT5 (Elnaggar et al., 2021), based on the Text-to-Text Transfer Transformer [T5, (Raffel et al., 2019)], is a model consisting of an encoder and a decoder, of which we used only the decoder. The original ProtT5 model was only pretrained on BFD (ProtT5 BFD), a later version was finetuned on UniRef 50 (ProtT5 XL U50). We used it in half precision mode (fp16) for speedier embedding generation without loss of accuracy (Elnaggar et al., 2021).

Embedding-based clustering

Step 1: k-nn index: We constructed an HNSW index (Hierarchical Navigable Small World Graphs) of M = 42 (Malkov and Yashunin, 2018), and searched with efSearch = 256 using faiss (Johnson et al., 2019). While storing the embeddings for our datasets required 642 MB, the HNSW index, which included both the embeddings and the HNSW graphs, required 1.4 GB.

Step 2: k-nn score: As basis for the combined method, we used negative log-transformed E-values for the hits from MMseqs2 and cosine similarities for the embeddings. As we chose E < 0.1 and since cosine scores were between 0 and 1, the transformed E-values were always larger than the cosine similarities, forming our new combined scored. However, for the raw E-values lower is better while for the k-nn scores the opposite held. The combined score used the same, more simplistic, normalization of "higher is better."

Hardware

All benchmarks were performed on a machine with 2 Intel Xeon Gold 6248 with a total of 80 threads, 400 GB RAM and an Nvidia Quadro RTX 8000 with 46 GB memory. The GPU was only used to generate embeddings.

Implementation and availability

The k-nn search was performed using the python interface of faiss version 1.6.3 (Johnson et al., 2019). To align the k-nn hits, we wrote them into a MMseqs2 prefilter database and ran MMseqs2 align with an E-value cutoff of 10,000 (−e 10,000). We made the code that reproduced all figures and tables as well as all figure data available at https://github.com/konstin/knn-for-homology, together with the raw data of the figures.

Performance measures

Top1(CATH20): For the CATH20 dataset, we searched each of the 10,874 domains from a family with more than one representative (query) against the 14,432 other domains in the CATH20 (database) with an exhaustive nearest neighbor search with cosine similarity. For each query, we considered the search result correct if the top hit was from the same homologous superfamily as the query. We considered only the top/first hit as it is has been suggested to be the most relevant for homology-based inference (Littmann et al., 2020). We reported two values reflecting performance accuracy. 1) Raw accuracy was defined as the fraction of correct hits:

$$QrawTop1\,(Dataset) = \frac{1}{size\,(CATH20)} \sum_{q \in CATH20} correct\,(q_i) \quad (1)$$

where DataSet were CATH20 and Pfam20, size (DataSet) gave the number of queries in the data set, qi was the i-th query in DataSet. (2) As another measure, we considered the normalized accuracy normalized by family size to remove the bias from large families:

$$QnormTop1\,(Dataset) = \frac{1}{num\,(families)} \sum_{f \in families} \frac{1}{size\,(family)} \sum_{q \in f} correct\,(f_i) \quad (2)$$

The latter was obtained by computing the accuracy for each family separately and taking the mean over these accuracies. These two scores have also been referred to as micro and macro average. For CATH20, we have reported both measures because the two often differed substantially. While the normalized accuracy removes the bias towards large queries, the raw accuracy adequately represents the abundance of sequences in the redundancy-reduced set. AUC1(Pfam20): For Pfam20 for each query, we recorded the fraction of true positive hits detected up to the first false positive. Errors were estimated using 500 rounds of bootstrapping and extracting the 95% confidence interval.

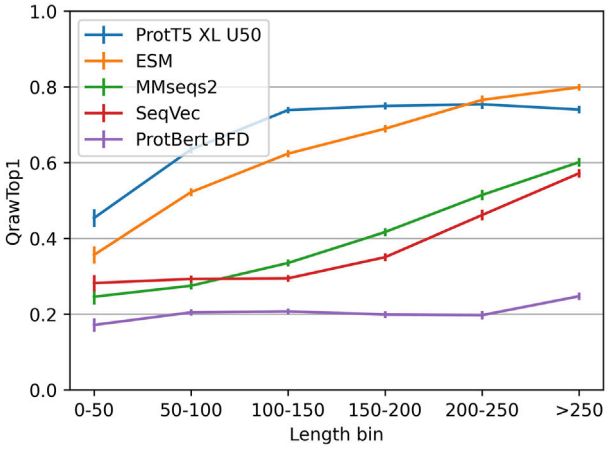

FIGURE 1
Performance better for longer proteins. The vertical y-axis QrawTop1 (Eq. 1) reflect the performance for proteins from the length interval specified on the horizontal x-axis (non-cumulative bins). While the embeddings from ProtT5 and ESM saturated quickly, the sequence alignment method MMseqs2 and the SeqVec-embeddings correlated more linearly with protein length.

Results

Single domains: CATH (set CATH20)

Proof-of-principle: Successful identification of domains

The embedding-based nearest neighbor search (k-nn) found more diverged/sequence-dissimilar homologous domains than MMseqs2, the state-of-the-art (SOTA) for fast and sensitive sequence-based search. Embeddings from more advanced pLMs clearly outperformed those from simpler pLMs (Figure 1; Table 1). This finding is easiest to illustrate using ">" to mean "better than" for the major pLMs (Figure 1), we observed: ProtT5 > ESM1b > ProtAlbert > ProtXLNet > ProtBert. All these differences were statistically significant at the 95% confidence interval (CI). While MMseqs2 outperformed the less advanced pLMs, it was outperformed by ProtT5 and ESM1b (Tables 1, 2). SeqVec's LSTM1 layer performed better than any combination of three layers (Supporting Online Material (SOM), Supplementary Figure S3).

