Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences

Alexander Rives[a,b,1,2], Joshua Meier[a,1], Tom Sercu[a,1], Siddharth Goyal[a,1], Zeming Lin[b], Jason Liu[a], Demi Guo[c,3], Myle Ott[a], C. Lawrence Zitnick[a], Jerry Ma[d,e,3], and Rob Fergus[b]

[a]Facebook AI Research, New York, NY 10003; [b]Department of Computer Science, New York University, New York, NY 10012; [c]Harvard University, Cambridge, MA 02138; [d]Booth School of Business, University of Chicago, Chicago, IL 60637; and [e]Yale Law School, New Haven, CT 06511

Edited by David T. Jones, University College London, London, United Kingdom, and accepted by Editorial Board Member William H. Press December 16, 2020 (received for review August 6, 2020)

In the field of artificial intelligence, a combination of scale in data and model capacity enabled by unsupervised learning has led to major advances in representation learning and statistical generation. In the life sciences, the anticipated growth of sequencing promises unprecedented data on natural sequence diversity. Protein language modeling at the scale of evolution is a logical step toward predictive and generative artificial intelligence for biology. To this end, we use unsupervised learning to train a deep contextual language model on 86 billion amino acids across 250 million protein sequences spanning evolutionary diversity. The resulting model contains information about biological properties in its representations. The representations are learned from sequence data alone. The learned representation space has a multiscale organization reflecting structure from the level of biochemical properties of amino acids to remote homology of proteins. Information about secondary and tertiary structure is encoded in the representations and can be identified by linear projections. Representation learning produces features that generalize across a range of applications, enabling state-of-the-art supervised prediction of mutational effect and secondary structure and improving state-of-the-art features for long-range contact prediction.

generative biology | representation learning | protein language model | deep learning | synthetic biology

Growth in the number of protein sequences in public databases has followed an exponential trend over decades, creating a deep view into the breadth and diversity of protein sequences across life. These data are a promising ground for studying predictive and generative models for biology using artificial intelligence. Our focus here will be to fit a single model to many diverse sequences from across evolution. Accordingly we study high-capacity neural networks, investigating what can be learned about the biology of proteins from modeling evolutionary data at scale.

The idea that biological function and structure are recorded in the statistics of protein sequences selected through evolution has a long history (1–3). Out of the possible random perturbations to a sequence, evolution is biased toward selecting those that are consistent with fitness (4). The unobserved variables that determine a protein's fitness, such as structure, function, and stability, leave a record in the distribution of observed natural sequences (4).

Unlocking the information encoded in protein sequence variation is a longstanding problem in biology. An analogous problem in the field of artificial intelligence is natural language understanding, where the distributional hypothesis posits that a word's semantics can be derived from the contexts in which it appears (5).

Recently, techniques based on self-supervision, a form of unsupervised learning in which context within the text is used to predict missing words, have been shown to materialize representations of word meaning that can generalize across natural language tasks (6–9). The ability to learn such representations improves significantly with larger training datasets (10, 11).

Protein sequences result from a process greatly dissimilar to natural language. It is uncertain whether the models and objective functions effective for natural language transfer across differences between the domains. We explore this question by training high-capacity Transformer language models on evolutionary data. We investigate the resulting unsupervised representations for the presence of biological organizing principles and information about intrinsic biological properties. We find metric structure in the representation space that accords with organizing principles at scales from physicochemical to remote homology. We also find that secondary and tertiary protein structure can be identified in representations. The structural properties captured by the representations generalize across folds. We apply the representations to a range of prediction tasks and find that they improve state-of-art features across the applications.

Background

Sequence alignment and search is a standard basis for comparative and statistical analysis of biological sequence data (12–15).

Significance

Learning biological properties from sequence data is a logical step toward generative and predictive artificial intelligence for biology. Here, we propose scaling a deep contextual language model with unsupervised learning to sequences spanning evolutionary diversity. We find that without prior knowledge, information emerges in the learned representations on fundamental properties of proteins such as secondary structure, contacts, and biological activity. We show the learned representations are useful across benchmarks for remote homology detection, prediction of secondary structure, long-range residue–residue contacts, and mutational effect. Unsupervised representation learning enables state-of-the-art supervised prediction of mutational effect and secondary structure and improves state-of-the-art features for long-range contact prediction.

Author contributions: A.R., J. Meier, T.S., S.G., Z.L., M.O., C.L.Z., J. Ma, and R.F. designed research; A.R., J. Meier, T.S., S.G., Z.L., J.L., D.G., and J. Ma performed research; A.R., J. Meier, T.S., S.G., Z.L., J.L., D.G., and J. Ma analyzed data; and A.R., J. Meier, T.S., S.G., Z.L., J.L., D.G., M.O., C.L.Z., J. Ma, and R.F. wrote the paper.

Competing interest statement: A.R., J. Meier, S.G., D.G., M.O., C.L.Z., J. Ma, and R.F. are coinventors on a US patent application relating to the work of this manuscript.

This article is a PNAS Direct Submission. D.T.J. is a guest editor invited by the Editorial Board.

[1]A.R., J. Meier., T.S., and S.G. contributed equally to this work.

[2]To whom correspondence may be addressed. Email: arives@cs.nyu.edu.

[3]Work performed while at Facebook AI Research.

This article contains supporting information online at https://www.pnas.org/lookup/suppl/doi:10.1073/pnas.2016239118/-/DCSupplemental.

Published April 5, 2021.

Search across large databases containing evolutionary diversity assembles related sequences into a multiple sequence alignment (MSA). Within sequence families, mutational patterns convey information about functional sites, stability, tertiary contacts, binding, and other properties (2–4). Conserved sites correlate with functional and structural importance (2). Local biochemical and structural contexts are reflected in preferences for distinct classes of amino acids (16). Covarying mutations have been associated with function, tertiary contacts, and binding (4).

