Biotechnology Advances 91 (2026) 108945

Contents lists available at ScienceDirect

Biotechnology Advances

journal homepage: www.elsevier.com/locate/biotechadv

Interpreting embeddings from genome and protein language models

Luke E. May [1], Jacob B. White [1], Garrett W. Roell [*]

Department of Molecular Biosciences and Bioengineering, University of Hawai'i at Mānoa, Honolulu, HI 96822, United States

ARTICLE INFO

Keywords:
Genome language models
Protein language models
Embeddings
Representation learning

ABSTRACT

Following advances in the field of natural language processing related to the transformer and the attention mechanism, many large language models have been trained using DNA and protein databases. While these models are designed for next-token prediction with the goal of generating novel sequences, researchers have found that the intermediate representations, or embeddings, of the DNA and amino acid sequences as they pass through these models can be mined for biological insights. Recently, more emphasis has been placed on studying the models' embeddings than model outputs. Genome language model embeddings have been used to predict the severity of single nucleotide variants, intron/exon boundaries, and anti-CRISPR activity. Protein language model embeddings have been used for predicting remote homology, protein structure, and protein function. These findings demonstrate that model embeddings provide a high-resolution, alignment-free framework for understanding the genome and proteome, and offer a transformative approach to enzyme engineering and functional genomics.

1. What is a language model?

In 1950, Alan Turing hypothesized that future computers would be able to mimic the behavior of the human brain (Turing, 1950). Since then, researchers have been working on machine learning systems that are capable of complex behavior. Initial efforts involved trying to write explicit logical rules (McCulloch and Pitts, 1943), but current methods are fully data-driven. These systems use backpropagation and gradient descent to update large numbers of weights and biases to minimize the errors in their outputs (LeCun et al., 2012; Rumelhart et al., 1986). When these principles are scaled to large neural networks with many hidden layers, emergent behavior including computer vision, long-term dependency in speech recognition (Tang and Glass, 2018), and generative image modeling (Brixi et al., 2026) has been achieved.

Simultaneously, research in natural language processing (NLP) focused on enabling computers to process and generate human language. Some technologies developed for this task include recurrent neural networks (RNNs) and long short-term memory (LSTM) networks, which processed text token-by-token. While effective, these models struggle with long-range dependencies and often fail to retain information from earlier tokens when processing later parts of a sequence (Hochreiter and Schmidhuber, 1997; Rumelhart et al., 1986). However, a major breakthrough occurred in 2017 with the emergence of the transformer architecture (Vaswani et al., 2017). The transformer replaced sequential processing with a "self-attention" mechanism. Rather than detecting features one step at a time, the model looks at every element in a sequence simultaneously and calculates an "attention score" between every pair of elements, allowing for larger context windows and the parallelization of model training. Suddenly, the cost to train large models decreased dramatically and allowed for much larger training datasets to be used.

The transformer architecture has had many successes with learning human languages, so researchers applied it to biological large language models (LLMs). Many genome language models for DNA sequences and protein language models for amino acid sequences have been published (Table 1) (Brixi et al., 2026; Dalla-Torre et al., 2025; Ji et al., 2021; Nguyen et al., 2023; Rives et al., 2021; Zhou et al., 2023). Table 1 provides a structural comparison of these models, detailing technical specifications such as hidden size, total layers, and context windows.

Language models for biological applications need to account for the specific constraints of the biological language they are working with. Unlike human text, DNA is bidirectional; therefore, genome language models need to be designed to process both the forward and reverse strands as equivalent inputs (Brixi et al., 2026; Nguyen et al., 2024). Furthermore, regulatory elements encoded in DNA can be up to 2 million base pairs away from the transcription start site (van Heyningen

* Corresponding author.
E-mail address: groell@hawaii.edu (G.W. Roell).
[1] These authors contributed equally to this work.

https://doi.org/10.1016/j.biotechadv.2026.108945
Received 30 March 2026; Received in revised form 4 June 2026; Accepted 4 June 2026
Available online 6 June 2026

Table 1
Architectural and training characteristics of representative state-of-the-art biological language models, highlighting domain specialization, tokenization strategy, model scale, and data composition. Domain indicates the biological sequence type modeled (e.g., DNA). Model class refers to the design of the model where 1st generation refers to sequence language models and second generation models refers to sequence-to-function models. Primary Goal is the intended use of the model. Token Level specifies the granularity of sequence representation (e.g., single nucleotide or k-mer tokenization). Architecture describes the underlying neural network framework. # of Parameters, Total Layers, and Model Dim. summarize model scale and representational capacity. Context Window denotes the maximum sequence length processed by the model, and Training Data identifies the primary dataset used for pretraining.