In terms of embedding measure (proximity/distance of two embeddings vectors of, e.g., 1,024 dimensions for ProtT5, representing query and database hit), the cosine similarity consistently outperformed the Euclidean distance, albeit only slightly (Supplementary Table S1). For instance, the normalized accuracy (QnormTop1) for ProtT5 dropped from 57.5% ± 1.6% (using cosine similarity to measure that the query-hit vector are

TABLE 1 Performance on CATH20[a].

Methods	QnormTop1	QrawTop1
Combined method	62.0% ± 1.4%	73.0% ± 0.8%
ProtT5	**57.5% ± 1.6%**	**70.9% ± 0.9%**
ProtT5 BFD	54.3% ± 1.8%	70.8% ± 0.9%
ESM1b	47.9% ± 2.0%	68.5% ± 0.9%
ESM	43.5% ± 2.0%	65.2% ± 0.9%
MMseqs2	**34.5% ± 1.2%**	**40.3% ± 1.0%**
MMseqs2 E < 0.01	26.1% ± 0.9%	28.2% ± 1.0%
ProtAlbert BFD	20.2% ± 1.3%	34.7% ± 0.9%
SeqVec LSTM1	18.6% ± 1.2%	37.4% ± 0.9%
SeqVec Sum	18.2% ± 1.4%	37.5% ± 0.9%
PLUS	17.7% ± 1.3%	36.0% ± 0.9%
SeqVec LSTM2	17.6% ± 1.3%	36.7% ± 0.9%
ProtXLNet UniRef100	15.4% ± 1.2%	34.2% ± 0.9%
ProtBert BFD	12.7% ± 0.9%	21.0% ± 0.8%
UniRep	9.1% ± 0.9%	22.4% ± 0.8%
SeqVec CharCNN	2.7% ± 0.4%	4.2% ± 0.4%
AA composition	2.5% ± 0.3%	4.0% ± 0.4%
CPCProt	2.1% ± 0.4%	3.9% ± 0.4%

[a]Data set: CATH20 (redundancy reduced at PIDE≤20); performance measures (columns): QrawTop1 (Eq. 1) reflected the percentage of queries for which the first hit was correct (same CATH, identifier), while QnormTop1 normalized by family size (Eq. 2); methods (rows, sorted by QnormTop1): ProtTrans (ProtT5, Prot5 BFD, ProtBert BFD, ProtAlbert BFD, ProtXLNet, UniRef100) (Elnaggar et al., 2021), ESM (Rives et al., 2021), MMseqs2 (Steinegger and Söding, 2017), SeqVec (Heinzinger et al., 2019), UniRep (Alley et al., 2019), CPCProt (Lu et al., 2020), combined method: MMseqs2 E < 0.01 + ProtT5 UniRef50; error estimates: the ± values provide the range of the 95% confidence interval corresponding to 1.96 standard errors; bold letters: highlighting the comparison between embedding-based and alignment-based lookup.

similar) to 55.3% ± 1.7% (using the Euclidean distance to measure the embedding similarity).

To clarify the novel contribution of embeddings, we replaced all hits from MMseqs2 with E-value > T and those with no match (11 cases) with hits from knnProtT5. The resulting combined search results (dubbed "combined method") outperformed both methods over a wide range of thresholds T (Supplementary Figure S2, S3). For instance, at an E-value threshold of $T = 0.01$, the raw accuracy increased by over two percentage points to 73.0% (Table 1, first row). The combined method outperforms both methods' accuracy over a large range of cutoffs (Supplementary Figure S1, S2).

Hypothetical best of both

If we could by some unknown procedure pick only the correct hits from each method, we would reach QrawTop1 = 78.2%. This implied a higher increase from combined method to hypothetical than from knnProtT5 to combined method, but much less than from MMseqs2 to combined. This hypothetical perfect merger marked the theoretical limit for the combined approach, which implied that the simple E-value threshold of $T = 0.01$ already reached almost half of the improvement theoretically possible.

Factors determining performance

To better understand the complementarity of alignment-based and embedding-based approaches, we zoomed into strengths and weaknesses of each approach. For MMseqs2 and SeqVec accuracy clearly correlated with protein length ("shorter proteins better"), while for ProtT5 and, to a slightly lesser extent, for ESM, the accuracy saturated for longer proteins. ProtBert-BFD, on the other hand, performed rather consistently across the spectrum of protein length (Figure 1). By design, CATH20 considered only single domains. We observed only a limited correlation between cosine similarities and E-values with a Spearman's ρ of −0.17 and Pearson Correlation Coefficient between cosine similarities and the log-transformed E-values of −0.14. This confirmed prior results (Littmann et al., 2020).

Full-length proteins: Pfam regions (set Pfam20)

Embedding-based knnProtT5 alone not competitive

The above assessment focused on the comparison of single-domain proteins or single domains. The Pfam20 (Methods) benchmark pulled in full-length proteins (still compared to

TABLE 2 Class imbalance for CATH hierarchy[a].

	Alpha	Beta	Alpha/beta	Few secondary structures
Number of queries	2,668	2,328	5,773	105
Number of targets	3,987	3,159	7,105	182
QrawTop1 family knnProtT5	65.6%	72.9%	73.7%	53.3%
QrawTop1 family MMseqs2	35.9%	36.6%	43.8%	45.7%
QrawTop1 class knnProtT5	93.1%	92.1%	95.7%	58.1%
QrawTop1 class MMseqs2	57.6%	56.7%	79.3%	45.7%

[a]Data set: CATH20 (redundancy reduced at PIDE≤20); performance measures: QrawTop1 (Eq. 1) reflected the percentage of queries for which the first hit was correct (same CATH, identifier); CATH, classes (columns): on the level of class (C), CATH (Orengo et al., 1997; Sillitoe et al., 2019) distinguishes between mostly-alpha, mostly-beta, mixed alpha/beta and "few secondary structure"; values (rows): number of queries and targets, and two different ways to compile accuracy, the first (QrawTop1 family) is the fraction of queries where the top hit is from the same CATH, family, the second (QrawTop1 class) does the same but considers one level higher in the CATH, hierarchy.

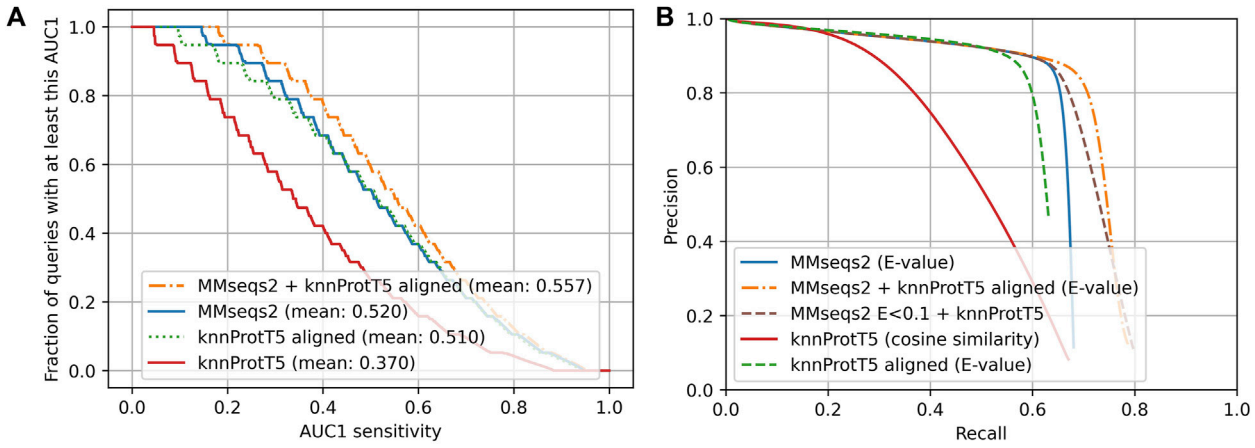

FIGURE 2
Sequence-based approaches better for full-length proteins (Pfam20). Data set: Pfam20; we searched all-against-all through 313,518 proteins with Pfam (Bateman et al., 2000) annotations. **(A)** Cumulative distribution of AUC1 that for each query reflected the fraction of queries (against the 313,518) with a matching Pfam annotation ranked above the first hit without matching Pfam annotation. Higher curves implied higher sensitivity. The steps in the lines originated from sampling 20 members for each Pfam family. MMseqs2 (Steinegger and Söding, 2017) as state-of-the-art in fast and sensitive sequence-based alignment in solid blue; the novel knnProtT5 method in solid red; knnProtT5 for search plus Smith-Waterman alignment performed by mmseqs align in dotted green; the combination of MMseqs2 and knnProtT5 in dot-dashed orange. **(B)** Precision-recall curve for the same methods as in panel A except for the dashed green marking an additional test combining unaligned knnProtT5 hits with MMseqs2 to show the power of embeddings on their own. The two combined approaches use MMSeqs2 E-values as base and knnProtT5 with cosine similarity only where MMseqs2 had low E-values. This explains the large overlap with MMseqs2.

single domain-like Pfam regions). Instead of outperforming MMseqs2 (for CATH20), knnProtT5 performed clearly worse for Pfam20 (Figure 2: solid red and dotted green below solid blue; AUC1(knn,Pfam20) = 0.367 vs. AUC1 (Mmseqs2,Pfam20) = 0.52). While knnProtT5 and MMSeqs2 found a similar fraction of homologs in the first 300 hits (68.0% vs. 67.4%), the vector distances (cosine similarity) between the per-protein representations did not sort the hits precisely enough, leading to a drops in AUC1 specifically and on the entire precision-recall more generally (Figure 3; Supplementary Figure S5).

Shuffling domains

By using Pfam annotations as ground truth, we might incorrectly consider a hit as incorrect when domain annotations are missing. The common solution is to shuffle the amino acid sequence outside of domain annotations and/or to add reversed or shuffled sequences as known incorrect hits (Brenner et al., 1998; Steinegger and Söding, 2017; Buchfink et al., 2021). However, we found that ProtT5 clearly separated between reversed or shuffled and real sequence (Supplementary Figure S7; Supplementary Table S2). Thus, we had to accept that some correct hits may be labeled as incorrect.

Combined method most sensitive

As for single-domains, the combined method (MMseqs2 + knnProtT5) considerably increased the overall sensitivity (Figure 2B: Combined method). Toward this end, we aligned the

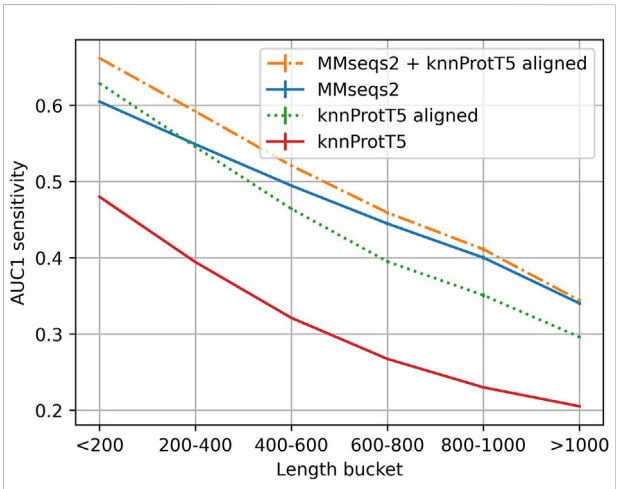

FIGURE 3
Longer proteins more difficult for Pfam20. Mean AUC1 sensitivity for different bins of lengths (number of residues) for the query protein (full proteins, not just domains compared to Pfam regions), showing how the combined method works across different sequence length. For those full-length proteins, all methods performed better for shorter than for longer proteins, e.g., MMseqs2 performance was almost half for proteins shorter than 200 residues than for those longer than 1,000 residues, and still substantially better for proteins with <600 residues that account for the majority of UniProt. Long proteins are often contain multiple domain which have more total homologs on average and are also more difficult to match on per-protein representation level (Discussion).