Ref	Model Name	Domain	Model Class & Objective	Primary Goal	Token Level	Architecture	# of Parameters	Model Scale (Layers / Dim)	Context Window	Training Data
(Brixi et al., 2026)	Evo 2 40B	DNA	1st Gen: Autoregressive (Next-token)	Sequence generation and long-range dependency	Single nucleotide	StripedHyena 2	40B	50 / 8192	1 M	OpenGenome2
(Dalla-Torre et al., 2025)	NT-Multispecies-v2	DNA	1st Gen: Masked (Bidirectional)	Understanding local genomic context and regulatory elements	6 k-mers	Transformer	500 M	29 / 1024	12 k	Multispecies
(Zhou et al., 2023)	DNABERT-2	DNA	1st Gen: Masked (Bidirectional)	Extracting site-specific features and motif detection	Byte pair encoding k-mer	Transformer	117 M	12 / ~768	22 k+	Multispecies
(Avsec et al., 2026)	AlphaGenome	DNA	2nd Gen: Sequence-to-Function	Direct prediction of genomic and epigenetic functional tracks	Single nucleotide	U-Net-inspired backbone (Encoder-Transformer tower-decoder)	~450 M	Enc: 7 / 1536, Trans-Tower: 9 / 1536, Pairwise: 5 / 2048, Dec: 7	~1 M	Functional genomics data from human (hg38) and mouse (mm10) samples
(Outeiral and Deane, 2024)	CaLM	AA Codons	1st Gen: Masked (Bidirectional)	Capturing codon bias and translational dynamics	Single codon	Transformer	86 M	12 / 768	1024	Curated European Nucleotide Archive
(de Oliveira et al., 2025)	SUPERMAGO+	AA Embeddings	2nd Gen: Sequence-to-Function (Ensemble)	Direct prediction of protein function and Gene Ontology (GO) annotation	Ensemble of SUPERMAGO and DIAMOND (Aggregation tool, assigns term-specific weights)	Transformer	900 M	10 Parallel MLPs +1 Stacking +1 Weighting Net / 4096–8192	1024	Final 5 layers of ESM2 T36 and ProtT5
(Rao et al., 2021)	MSA Transformer	AA	1st Gen: Masked (MSA-based)	Extracting co-evolutionary signals and predicting residue contacts	Single amino acid residue	Transformer	100 M	12 / 768	1024	26 million MSAs
(Elnaggar et al., 2022)	ProtT5-XXL	AA	1st Gen: Span Corruption (Denoising)	High-capacity general representation for per-residue downstream tasks	Single amino acid residue	Enc-Dec (T5)	11B	24 / 4096	1024+	UniRef100 + BFD
(Brandes et al., 2022)	ProteinBERT	AA	Hybrid 1st/2nd Gen: Masked (MLM) + Functional (GO)	Joint learning of biological grammar and functional annotation	Single amino acid residue	Dual-track Transformer (Local/ Global)	16 M	6 / Local: 128, Global: 512	~10 k+	UniProtKB/UniRef90
(Rao et al., 2019)	TAPE-Transformer	AA	1st Gen: Masked (Bidirectional)	General biological feature extraction and standardized benchmarking	Single amino acid residue	Transformer	~38 M	12 / 512	N/A	Pfam database
(Weissenow et al., 2022)	EMBER2	AA	Structure Predictor (Contact Map Prediction)	Fast 3D contact map and distance prediction	Single amino acid residue	ResNet-like Convolutional Network	~6.5 M	360 / 64	1024+	ProteinNet12 reduced with MMseqs2. validation: SetValCASP12. Test: SetTst29
(Lin et al., 2022)	ESMFold	AA	1st Gen: Masked (ESM-2 base)	Atomic-level 3D structure prediction	Single amino acid residue	ESM-2	15B	48 / 1024/128	1024 Rotary position embed-dings	UniRef50
(Dejean et al., 2025)	LV-5B	AA	1st Gen: Masked (Sparse Attention)	Viral sequence representation and understanding virus-host interactions	Single amino acid residue	Transformer with protein interaction sparse attention	650 M	33 / 1280	48 K	Parameters initialized from ESM-2, Entire viral genomes from NCBI Virus database

and Bickmore, 2013). To account for this, genome language models need to be designed with very large context lengths. Understanding how these models compress such vast sequences is key to understanding their predictive power.

2. What is an embedding?

LLMs, including genome and protein language models, work by transforming an input sequence through a series of transformations that ends with a prediction of the next output (Fig. 1). The first step in this process is called *tokenization*, where the sequence is broken into atomic units (Mielke et al., 2021). This can be done at a single base (A, T, C, G) or amino acid residue level, or at a k-mer level, meaning that each token represents a few nucleotides. For example, Evo 2 and AlphaGenome use single base resolution, while the Nucleotide Transformer uses 6-mer chunks of DNA (Avsec et al., 2026; Brixi et al., 2026; Dalla-Torre et al., 2025). The next step, called *embedding*, converts each token in the sequence into a vector of the *model dimension*. The model dimension is the length of each internal representation in the latent space. After the initial embedding, the vector of the model dimension is passed through a series of blocks where it is modified at each step. These blocks take embedding vectors as inputs and return modified embedding vectors as outputs. The specific math inside of each block can vary. Depending on the model, a given block could contain the attention mechanism in transformer blocks, convolutions in convolutional blocks, or a combination in Hyena blocks (He et al., 2016; Poli et al., 2023). The output of one block is used as the input to the next block. Because each block has the effect of modifying the vector, each block acts as a correction or adjustment. For this reason, this series of intermediate vectors is commonly referred to as the *residual stream*. After the vector passes through the final block, it gets projected back into a vector of the vocabulary length where each element in the vector corresponds to the probability of that token being the next token in the sequence.

Embeddings are the vectors that are returned at the end of each block. By training a model to predict the next nucleotide or amino acid in a sequence, it naturally develops internal representations of evolution's structural rules and biological motifs (Brown et al., 2020). The embeddings from genome and protein language models have been shown to contain rich biological information.

The work on extracting human-understandable information from embeddings has been guided by findings from the field of *mechanistic interpretability* (MI), which is dedicated to decomposing the model's internal states into interpretable features. Here, *features* refer to human-understandable properties of inputs as the model represents them (Elhage et al., 2022).

2.1. How can biological language models be classified?

Biological language models can be broadly categorized into three classes. The first two are language-based architectures. First-generation sequence language models are primarily designed to generate biological sequences and produce interpretable embeddings. Second-generation sequence-to-function models extend this framework by directly predicting functional properties from input sequences. The third class comprises structure predictors, which differ fundamentally in both objective and output representation from sequence-based language models.