top 300 knnProtT5 hits using MMseqs2 and merged those 300 with the top hits from MMseqs2 ranked by E-value. This combination raised the AUC1 from 0.52 to 0.557 (Figure 3) and improved the recall even for high precision (Figure 3). The number of hits was the main hyperparameter for knnProtT5 and the subsequent alignment. We chose 300 to be identical to the MMseqs2 prefilter default. Although for the embeddings of some pLMs the value mattered more, essentially all embeddings appeared relatively stable for choices above 150 hits (Supplementary Figure S6). In particular, the combined method clearly outperformed MMseqs2 choosing the top 600 rather than 300 hits (AUC1(300) = 0.520 vs. AUC1(600) = 0.523). Thus, more hits correlated with increased AUC1 at the expense of runtime (below).

We also combined MMseqs2 and knnProtT5 without aligning the k-nn hits. Toward this end, we first filled the result list for a query with the MMseqs2 hits at E-values<0.1 and then appended the k-nn hits, i.e., we accepted very reliable MMseqs2 hits, and added knnProtT5 hits to fill up the hitlist to 300. This simpler scheme also reached an AUC1 of 0.558, but the recall only improved for lower precision (Figure 2B). The results were very similar to those for the combined method with re-alignment, reaching the same AUC1 of 59%.

Aligning knnProtT5 hits competitive

Aligning the knnProtT5 hits with Smith-Waterman (Smith and Waterman, 1981), yielded an AUC1 similar to that of MMseqs2. Using knnProtT5 as prefilter instead of MMseqs2 is, however, infeasible in practice due to the immense amount of time needed to compute per-protein embeddings. Although we aligned with an E-value cutoff of 10,000, the mean recall over all up to 300 hits dropped from 67.4% to 63.8%, or put differently: 3.6% of all homologs were correctly found by knnProtT5 but then dropped because they could not be aligned adequately.

Lower AUC1 for longer proteins

While knnProtT5 correctly retrieved many hits for long proteins, barely any of those were not already found by MMseqs2 (Figure 3). Splitting long proteins into overlapping slices of 600 residues, which were embedded individually and searched all-against-all against a databases of slices, did not improve (results not shown). Due to the quadratic growth in terms of costs (time and memory) for computing ProtT5 embeddings for longer proteins, we had to remove 0.6% of the proteins with over than 3,096 residues, worsening the results for the >1,024 bucket slightly (Figure 3). We observed that long sequences had disproportionally many hits with high cosine similarity (>0.95) and no matching annotations, most likely due to missing annotations.

Runtime: Method fast, but slowed down by embedding-lookup

MMseqs2 took 17 min 39 s for the search (12 m 2 s prefilter and 5 m 37 s align). Generating embeddings took 7 h 23 min, giving an average of 0.08 s per protein. Generating a Hierarchical Navigable Small World Graph (HNSW (Malkov and Yashunin, 2018) took 15 s, the search took 77 s. Compared to exhaustive nearest neighbor search we lost 0.004 AUC1 sensitivity (0.367–0.371), while the effect was below standard error for aligned knnProtT5 and the combined method.

Discussion

Key step: Comparing generalized sequences

Embeddings from protein Language Models (pLMs) appear to carry information about aspects such as protein function, structure, and evolution (Rao et al., 2019; Littmann et al., 2020; Littmann et al., 2021a; Littmann et al., 2021b; Elnaggar et al., 2021; Marquet et al., 2021; Rives et al., 2021; Dunham et al., 2022; Heinzinger et al., 2022; Weissenow et al., 2022). In this sense, they constitute what we might refer to as "generalized sequences." The key advance underlying our novel approach is to use generalized sequences for remote homology detection. While the idea for this is not new (Alley et al., 2019; Littmann et al., 2021a; Littmann et al., 2021b; Ofer et al., 2021; Bileschi et al., 2022; Heinzinger et al., 2022; Nallapareddy et al., 2022), here we presented a more rigorous and generic framework for directly comparing embedding-based to sequence-based alignments which have been optimized which have been optimized for half a century. Despite this advantage of being decades ahead in experience, methods that train on embeddings to map proteins to particular databases, such as CATH (Orengo et al., 1997; Sillitoe et al., 2019) and Pfam (Bateman et al., 2000) already outperform traditional sequence-based methods, even those based on profiles (Bileschi et al., 2022; Heinzinger et al., 2022; Nallapareddy et al., 2022). Here, we explored to which extent pLM embeddings directly, i.e., without any further training, are competitive in terms of performance and speed to a state-of-the-art (SOTA) sequence-based identification of homologs.

Our approach of finding k-nn embedding matches appeared to have several advantages. Firstly, by using k-nn matches (dubbed knnProtT5), we could explicitly drop the incorrect and limiting assumption that alignments at position P1 are statistically independent of those at position P2. Secondly, by replacing an amino acid alphabet with a vector condensing information from other residues, possibly far away in terms of sequence separation, that influence the evolution, function and structure at each residue position P, we implicitly use such constraints to compare sequences. Thus, although other novel solutions for non-iterated homology searches based on embeddings tend to focus on speed, our solution tried to combine speed and sensitivity. This was most prominently

exemplified by the 3.6% of ground truth homologs which knnProtT5 found but which could not be reasonably aligned [neither by MMseqs2 (Steinegger and Söding, 2017) nor by Smith-Waterman (Smith and Waterman, 1981)].

Overall, raw embeddings through k-nn (knnProtT5) could improve over traditional sequence similarity searched, both for single-domain vs. single-domain homology-based inference (CATH20, Figure 1) and for more general full-length vs. single-domain/Pfam-region homology searches (Pfam20, Figure 2). Merging sequence- and embedding-based (MMseqs2 + knnProtT5) did better than any of the two throughout (Figures 1, 2: combined method). However, for the more realistic use-case of comparing full-length proteins against Pfam-regions, MMseqs2 overall clearly outperformed the raw embeddings (Figure 2). Thus, we established an idea for a simple solution of using embeddings without further machine learning that showed some promises and strengths without breaking through.

Speed not necessarily sufficient

While the nearest neighbor search itself is blazingly fast, the time needed to generate the per-protein embeddings by ProtT5 and the effective GPU requirement might throw up major hurdles for adopting our approach. In fact for full-length proteins, the iterated profile search from MMseqs2 currently yields better results in shorter time. Two future eventualities might change this: firstly, databases such as UniProt (UniProt Consortium, 2017) might offer per-protein embeddings for all their proteins. If so, knnProtT5 would immediately become competitive. Secondly, judging from advances in NLP, we expect significant model speedups, potentially making k-nn with alignment viable on its own. PLMs leaped through rounds of exponential improvements over the last 3 years: from embedding-based prediction methods being faster than multiple sequence alignment (MSA) based predictions but much worse to outperforming MSA-based methods. Given this rapid evolution, large pLMs may soon become sufficiently good and/or small (NLP on smart phone) to justify the runtime cost. For instance, only the more recent ESM (Rives et al., 2021) and ProtT5 (Elnaggar et al., 2021) outperformed MMseqs2 for CATH20, while slightly earlier models such as SeqVec, or ProtBERT failed to do so. Maybe the next leap for the next pLM by increasing power and reducing the size will shift the balance more toward embedding-than sequence-based solutions.

Multiple domains will continue to challenge comparisons of entire proteins

As many as 80% of all proteins may consist of several domains. If we chopped all proteins into their domains and compared all-domains-against-all, the then full-domain embedding based k-nn succeeded (Figure 1), while for the comparison of full-length proteins against domains, the average-pooled per-protein embeddings of the queries are too coarse-grained (Figure 2). Sequence-based solutions built upon the local Smith-Waterman concept (Smith and Waterman, 1981) still succeed because of matching subsequences. In fact, this most likely explained the lack of improvement for proteins longer than 1,000 residues (Figure 3). Another possible path might be to directly create pLMs capturing entire domains as the "units," however, so far there seems no solution in this direction that succeeded without having to retrain specific AI models on top of embeddings, such as CATHe (Vaswani et al., 2017) or ProtTucker (Heinzinger et al., 2022), or the Pfam-AI (Bileschi et al., 2022).

While only tangentially relevant for homology search, we considered ProtT5's ability to detect "fake" sequences and make them trivially separable a noticeable result in its own right.

No advance without alignment?

A fundamental strength and limitation of our approach is that k-nn hits need to be aligned, e.g., by Smith-Waterman (Smith and Waterman, 1981). Alignments become unreliable at low levels of sequence identity, i.e., exactly in the realm for which embedding similarity promises to be useful (Rost, 1999; Steinegger and Söding, 2017; Littmann et al., 2020; Buchfink et al., 2021). Indeed, knnProtT5 found many correct hits below 20% PIDE that could not be aligned correctly (Supplementary Figure S6: knnProtT5 vs. knnProtT5_aligned). These results might suggest the need for the development of an embedding-based local alignment method to use the full potential of embedding based homology search and to make hits interpretable beyond a single score. A few approaches have been proposed toward this end, however, these are limited to global alignments (Bepler and Berger, 2019; Morton et al., 2020), i.e., will even worsen the decline from domain vs. domain (CATH20, Figure 1) tofull-length vs. domain (Pfam20, Figure 2).

K-nn index for fast pre-filtering?

An exhaustive k-nn index search has quadratic complexity, making it unworkable for large datasets, compared to modern indices with log-linear runtime complexity (n·logn). The HNSW index we chose operated considerably faster than the MMseqs2 prefilter while finding a comparable number of correct hits. It scaled well to billions of vectors (Malkov and Yashunin, 2018), making it feasible to search even metagenomics databases such as BFD (Steinegger and Söding, 2018; Steinegger et al., 2019). Combined with a fast pLM, this has the potential to outperform and outscale k-mer based approaches.

Conclusion

We demonstrated nearest neighbor search with protein Language Model (pLM) embeddings (here knnProtT5) to constitute a valuable complement to MMseqs2, enabling more sensitive homology searches. The embedding-based solution exceled in detecting distantly related domains (remote homology detection), finding hits that previously were not accessible by non-iterated homology search. Thus, embedding-based solutions might offer a more stringent baseline for the reach of homology based inference (if SIM(Q,A) > threshold, copy annotation from A to Q). knnProtT5 also scales to increasingly large databases. The only limitation at the moment, the generation of per-protein embeddings, might be removed as an obstacle through database resources such as UniProt providing such embeddings in prestored fashion for all proteins. Either way, rapid progress of pLMs already renders nearest neighbor searches on embeddings a promising new path, allowing us to tap into a new pool of homologs from embedding space and to go beyond sequence similarity.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Materials, further inquiries can be directed to the corresponding author.

Author contributions

KS, MH, MS, and BR conceived and planned the experiments. KS carried out the experiments. KS wrote the manuscript with MH and BR. MS contributes MMseqs2 benchmarking.