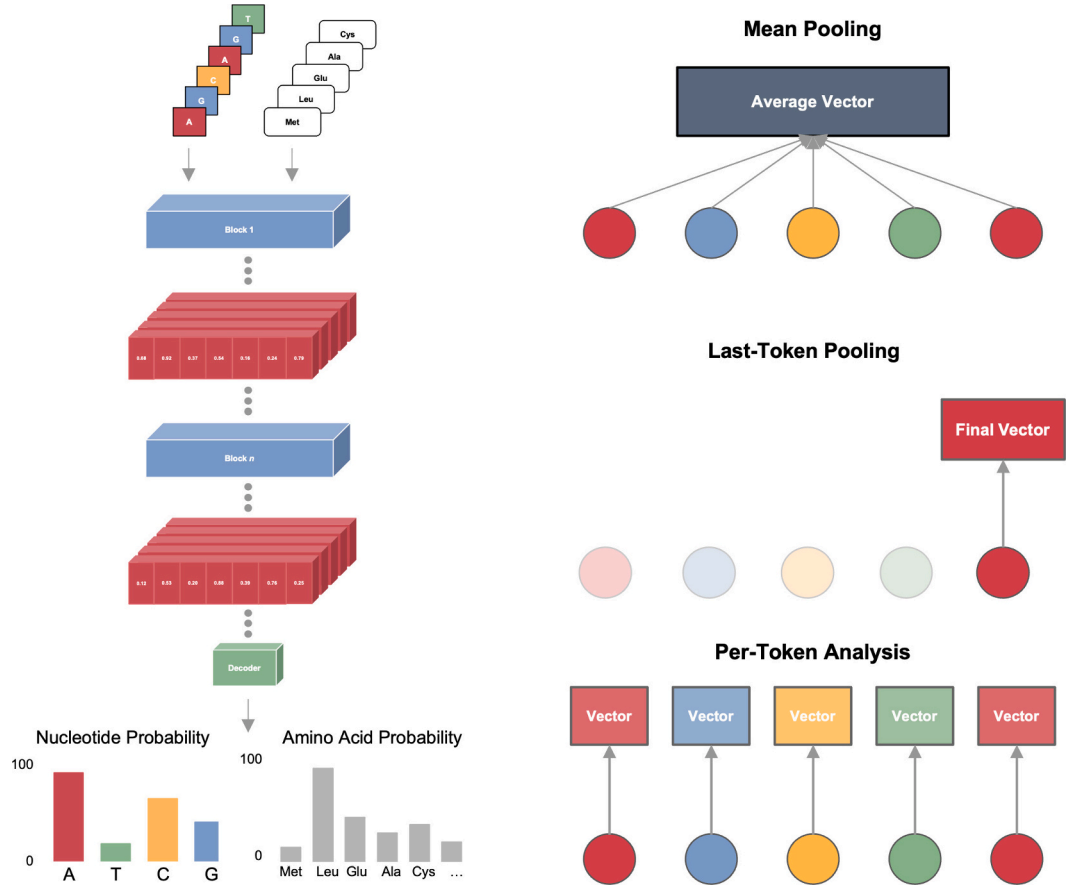

Fig. 1. Schematic overview of data flow in biological language models. The diagram highlights how token-level vectors are generated across model blocks, and how sequence-level embeddings are formed through pooling strategies (per-token analysis, mean pooling, or last-token pooling).

2.2. Sequence Language Models

Sequence language models are trained on classical language-modeling objectives: either predicting the next token in a sequence (autoregressive language modeling, e.g., Evo 2) or recovering hidden tokens within a sequence (masked language modeling, e.g., ESM-2).

These models can be used for two different tasks: sequence generation and embedding-based analysis. When the objective is to design novel proteins, genomes, or regulatory elements, the generated sequence itself is the primary output. In contrast, when the goal is to characterize, classify, or compare biological sequences, the model's embeddings are the more important analytical tool.

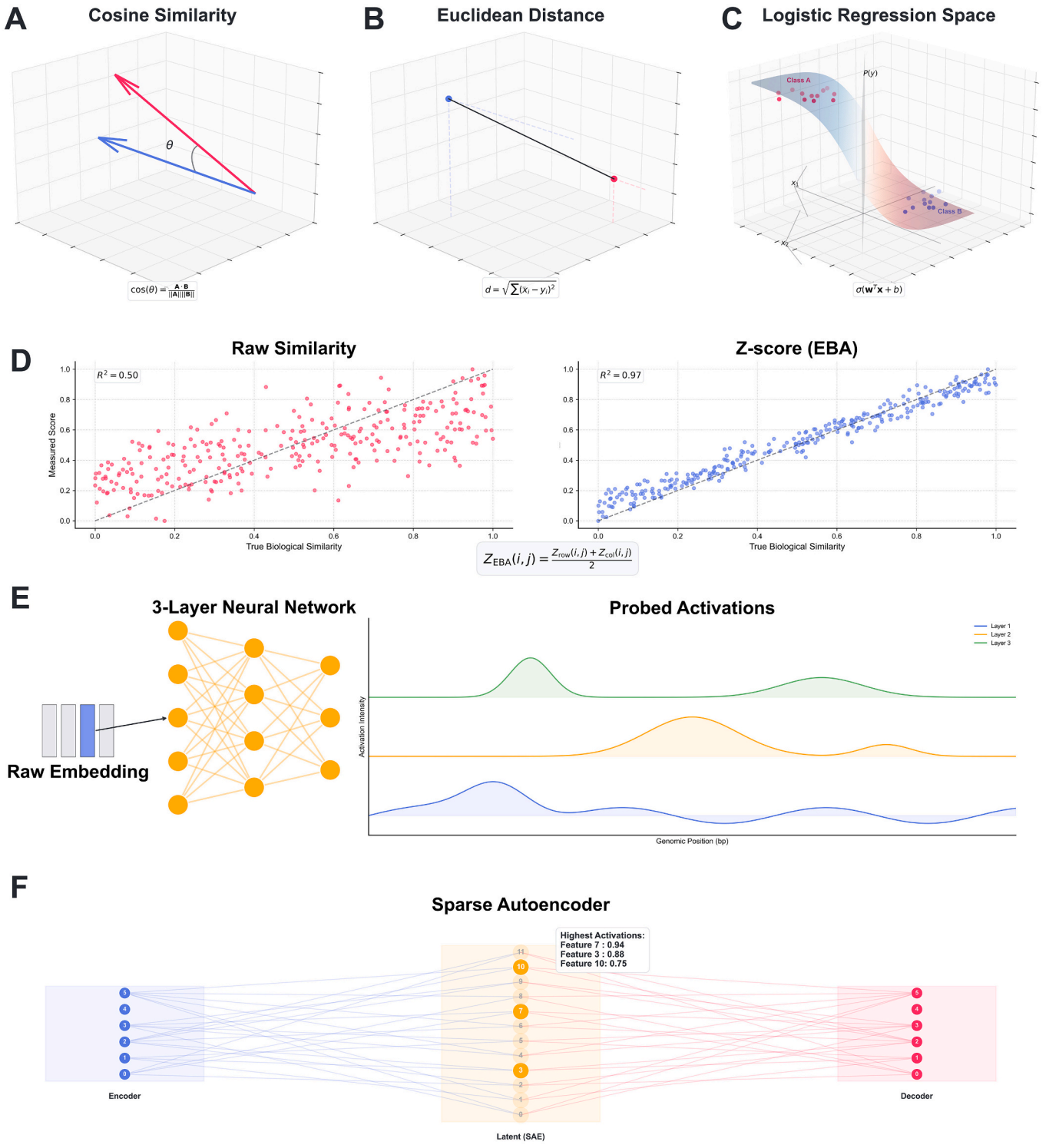

Fig. 2. Geometric and mechanistic interpretability of biological language model embeddings. (A) Cosine similarity: Angular relationship between two embedding vectors. (B) Euclidean distance: Spatial separation between embeddings in latent space. (C) Logistic regression in embedding space: Geometric interpretation of a classification boundary. (D) Embedding-based alignment (EBA) normalization: Comparison of raw similarity scores and *Z*-score–normalized signals. (E) Transformer probing: using embeddings from an intermediate model block for supervised downstream prediction. (F) Sparse autoencoder (SAE) feature decomposition: Expansion of dense embeddings into a higher-dimensional sparse representation.

2.3. Sequence-to-function models: the second generation

Second-generation models, such as AlphaGenome (Avsec et al., 2026), utilize transformer-based architectures but are optimized for a fundamentally different task: predicting functional properties. Rather than learning the probability of the next nucleotide, these models are directly trained to map DNA sequences to functional outputs, such as gene expression levels or chromatin accessibility. While they share architectural components with sequence language models, their primary goal is direct functional mapping rather than unsupervised sequence modeling or sequence generation.

2.4. Structure predictors

Finally, structure predicting models, like AlphaFold (Jumper et al., 2021), are grouped with biological language models because they incorporate attention-based mechanisms. However, these geometric deep-learning systems are designed to predict 3D spatial coordinates (structure) rather than sequence probability. While sequence language model embeddings often correlate with structural features, structure predictors are not trained with language-modeling objectives.

2.5. How have embeddings been processed to extract information?

Embeddings contain information about the sequence that was passed into them, and they can be used in their raw form or after being processed. Two sequences that have similar biological functions have been shown to have similar internal representations in the latent space of language models (Pantolini et al., 2024). It has been shown that embeddings derived from language models can capture evolutionary information comparable to, or in some cases exceeding, that obtained from multiple sequence alignments (Harrigan et al., 2024; Rao et al., 2021). When comparing the embeddings from two sequences, it is important to be intentional in the preprocessing decisions that are made as they can greatly influence downstream analyses and conclusions (Fig. 2).

LLMs generate embedding vectors for each token at each layer of the model. The pooling method used to extract a single embedding vector for a given sequence affects the downstream analysis. Typically, authors have used the embeddings from the final token to represent a whole sequence, as the last token's embedding incorporates context from all the preceding tokens (Elnaggar et al., 2022). An alternative approach is to use mean pooling, where the embeddings from all the tokens in a sequence are averaged to yield a single vector for the sequence (Elnaggar et al., 2022). In contrast, some analyses skip pooling entirely and only look at embeddings at a per-token level to preserve localized biological features.

Another important consideration when analyzing embeddings is choosing the block that contains the most biologically relevant information (Table 2). Biological meaning is rarely found in the earliest layers of a model. Instead, researchers often identify information-rich intermediate or late layers in a given model. Recent interpretability studies on models like ESM-2 and Evo 2 suggest that while early layers encode local structural motifs, high-level functional properties such as subcellular localization or Gene Ontology (GO) terms are most prominently represented in the middle-to-late layers (Lin et al., 2022; Rives et al., 2021). For example, in the 32-layer Evo 2 7B model, Layer 26 serves as a critical point where the model contains the highest concentration of features related to known biological structures (Brixi et al., 2026). Most often, researchers extract representations from the final layer (e.g. 100% depth) to serve as feature vectors for downstream tasks (Brandes et al., 2022; Lin et al., 2022; Rao et al., 2021; Rives et al., 2021; Villegas-Morcillo et al., 2022). In contrast, a study involving the Nucleotide Transformer demonstrated that intermediate layers (approximately 15–65% depth) are more informative for biological mechanistic interpretation, as performance on structural and regulatory tasks decreased in the final layers of the model (Dalla-Torre et al., 2025). Researchers focused on mechanistic interpretability often target intermediate layers, as these middle layers tend to exhibit the highest degree of polysemanticity (Bricken et al., 2023; Cunningham et al., 2023; Touretzky and Pomerleau, 1991).

Information can be extracted from the embeddings of two sequences with minimal processing using Euclidean distance or cosine similarity

Table 2
Embedding extraction strategies across representative biological language models, highlighting layer selection, pooling approaches, and downstream analytical frameworks. Block Used indicates the transformer layer from which embeddings were extracted (e.g., 26 (52%), 24 (75%), 48 (100%)), where the percentage denotes the relative depth within the full model. Pooling Method describes how token-level representations were combined into a sequence-level embedding, and Classification Method specifies how the extracted embeddings were subsequently processed to generate predictions.

Ref	Model Name	Total Blocks	Block Used	Biological Predictions	Pooling Method	Classification Method
Modern Transformer-Based Models						
(Brixi et al., 2026)	Evo 2 40B	50	26 (52%)	Exon-intron boundaries, TF binding, BRCA1 variants	Mean Pooling	Sparse Autoencoders (SAEs)
(Nguyen et al., 2024)	Evo	32	24 (75%)	CRISPR Repeat Detection, Prophage Sites	Mean Pooling	Sparse Autoencoders (SAEs), Zero-Shot
(Lin et al., 2022)	ESM-2 15B	48	48 (100%)	Subcellular Localization, Structure, Mutation Effect	CLS Token / Mean	Euclidean Distance, Linear Probes
(Elnaggar et al., 2022)	ProtT5-XL	24	24 (100%)	Secondary Structure, Conservation, Stability	Concatenation	CNN, Logistic Regression
(Dalla-Torre et al., 2025)	Nucleotide Transformer	32	All / Final	Enhancers, Splice Sites, Chromatin State	Mean Pooling	MLP, Logistic Regression
(Rao et al., 2021)	MSA Transformer	12	12 (100%)	Protein Contact Prediction, Homology	Column-wise Mean	Row-Column Symmetrization
(Zhou et al., 2023)	DNABERT-2	12	12 (100%)	Transcription Factor Binding, Promoters	Mean / Max Pooling	MLP, CAMP (CNN-Attention-Mean)
(Nguyen et al., 2023)	HyenaDNA	2	2 (100%)	Species Identification, Regulatory Elements	Mean Pooling	Linear Decoder
(Shaw et al., 2025)	ESM C 600 M	36	Final / Last 4 / Last 8	Functional Similarity	PSWE	Cosine Similarity
(Outeiral and Deane, 2024)	CaLM	12	12	Protein Structure Prediction	Mean Pooling	Logistic Regression
Methodologically Distinct Frameworks						
(Bepler and Berger, 2019)	Bi-LSTM	4	4	Protein Structural Similarity, Transmembrane Domains	Mean Pooling	CNN, BiLSTM-CRF
(Yang et al., 2018)	Doc2Vec	2	2	Protein Functional Properties	Mean Pooling	Cosine Similarity