Funding

This work was supported by the BMBF through the program "Software Campus 2.0 (TU München)," project number 01IS17049 and by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) Grant RO 1320/4-1. Martin Steinegger acknowledges support from the National Research Foundation of Korea grant [2019R1A6A1A10073437, 2020M3A9G7103933, and 2021R1C1C102065]; New Faculty Startup Fund and the Creative-Pioneering Researchers Program through Seoul National University.

Acknowledgments

Thanks to Tim Karl (TUM) for help with hardware and software; to Inga Weise (TUM) for support with many other aspects of this work. Christian Dallago (TUM) leading bio_embeddings and feedback throughout this work. Last, not least, thanks to all those who deposit their experimental data in public databases, and to those who maintain these databases.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fbinf.2022.1033775/full#supplementary-material

References

Alley, E. C., Khimulya, G., Biswas, S., Alquraishi, M., and Church, G. M. (2019). Unified rational protein engineering with sequence-based deep representation learning. *Nat. Methods* 16, 1315–1322. doi:10.1038/s41592-019-0598-1

Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., Miller, W., et al. (1997). Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic acids Res.* 25, 3389–3402. doi:10.1093/nar/25.17.3389

Babenko, A., and Lempitsky, V. (2014). The inverted multi-index. *IEEE Trans. Pattern Anal. Mach. Intell.* 37, 1247–1260. doi:10.1109/tpami.2014.2361319

Baranchuk, D., Babenko, A., and Malkov, Y. (2018) Revisiting the inverted indices for billion-scale approximate nearest neighbors"), 202–216.

Bateman, A., Birney, E., Durbin, R., Eddy, S. R., Howe, K. L., and Sonnhammer, E. L. (2000). The Pfam protein families database. *Nucleic Acids Res.* 28, 263–266. doi:10.1093/nar/28.1.263

Bepler, T., and Berger, B. (2019). Learning protein sequence embeddings using information from structure. arXiv preprint arXiv:1902.08661.

Bernhardsson, E. (2020). Annoy: Approximate nearest neighbors in C++/Python optimized for memory usage and loading/saving to disk. GitHub. Available at: https://github.com/spotify/annoy.

Bileschi, M. L., Belanger, D., Bryant, D. H., Sanderson, T., Carter, B., Sculley, D., et al. (2022). Using deep learning to annotate the protein universe. *Nat. Biotechnol.* 40, 932–937. doi:10.1038/s41587-021-01179-w

Brenner, S. E., Chothia, C., and Hubbard, T. J. (1998). Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships. *Proc. Natl. Acad. Sci. U. S. A.* 95, 6073–6078. doi:10.1073/pnas.95.11.6073

Buchfink, B., Reuter, K., and Drost, H.-G. (2021). Sensitive protein alignments at tree-of-life scale using DIAMOND. *Nat. Methods* 18, 366–368. doi:10.1038/s41592-021-01101-x

Burley, S. K., Berman, H. M., Kleywegt, G. J., Markley, J. L., Nakamura, H., and Velankar, S. (2017). Protein Data Bank (PDB): The single global macromolecular structure archive. *Methods Mol. Biol.* 1607, 627–641. doi:10.1007/978-1-4939-7000-1_26

Dallago, C., Schütze, K., Heinzinger, M., Olenyi, T., Littmann, M., Lu, A. X., et al. (2021). Learned embeddings from deep learning to visualize and predict protein sets. *Curr. Protoc.* 1, e113. doi:10.1002/cpz1.113

Darwin, C. (1859). *The origin of species by means of natural selection, or the preservation of favoured races in the struggle for life*. London: John Murray.

Doolittle, R. F. (1986). *Of URFs and ORFs: A primer on how to analyze derived amino acid sequences*. Mill Valley, CA: University Science Books.

Dunham, A. S., Beltrao, P., and Alquraishi, M. (2022). High-throughput deep learning variant effect prediction with Sequence UNET. bioRxiv.

El-Gebali, S., Mistry, J., Bateman, A., Eddy, S. R., Luciani, A., Potter, S. C., et al. (2019). The Pfam protein families database in 2019. *Nucleic acids Res.* 47, D427–D432. doi:10.1093/nar/gky995

Elnaggar, A., Heinzinger, M., Dallago, C., Rehawi, G., Yu, W., Jones, L., et al. (2021). ProtTrans: Towards cracking the language of lifes code through self-supervised deep learning and high performance computing. IEEE Trans Pattern Anal Mach Intell PP.

Elnaggar, A., Heinzinger, M., Dallago, C., Rihawi, G., Wang, Y., Jones, L., et al. 2020. ProtTrans: Towards cracking the language of life's code through self-supervised deep learning and high performance computing. bioRxiv [Online].

Finn, R. D., Clements, J., and Eddy, S. R. (2011). HMMER web server: Interactive sequence similarity searching. *Nucleic Acids Res.* 39, W29–W37. doi:10.1093/nar/gkr367

Friedberg, I., Kaplan, T., and Margalit, H. (2000). Glimmers in the midnight zone: Characterization of aligned identical residues in sequence-dissimilar proteins sharing a common fold. *Proc. Int. Conf. Intell. Syst. Mol. Biol.* 8, 162–170.