(Fig. 2). Euclidean distance measures the absolute separation between two vectors in latent space, whereas cosine similarity quantifies the angular similarity between them and effectively normalizes for vector magnitude. Recent findings suggest that Euclidean distance is more representative of true similarity than cosine measurements because it factors in both magnitude and angle (Steck et al., 2024). A reason to use cosine similarity over Euclidean distance is when the magnitude is considered noise (Wang et al., 2020). Both of these methods assume that the latent space is isotropic (i.e., all components of the embedding vector have equal variance). One approach to account for this shortcoming is to use the embedding-based alignment (EBA) *Z*-score method (Pantolini et al., 2024). Here, the variability of the components of the embedding vector is normalized, so that a single high variability dimension does not dominate distance or angle comparisons.

Alternatively, embeddings can be used as inputs to relatively simple machine learning models. Examples include a two-layer convolutional neural network trained on the mean pooling of the last hidden layer of ProtT5-XL, a multi-layer perceptron stacking neural network trained on the mean pooling of the last hidden layer of ESM-1b, and the PaCRISPR classifier, an ensemble model that uses mean-pooled embeddings from layer 6 of Evo 1.5 (Table 2) (de Oliveira et al., 2025; Marquet et al., 2022; Merchant et al., 2024).

A fundamental issue with using raw embeddings is *polysemanticity* (Cunningham et al., 2023). Polysemanticity refers to the fact that a single component of an embedding vector can have different meanings depending on the pattern of the other components in the embedding vector. Essentially, to allow a language model to represent more concepts than it has model dimensions, or components in the embedding dimension, LLMs have been found to use the pattern of embedding components to represent single concepts rather than using individual components of the embedding vector to encode a single concept. This phenomenon is called *superposition* (Elhage et al., 2022).

Sparse autoencoders (SAE) have been shown to be able to extract singular concepts from embedding vectors (Cunningham et al., 2023). The goal of an SAE is to expand a given embedding vector, so that the individual concepts that are in superposition can be separated into individual components of a larger vector. Recently published SAEs are single hidden layer neural network models that are designed to expand the embedding vector and then recreate it. By applying a constraint known as BatchTopK, the SAE ensures that only a few specific features are active at any given time (Bussmann et al., 2024). This allows researchers to move from abstract math to identifiable biology, isolating specific interpretable units such as CRISPR repeats, transcription factor motifs, or prophage sites (Brixi et al., 2026).

2.6. How have embeddings been used for biological discovery and translational biotechnology?

Genome and protein language models are designed for generating novel sequences, but to do this they need to be able to understand sequence motifs at a deep level to make biologically functional sequences. Embeddings have been shown to encode information related to function, form, and location of proteins.

2.7. Variant fitness, conservation, and clinical diagnostics

At the single-nucleotide and residue scale, embeddings provide a powerful framework for evaluating mutational tolerance and predicting variant fitness. Deep Mutational Scanning (DMS) assays provide quantitative ground truths that demonstrate the validity of these models. For instance, zero-shot predictions using the raw logits of autoregressive models like Evo 2 can accurately evaluate the deleteriousness of mutations within the BRCA1 gene without requiring task-specific fine-tuning (Brixi et al., 2026; Meier et al., 2021). Alternatively, extracting token-level embeddings and applying simple downstream models, such as logistic regression, yields sequence conservation predictions that match or exceed the accuracy of established, MSA-dependent tools directly from a single sequence (Marquet et al., 2022).

In clinical diagnostics, this predictive capacity enables the high-throughput triage of Variants of Uncertain Significance (VUS) (Brixi et al., 2026; Cheng et al., 2023). By mapping patient-specific mutations directly into a latent space trained on millions of evolutionary sequences, clinicians can rapidly predict whether a variant is benign or pathogenic, effectively translating raw sequencing files into actionable medical insights.

2.8. Protein function, annotation transfer, and de novo synthetic discovery

Beyond point mutations, embeddings capture higher-level semantic features that allow for functional characterization even when sequences share low sequence identity. Models like Tranception leverage these sequence-level representations to predict broad protein fitness across diverse datasets (Notin et al., 2022). This allows researchers to transfer Gene Ontology (GO) annotations based on embedding similarity rather than explicit homology matches, bypassing traditional alignment bottlenecks (Littmann et al., 2021).

In synthetic biology, this functional mapping enables the semantic mining of completely unexplored sequence spaces. For example, recent work utilizing the Evo language model led to the construction of *SynGenome*, a comprehensive database of AI-generated synthetic DNA sequences (Merchant et al., 2024). By applying embedding-based screening to this dataset, researchers bypassed traditional homology constraints to identify and experimentally validate highly divergent, functional de novo toxin-antitoxin systems. This integrated workflow, where generative models write raw sequences and embedding analytics prioritize them, establishes a scalable roadmap for discovering novel biosensors, industrial enzymes, and therapeutics that do not exist in natural history.

2.9. Biophysical properties (solubility, localization, binding sites) and enzyme engineering

For structural and biochemical engineering, residue-level embeddings natively encode localized physical constraints. In *bindEmbed21*, per-residue embeddings extracted from ProtT5 were passed into task-specific deep learning classifiers to predict whether individual residues bind metal ions, nucleic acids, or small molecules, matching the performance of MSA-based methods directly from primary text (Littmann et al., 2021). Similarly, combining residue representations via attention-based pooling allows for the accurate prediction of subcellular localization (Thumuluri et al., 2022a), while simple mean pooling operations yield feature vectors that accurately predict macroscopic biophysical properties like protein solubility in *E. coli* expression systems (Thumuluri et al., 2022b).

In industrial biotechnology and enzyme engineering, these biophysical predictions provide an operational decision-making framework. Instead of executing exhaustive wet-lab screening, researchers can implement embedding-guided prioritization to rank thousands of sequences in silico based on their latent fitness and solubility profiles (Meier et al., 2021). This dramatically compresses the experimental search space, ensuring that only the most viable candidates are advanced to targeted mutagenesis, directed evolution, and downstream purification.