Haussler, D., Krogh, A., Mian, I. S., and Sjölander, K. (1993). "Protein modeling using hidden Markov models: Analysis of globins,". Editor L. Hunter (Los Alamitos, CA: IEEE Computer Society Press), 792–802.Proceedings for the 26th Hawaii International Conference on Systems Sciences

Heinzinger, M., Elnaggar, A., Wang, Y., Dallago, C., Nechaev, D., Matthes, F., et al. (2019). Modeling aspects of the language of life through transfer-learning protein sequences. *BMC Bioinforma.* 20, 723. doi:10.1186/s12859-019-3220-8

Heinzinger, M., Littmann, M., Silllitoe, I., Bordin, N., Orengo, C., and Rost, B. (2022). Contrastive learning on protein embeddings enlightens midnight zone. *Nar. Genom. Bioinform.* 4, lqac043. doi:10.1093/nargab/lqac043

Higgins, D. G., and Sharp, P. M. (1989). Fast and sensitive multiple sequence alignments on a microcomputer. *Bioinformatics* 5, 151–153. doi:10.1093/bioinformatics/5.2.151

Jegou, H., Douze, M., and Schmid, C. (2010). Product quantization for nearest neighbor search. *IEEE Trans. Pattern Anal. Mach. Intell.* 33, 117–128. doi:10.1109/tpami.2010.57

Johnson, J., Douze, M., and Jégou, H. (2019). Billion-scale similarity search with GPUs. *IEEE Trans. Big Data*.

Johnson, L. S., Eddy, S. R., and Portugaly, E. (2010). Hidden Markov model speed heuristic and iterative HMM search procedure. *BMC Bioinforma.* 11, 431. doi:10.1186/1471-2105-11-431

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., et al. (2021). Highly accurate protein structure prediction with AlphaFold. *Nature* 596, 583–589. doi:10.1038/s41586-021-03819-2

Karlin, S., and Altschul, S. F. (1990). Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proc. Natl. Acad. Sci. U. S. A.* 87, 2264–2268. doi:10.1073/pnas.87.6.2264

Khandelwal, U., Levy, O., Jurafsky, D., Zettlemoyer, L., and Lewis, M. (2019). Generalization through memorization: Nearest neighbor language models. arXiv preprint arXiv:1911.00172.

Kolodny, R., Koehl, P., and Levitt, M. (2005). Comprehensive evaluation of protein structure alignment methods: Scoring by geometric measures. *J. Mol. Biol.* 346, 1173–1188. doi:10.1016/j.jmb.2004.12.032

Krogh, A., Brown, M., Mian, I. S., Sjölander, K., and Haussler, D. (1994). Hidden Markov models in computational biology: Applications to protein modeling. *J. Mol. Biol.* 235, 1501–1531. doi:10.1006/jmbi.1994.1104

Li, J., Liu, H., Gui, C., Chen, J., Ni, Z., Wang, N., et al. (2019). "The design and implementation of a real time visual search system on JD E-commerce platform"), 9–16.

Littmann, M., Bordin, N., Heinzinger, M., Schütze, K., Dallago, C., Orengo, C., et al. (2021a). Clustering FunFams using sequence embeddings improves EC purity. *Bioinformatics* 37, 3449–3455. doi:10.1093/bioinformatics/btab371

Littmann, M., Heinzinger, M., Dallago, C., Olenyi, T., and Rost, B. (2020). Embeddings from deep learning transfer GO annotations beyond homology. bioRxiv.

Littmann, M., Heinzinger, M., Dallago, C., Olenyi, T., and Rost, B. (2021b). Embeddings from deep learning transfer GO annotations beyond homology. *Sci. Rep.* 11, 1160. doi:10.1038/s41598-020-80786-0

Littmann, M., Heinzinger, M., Dallago, C., Weissenow, K., and Rost, B. (2021c). Protein embeddings and deep learning predict binding residues for various ligand classes. *Sci. Rep.* 11, 23916. doi:10.1038/s41598-021-03431-4

Liu, J., Hegyi, H., Acton, T. B., Montelione, G. T., and Rost, B. (2004). Automatic target selection for structural genomics on eukaryotes. *Proteins.* 56, 188–200. doi:10.1002/prot.20012

Liu, J., and Rost, B. (2003). Domains, motifs, and clusters in the protein universe. *Curr. Opin. Chem. Biol.* 7, 5–11. doi:10.1016/s1367-5931(02)00003-0

Liu, T., Rosenberg, C., and Rowley, H. A. (2007). *Clustering billions of images with large scale nearest neighbor search*, 28.()*IEEE)*

Lu, A. X., Zhang, H., Ghassemi, M., and Moses, A. (2020). Self-supervised contrastive learning of protein representations by mutual information maximization. bioRxiv.

Malkov, Y. A., and Yashunin, D. A. (2018). "Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs," in IEEE transactions on pattern analysis and machine intelligence.

Marquet, C., Heinzinger, M., Olenyi, T., Dallago, C., Erckert, K., Bernhofer, M., et al. (2021). Embeddings from protein language models predict conservation and variant effects. *Hum. Genet.* 141, 1629–1647. doi:10.1007/s00439-021-02411-y

Marx, V. (2022). Method of the year: Protein structure prediction. *Nat. Methods* 19, 5–10. doi:10.1038/s41592-021-01359-1

Matsui, Y., Uchida, Y., Jégou, H., and Satoh, S. I. (2018). [Invited paper] A survey of product quantization. *ITE Trans. Media Technol. Appl.* 6, 2–10. doi:10.3169/mta.6.2

Mirdita, M., Schütze, K., Moriwaki, Y., Heo, L., Ovchinnikov, S., and Steinegger, M. (2021). ColabFold-Making protein folding accessible to all.

Morton, J., Strauss, C., Blackwell, R., Berenberg, D., Gligorijevic, V., and Bonneau, R. (2020). Protein structural alignments from sequence. bioRxiv.

Nallapareddy, V., Bordin, N., Sillitoe, I., Heinzinger, M., Littmann, M., Waman, V. P., et al. (2022). CATHe: Detection of remote homologues for CATH superfamilies using embeddings from protein language models. bioRxiv.

Nepomnyachiy, S., Ben-Tal, N., and Kolodny, R. (2017). Complex evolutionary footprints revealed in an analysis of reused protein segments of diverse lengths. *Proc. Natl. Acad. Sci. U. S. A.* 114, 11703–11708. doi:10.1073/pnas.1707642114

Ofer, D., Brandes, N., and Linial, M. (2021). the language of proteins: NLP, machine learning & protein sequences. *Comput. Struct. Biotechnol. J.* 19, 1750–1758. doi:10.1016/j.csbj.2021.03.022

Orengo, C. A., Brown, N. P., and Taylor, W. T. (1992). Fast structure alignment for protein databank searching. *Proteins.* 14, 139–167. doi:10.1002/prot.340140203

Orengo, C. A., Michie, A. D., Jones, S., Jones, D. T., Swindells, M. B., and Thornton, J. M. (1997). Cath - a hierarchic classification of protein domain structures. *Structure* 5, 1093–1109. doi:10.1016/s0969-2126(97)00260-8

Orengo, C. A., Sillitoe, I., Reeves, G., and Pearl, F. M. (2001). Review: What can structural classifications reveal about protein evolution? *J. Struct. Biol.* 134, 145–165. doi:10.1006/jsbi.2001.4398

Owen, R. (1848). *On the archetype and homologies of the vertebrate skeleton*. London: Richard and John E. Taylor.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., et al. (2018). Deep contextualized word representations. arXiv preprint arXiv:1802.05365.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., et al. (2019). Exploring the limits of transfer learning with a unified text-to-text transformer. arXiv preprint arXiv:1910.10683.

Rao, R., Bhattacharya, N., Thomas, N., Duan, Y., Chen, P., Canny, J., et al. (2019), Evaluating protein transfer learning with TAPE"), 9689–9701.

Remmert, M., Biegert, A., Hauser, A., and Soding, J. (2012). HHblits: Lightning-fast iterative protein sequence searching by HMM-HMM alignment. *Nat. Methods* 9, 173–175. doi:10.1038/nmeth.1818

Rives, A., Meier, J., Sercu, T., Goyal, S., Lin, Z., Liu, J., et al. (2021). Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc. Natl. Acad. Sci. U. S. A.* 118, e2016239118. doi:10.1073/pnas.2016239118

Rost, B. (2002). Enzyme function less conserved than anticipated. *J. Mol. Biol.* 318, 595–608. doi:10.1016/s0022-2836(02)00016-5

Rost, B. (1997). Protein structures sustain evolutionary drift. *Fold. Des.* 2, S19–S24. doi:10.1016/s1359-0278(97)00059-x

Rost, B. (1999). Twilight zone of protein sequence alignments. *Protein Eng. Des. Sel.* 12, 85–94. doi:10.1093/protein/12.2.85

Sillitoe, I., Dawson, N., Lewis, T. E., Das, S., Lees, J. G., Ashford, P., et al. (2019). Cath: Expanding the horizons of structure-based functional annotations for genome sequences. *Nucleic acids Res.* 47, D280–D284. doi:10.1093/nar/gky1097

Slaney, M., and Casey, M. (2008). Locality-sensitive hashing for finding nearest neighbors [lecture notes]. *IEEE Signal Process. Mag.* 25, 128–131. doi:10.1109/msp.2007.914237

Smith, T. F., and Waterman, M. S. (1981). Identification of common molecular subsequences. *J. Mol. Biol.* 147, 195–197. doi:10.1016/0022-2836(81)90087-5

Sonnhammer, E. L., Eddy, S. R., and Durbin, R. (1997). Pfam: A comprehensive database of protein domain families based on seed alignments. *Proteins.* 28, 405–420. doi:10.1002/(sici)1097-0134(199707)28:3<405::aid-prot10>3.0.co;2-l

Staerk, H., Dallago, C., Heinzinger, M., and Rost, B. (2021). Light attention predicts protein location from the language of life. bioRxiv.

Steinegger, M., Mirdita, M., and Söding, J. (2019). Protein-level assembly increases protein sequence recovery from metagenomic samples manyfold. *Nat. Methods* 16, 603–606. doi:10.1038/s41592-019-0437-4

Steinegger, M., and Söding, J. (2018). Clustering huge protein sequence sets in linear time. *Nat. Commun.* 9, 2542–2548. doi:10.1038/s41467-018-04964-5

Steinegger, M., and Söding, J. (2017). MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat. Biotechnol.* 35, 1026–1028. doi:10.1038/nbt.3988

Suzek, B. E., Wang, Y., Huang, H., Mcgarvey, P. B., Wu, C. H., and Consortium, U. (2015). UniRef clusters: A comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics* 31, 926–932. doi:10.1093/bioinformatics/btu739

Teufel, F., Armenteros, J. J. A., Johansen, A. R., Gislason, M. H., Pihl, S. I., Tsirigos, K. D., et al. (2021). SignalP 6.0 achieves signal peptide prediction across all types using protein language models. bioRxiv.

Tunyasuvunakool, K., Adler, J., Wu, Z., Green, T., Zielinski, M., Zidek, A., et al. (2021). Highly accurate protein structure prediction for the human proteome. *Nature* 596, 590–596. doi:10.1038/s41586-021-03828-1

Uniprot Consortium (2017). UniProt: The universal protein knowledgebase. *Nucleic Acids Res.* 45, D158–D169. doi:10.1093/nar/gkw1099

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017), Attention is all you need"), 5998–6008.

Weissenow, K., Heinzinger, M., and Rost, B. (2022). Protein language-model embeddings for fast, accurate, and alignment-free protein structure prediction. *Structure* 30, 1169–1177.e4. 2021.2007.2031. doi:10.1016/j.str.2022.05.001

Yona, G., and Levitt, M. (2002). Within the twilight zone: A sensitive profile-profile comparison tool based on information theory. *J. Mol. Biol.* 315, 1257–1275. doi:10.1006/jmbi.2001.5293