3. Conclusion and perspectives

Applying techniques from the field of mechanistic interpretability to genome and protein language models has been very productive. These models learn high-dimensional representations that encode regulatory motifs, structural constraints, evolutionary relationships, and latent functional signals. Biological insights can be extracted by studying these

internal structures, and not only treating models as black boxes. Interpretability is also central to the broader alignment and safety discourse in machine learning (Hubinger et al., 2024). If we understand how representations encode meaning in language models, we can ensure they will give outputs that are reliable and value-aligned in high-stakes biological and health applications.

Embeddings have the potential to narrow the gap between well-studied model organisms and understudied species, but doing so will require computational frameworks that ensure these representations generalize across diverse taxa and can be converted into reliable functional annotations. AI-generated annotations derived from embeddings could be used to expand and update protein and gene databases, helping to fill long-standing gaps in knowledge. However, these resources must be developed responsibly. AI-derived annotations should be explicitly labeled, versioned, and linked to model provenance, and reproducible code and model checkpoints should accompany publications to enable verification. Critically, computational predictions must be paired with experimental validation. Rather than replacing biological experimentation, interpretability-driven hypotheses can guide experiments, for example, by designing targeted CRISPR screens, mutagenesis assays, and high-throughput phenotyping strategies. Establishing hybrid computational-experimental pipelines will be essential for translating representation-level discoveries into robust biological knowledge.

The integration of genome and protein language models into clinical or therapeutic decision-making raises profound challenges. If such models are ever used to inform disease diagnosis, medication strategies, or gene therapy design, we must first understand how they arrive at their conclusions. Open questions remain regarding feature polysemy, circuit-level organization, and the degree to which outputs can be traced to training data sources. Developing tools for attribution, memorization detection, and data lineage tracking will be critical for trust and governance. Ultimately, the potential impact of genome and protein language models will depend not solely on scaling model size, but on our ability to interpret, audit, and experimentally ground their internal representations.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was partially supported by the USDA Western Sun Grant. This work was also supported in part by the S1090 Multistate Research Project, “AI in Agroecosystems: Big Data and Smart Technology-Driven Sustainable Production.”

Data availability

No data was used for the research described in the article.

References

Avsec, Ž., Latysheva, N., Cheng, J., Novati, G., Taylor, K.R., Ward, T., Bycroft, C., Nicolaisen, L., Arvaniti, E., Pan, J., Thomas, R., Dutordoir, V., Perino, M., De, S., Karollus, A., Gayoso, A., Sargeant, T., Mottram, A., Wong, L.H., Drotár, P., Kosiorek, A., Senior, A., Tanburn, R., Applebaum, T., Basu, S., Hassabis, D., Kohli, P., 2026. Advancing regulatory variant effect prediction with AlphaGenome. Nature 649 (8099), 1206–1218.

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

Brandes, N., Ofer, D., Peleg, Y., Rappoport, N., Linial, M., 2022. ProteinBERT: a universal deep-learning model of protein sequence and function. Bioinformatics 38 (8), 2102–2110.

Bricken, T., Templeton, A., Batson, J., Chen, B., Jermyn, A., Conerly, T., Turner, N., Anil, C., Denison, C., Askell, A., Lasenby, R., Wu, Y., Kravec, S., Schiefer, N., Maxwell, T., Joseph, N., Hatfield-Dodds, Z., Tamkin, A., Nguyen, K., McLean, B., Burke, J.E., Hume, T., Carter, S., Henighan, T., Olah, C., 2023. Towards Monosemanticity: Decomposing Language Models with Dictionary Learning. (Transformer Circuits Thread).

Brixi, G., Durrant, M.G., Ku, J., Poli, M., Brockman, G., Chang, D., Gonzalez, G.A., King, S.H., Li, D.B., Merchant, A.T., Naghipourfar, M., Nguyen, E., Ricci-Tam, C., Romero, D.W., Sun, G., Taghibakshi, A., Vorontsov, A., Yang, B., Deng, M., Gorton, L., Nguyen, N., Wang, N.K., Adams, E., Baccus, S.A., Dillmann, S., Ermon, S., Guo, D., Ilango, R., Janik, K., Lu, A.X., Mehta, R., Mofrad, M.R.K., Ng, M.Y., Pannu, J., Ré, C., Schmok, J.C., John, J.S., Sullivan, J., Zhu, K., Zynda, G., Balsam, D., Collison, P., Costa, A.B., Hernandez-Boussard, T., Ho, E., Liu, M.-Y., McGrath, T., Powell, K., Burke, D.P., Goodarzi, H., Hsu, P.D., Hie, B.L., 2026. Genome modeling and design across all domains of life with evo 2. Nature 652, 1349–1361.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., 2020. Language models are few-shot learners. Adv. Neural Inf. Proces. Syst. 33, 1877–1901.

Bussmann, B., Leask, P., Nanda, N., 2024. Batchtopk sparse autoencoders. arXiv preprint arXiv:2412.06410.

Cheng, J., Novati, G., Pan, J., Bycroft, C., Žemgulytė, A., Applebaum, T., Pritzel, A., Wong, L.H., Zielinski, M., Sargeant, T., Schneider, R.G., Senior, A.W., Jumper, J., Hassabis, D., Kohli, P., Avsec, Ž., 2023. Accurate proteome-wide missense variant effect prediction with AlphaMissense. Science 381, eadg7492.

Cunningham, H., Ewart, A., Riggs, L., Huben, R., Sharkey, L., 2023. Sparse autoencoders find highly interpretable features in language models. arXiv preprint arXiv:2309.08600.

Dalla-Torre, H., Gonzalez, L., Mendoza-Revilla, J., Lopez Carranza, N., Grzywaczewski, A.H., Oteri, F., Dallago, C., Trop, E., de Almeida, B.P., Sirelkhatim, H., Richard, G., Skwark, M., Beguir, K., Lopez, M., Pierrot, T., 2025. Nucleotide transformer: building and evaluating robust foundation models for human genomics. Nat. Methods 22 (2), 287–297.

de Oliveira, G.B., Pedrini, H., Dias, Z., 2025. SUPERMAGO: protein function prediction based on transformer embeddings. Proteins: Struct. Funct. Bioinf. 93 (5), 981–996.

Dejean, T., Ferrell, B.D., Harrigan, W., Schreiber, Z.D., Sawhney, R., Wommack, K.E., Polson, S.W., Belcaid, M., 2025. Extending protein language models to a viral genomic scale using biologically induced sparse attention. bioRxiv, 2025.2005.2029.656907.

Elhage, N., Hume, T., Olsson, C., Schiefer, N., Henighan, T., Kravec, S., Hatfield-Dodds, Z., Lasenby, R., Drain, D., Chen, C., 2022. Toy models of superposition. arXiv preprint arXiv:2209.10652.

Elnaggar, A., Heinzinger, M., Dallago, C., Rehawi, G., Wang, Y., Jones, L., Gibbs, T., Feher, T., Angerer, C., Steinegger, M., Bhowmik, D., Rost, B., 2022. ProtTrans: toward understanding the language of life through self-supervised learning. IEEE Trans. Pattern Anal. Mach. Intell. 44 (10), 7112–7127.

Harrigan, W.L., Ferrell, B.D., Wommack, K.E., Polson, S.W., Schreiber, Z.D., Belcaid, M., 2024. Improvements in viral gene annotation using large language models and soft alignments. BMC Bioinf. 25 (1), 165.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. Proc. IEEE Conf. Comput. Vis. Pattern Recognit. 770–778.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780.

Hubinger, E., Denison, C., Mu, J., Lambert, M., Tong, M., MacDiarmid, M., Lanham, T., Ziegler, D.M., Maxwell, T., Cheng, N., 2024. Sleeper agents: training deceptive llms that persist through safety training. arXiv preprint arXiv:2401.05566.

Ji, Y., Zhou, Z., Liu, H., Davuluri, R.V., 2021. DNABERT: pre-trained bidirectional encoder representations from transformers model for DNA-language in genome. Bioinformatics 37 (15), 2112–2120.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S.A.A., Ballard, A.J., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., Back, T., Petersen, S., Reiman, D., Clancy, E., Zielinski, M., Steinegger, M., Pacholska, M., Berghammer, T., Bodenstein, S., Silver, D., Vinyals, O., Senior, A.W., Kavukcuoglu, K., Kohli, P., Hassabis, D., 2021. Highly accurate protein structure prediction with AlphaFold. Nature 596, 583–589.

LeCun, Y.A., Bottou, L., Orr, G.B., Müller, K.-R., 2012. Efficient BackProp. In: Montavon, G., Orr, G.B., Müller, K.-R. (Eds.), Neural Networks: Tricks of the Trade: Second Edition. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 9–48.

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Santos Costa, A.d., Fazel-Zarandi, M., Sercu, T., Candido, S., Rives, A., 2022. Language models of protein sequences at the scale of evolution enable accurate structure prediction. bioRxiv, 2022.2007.2020.500902.

Littmann, M., Heinzinger, M., Dallago, C., Weissenow, K., Rost, B., 2021. Protein embeddings and deep learning predict binding residues for various ligand classes. Sci. Rep. 11 (1), 23916.

Marquet, C., Heinzinger, M., Olenyi, T., Dallago, C., Erckert, K., Bernhofer, M., Nechaev, D., Rost, B., 2022. Embeddings from protein language models predict conservation and variant effects. Hum. Genet. 141 (10), 1629–1647.

McCulloch, W.S., Pitts, W., 1943. A logical calculus of the ideas immanent in nervous activity. Bull. Math. Biophys. 5 (4), 115–133.

Meier, J., Rao, R., Verkuil, R., Liu, J., Sercu, T., Rives, A., 2021. Language models enable zero-shot prediction of the effects of mutations on protein function. bioRxiv, 2021.2007.2009.450648.

Merchant, A.T., King, S.H., Nguyen, E., Hie, B.L., 2024. Semantic mining of functional *de novo* genes from a genomic language model. bioRxiv, 2024.2012.2017.628962.

Mielke, S.J., Alyafeai, Z., Salesky, E., Raffel, C., Dey, M., Gallé, M., Raja, A., Si, C., Lee, W.Y., Sagot, B., 2021. Between words and characters: a brief history of open-vocabulary modeling and tokenization in NLP. arXiv Preprint arXiv:2112.10508.

Nguyen, E., Poli, M., Faizi, M., Thomas, A., Birch-Sykes, C., Wornow, M., Patel, A., Rabideau, C., Massaroli, S., Bengio, Y., Ermon, S., Baccus, S.A., Ré, C., 2023. HyenaDNA: Long-Range Genomic Sequence Modeling at Single Nucleotide Resolution (ArXiv).
Nguyen, E., Poli, M., Durrant, M.G., Kang, B., Katrekar, D., Li, D.B., Bartie, L.J., Thomas, A.W., King, S.H., Brixi, G., Sullivan, J., Ng, M.Y., Lewis, A., Lou, A., Ermon, S., Baccus, S.A., Hernandez-Boussard, T., Ré, C., Hsu, P.D., Hie, B.L., 2024. Sequence modeling and design from molecular to genome scale with evo. Science 386 (6723), eado9336.
Notin, P., Dias, M., Frazer, J., Marchena-Hurtado, J., Gomez, A.N., Marks, D., Gal, Y., 2022. Tranception: protein fitness prediction with autoregressive transformers and inference-time retrieval. Int. Conf. Machine Learn. PMLR 16990–17017.
Outeiral, C., Deane, C.M., 2024. Codon language embeddings provide strong signals for use in protein engineering. Nat. Mach. Intell. 6 (2), 170–179.
Pantolini, L., Studer, G., Pereira, J., Durairaj, J., Tauriello, G., Schwede, T., 2024. Embedding-based alignment: combining protein language models with dynamic programming alignment to detect structural similarities in the twilight-zone. Bioinformatics 40 (1).
Poli, M., Massaroli, S., Nguyen, E., Fu, D.Y., Dao, T., Baccus, S., Bengio, Y., Ermon, S., Ré, C., 2023. Hyena hierarchy: towards larger convolutional language models. Int. Conf. Machine Learn. PMLR 28043–28078.
Rao, R., Bhattacharya, N., Thomas, N., Duan, Y., Chen, X., Canny, J., Abbeel, P., Song, Y. S., 2019. Evaluating protein transfer learning with TAPE. Adv. Neural Inf. Proces. Syst. 32, 9689–9701.
Rao, R., Liu, J., Verkuil, R., Meier, J., Canny, J.F., Abbeel, P., Sercu, T., Rives, A., 2021. MSA Transformer. bioRxiv, 2021.2002.2012.430858.
Rives, A., Meier, J., Sercu, T., Goyal, S., Lin, Z., Liu, J., Guo, D., Ott, M., Zitnick, C.L., Ma, J., Fergus, R., 2021. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. Proc. Natl. Acad. Sci. USA 118 (15).
Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning representations by back-propagating errors. Nature 323 (6088), 533–536.
Shaw, R., Love, S.D., McWhite, C.D., 2025. Evaluating pretrained protein language model embeddings as proxies for functional similarity. J. Mol. Evol. 93 (6), 765–776.
Steck, H., Ekanadham, C., Kallus, N., 2024. Is cosine-similarity of embeddings really about similarity? Compan. Proc. ACM Web Conf. 2024, 887–890.
Tang, H., Glass, J., 2018. On training recurrent networks with truncated backpropagation through time in speech recognition, 2018 IEEE spoken language technology workshop (SLT). IEEE 48–55.
Thumuluri, V., Almagro Armenteros, J.J., Johansen, Alexander R., Nielsen, H., Winther, O., 2022a. DeepLoc 2.0: multi-label subcellular localization prediction using protein language models. Nucleic Acids Res. 50 (W1), W228–W234.
Thumuluri, V., Martiny, H.-M., Almagro Armenteros, J.J., Salomon, J., Nielsen, H., Johansen, A.R., 2022b. NetSolP: predicting protein solubility in Escherichia coli using language models. Bioinformatics 38 (4), 941–946.
Touretzky, D.S., Pomerleau, D.A., 1991. What's hidden in the hidden layers? AI Mag. 12 (3), 49–62.
Turing, A.M., 1950. I.—Computing Machinery and Intelligence. Mind LIX (236), 433–460.
van Heyningen, V., Bickmore, W., 2013. Regulation from a distance: long-range control of gene expression in development and disease. Philos. Trans. R. Soc. Lond. Ser. B Biol. Sci. 368 (1620), 20120372.
Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Adv. Neural Inf. Proces. Syst. 30.
Villegas-Morcillo, A., Gomez, A.M., Sanchez, V., 2022. An analysis of protein language model embeddings for fold prediction. Brief. Bioinform. 23 (3), bbac142.
Wang, J., Dai, W., Li, J., Xie, R., Dunstan, R.A., Stubenrauch, C., Zhang, Y., Lithgow, T., 2020. PaCRISPR: a server for predicting and visualizing anti-CRISPR proteins. Nucleic Acids Res. 48 (W1), W348–w357.
Weissenow, K., Heinzinger, M., Rost, B., 2022. Protein language-model embeddings for fast, accurate, and alignment-free protein structure prediction. Structure 30 (8), 1169–1177.e1164.
Yang, K.K., Wu, Z., Bedbrook, C.N., Arnold, F.H., 2018. Learned protein embeddings for machine learning. Bioinformatics 34 (15), 2642–2648.
Zhou, Z., Ji, Y., Li, W., Dutta, P., Davuluri, R., Liu, H., 2023. Dnabert-2: efficient foundation model and benchmark for multi-species genome. arXiv Preprint arXiv: 2306.15006.

Glossary of technical terms

Fundamental Machine Learning & Neural Networks:

Architecture: The structural design of an AI model (e.g., Transformer, CNN).

Backpropagation & Gradient Descent: The primary algorithms used to train models; backpropagation calculates the error, and gradient descent adjusts the model to minimize that error.

Weights & Biases: The internal "knobs" of a model. Weights determine the strength of a connection between data points, while biases allow the model to shift its decision-making threshold.

Hidden Layers: The intermediate layers between input and output where the model extracts increasingly complex features from the data.

Parameters: The total number of weights and biases in a model. The number of parameters in a model is often used as a proxy for its representational capacity.

Features: Individual measurable properties or characteristics of the data being analyzed.

Architecture & Sequence Modeling:

Tokens & Tokenization: The process of breaking a sequence (DNA or protein) into smaller units. k-mer tokenization uses fixed-length overlapping segments, while single-character tokenization treats each nucleotide or amino acid individually.

Context Window: The maximum number of tokens a model can "see" and process at one time before degrading in quality.

Self-Attention: A mechanism that allows a model to weigh the importance of different tokens in a sequence relative to one another.

Layers/Blocks: Repeating units of the architecture (often including attention and feed-forward networks) that build hierarchical representations.

Model Dimension: The size of the vector used to represent each token within the hidden layers.

Recurrent Neural Networks (RNN) / LSTM: An architecture designed for sequential data that process tokens one by one that's been largely succeeded by Transformers in biological AI applications.

Training Objectives & Models:

Autoregressive: A model trained to predict the "next" token in a sequence based on previous tokens (e.g., GPT-style).

Masked (MLM): A model trained to predict "hidden" or corrupted tokens using information from both directions (e.g., BERT-style).

Auto-encoder: A model trained to compress input data into a lower-dimensional representation and then reconstruct the original input from that compression.

Zero-shot Prediction: Using a pretrained model to make a prediction on a task it was not specifically trained for, without further fine-tuning.

Model Logits: The raw, unnormalized scores produced by the last layer of a model before they are converted into probabilities.

Embeddings & Mathematical Spaces:

Latent Space: The high-dimensional representation space where the model organizes its embeddings based on their learned relationships.

Cosine Similarity & Euclidean Distance: Mathematical metrics used to measure how "similar" two embeddings are within the latent space.

Token Pooling: A method of aggregating individual token embeddings into a single vector that represents an entire protein or genomic region.

Embedding-Based Alignment: Aligning sequences by comparing their high-dimensional embeddings rather than their raw character-level matches.

Multiple Sequence Alignment (MSA): A traditional bioinformatic method of arranging DNA/protein sequences to identify regions of similarity that may indicate evolutionary relationships.

Interpretability & Biological Context:

Mechanistic Interpretability: The study of "opening the black box" to understand how specific internal components of a model (like attention heads) correspond to biological features.

Sparse Autoencoder (SAE): A tool used in interpretability to "disentangle" complex embeddings into human-understandable biological concepts.

Polysemanticity: When a single neuron in a model responds to multiple distinct biological concepts or patterns, making it harder to interpret.