The prospect of inferring biological structure and function from evolutionary statistics has motivated development of machine learning on individual sequence families. Direct coupling analysis (17–19) infers constraints on the structure of a protein by fitting a generative model in the form of a Markov random field (MRF) to the sequences in the protein's MSA. Various methods have been developed to fit the MRF (20–23). The approach can also be used to infer functional constraints (24, 25), and the generative picture can be extended to include latent variables (26).

Recently, self-supervision has emerged as a core direction in artificial intelligence research. Unlike supervised learning, which requires manual annotation of each datapoint, self-supervised methods use unlabeled datasets and therefore can exploit far larger amounts of data. Self-supervised learning uses proxy tasks for training such as predicting the next word in a sentence given all previous words (8, 9, 11, 27, 28) or predicting words that have been masked from their context (6, 29).

Increasing the dataset size and the model capacity has shown improvements in the learned representations. In recent work, self-supervision methods used in conjunction with large data and high-capacity models produced new state-of-the-art results approaching human performance on various question answering and semantic reasoning benchmarks (6) and coherent natural text generation (11).

This paper explores self-supervised language modeling approaches that have demonstrated state-of-the-art performance on a range of natural language processing tasks, applying them to protein data in the form of unlabeled amino acid sequences. Since protein sequences use a small vocabulary of 20 canonical elements, the modeling problem is more similar to character-level language models (30, 31) than word-level models. Like natural language, protein sequences also contain long-range dependencies, motivating use of architectures that detect and model distant context (32).

Scaling Language Models to 250 Million Diverse Protein Sequences

Large protein sequence databases contain diverse sequences sampled across life. In our experiments, we explore datasets with up to 250 million sequences of the UniParc database (33), which has 86 billion amino acids. These data are comparable in size to large text datasets that are being used to train high-capacity neural network architectures on natural language (6, 11). To model the data of evolution with fidelity, neural network architectures must have capacity and inductive biases to represent its breadth and diversity.

We investigate the Transformer (32), which has emerged as a powerful general purpose model architecture for representation learning and generative modeling, outperforming recurrent and convolutional architectures in natural language settings. We use a deep Transformer (6), taking as input amino acid character sequences.

The Transformer processes inputs through a series of blocks that alternate self-attention with feed-forward connections. Self-attention allows the network to build up complex representations that incorporate context from across the sequence. Since self-attention explicitly constructs pairwise interactions between all positions in the sequence, the Transformer architecture directly represents residue–residue interactions.

We train models using the masked language modeling objective (6). Each input sequence is corrupted by replacing a fraction of the amino acids with a special mask token. The network is trained to predict the missing tokens from the corrupted sequence:

$$\mathcal{L}_{MLM} = \mathbb{E}_{x\sim X}\mathbb{E}_M \sum_{i\in M} -logp\left(x_i \middle| x_{/M}\right). \quad [1]$$

For each sequence x, we sample a set of indices M to mask, replacing the true token at each index i with the mask token. For each masked token, we independently minimize the negative log likelihood of the true amino acid x_i given the masked sequence $x_{/M}$ as context. Intuitively, to make a prediction for a masked position, the model must identify dependencies between the masked site and the unmasked parts of the sequence.

Evaluation of Language Models. We begin by training a series of Transformers on all the sequences in UniParc (33), holding out a random sample of 1 M sequences for validation. We use these models throughout to investigate properties of the representations and the information learned during pretraining.

To comparatively evaluate generalization performance of different language models, we use UniRef50 (34), a clustering of UniParc at 50% sequence identity. For evaluation, a held-out set of 10% of the UniRef50 clusters is randomly sampled. The evaluation dataset consists of the representative sequences of these clusters. All sequences belonging to the held-out clusters are removed from the pretraining datasets.

We explore the effect of the underlying sequence diversity in the pretraining data. Clustering UniParc shows a power law distribution of cluster sizes (35), implying the majority of sequences belong to a small fraction of clusters. Training using a clustering of the sequences results in a reweighting of the masked language modeling loss toward a more diverse set of sequences. We use UniRef (34) to create three pretraining datasets with differing levels of diversity: 1) the low-diversity dataset (UR100) uses the UniRef100 representative sequences, 2) the high-diversity sparse dataset (UR50/S) uses the UniRef50 representative sequences, and 3) the high-diversity dense dataset (UR50/D) samples the UniRef100 sequences evenly across the UniRef50 clusters.

Table 1 presents modeling performance on the held-out UniRef50 sequences across a series of experiments exploring different model classes, number of parameters, and pretraining datasets. Models are compared using the exponentiated cross entropy (ECE) metric, which is the exponential of the model's loss averaged per token. In the case of the Transformer, this is $2^{\mathcal{L}_{MLM}}$. ECE describes the mean uncertainty of the model among its set of options for every prediction: ranging from one for an ideal model to 25 (the number of unique amino acid tokens in the data) for a completely random prediction. To measure the difficulty of generalization to the evaluation set, we train a series of n-gram models across a range of context lengths and settings of Laplace smoothing on UR50/S. The best n-gram model has an ECE of 17.18 with context size of four.

As a baseline, we train recurrent long short-term memory (LSTM) bidirectional language models (9), which are state of the art for recurrent models in the text domain. Unlike standard left-to-right autoregressive LSTMs, these models use the whole sequence context, making them comparable to the Transformers we study. We evaluate a small model with ∼25 M parameters and a large model with ∼113 M parameters. Trained on the UR50/S dataset, the small and large LSTM models have an ECE of 14.4 and 13.5, respectively.

We also train two small Transformers, a 12-layer (85.1 M parameters) and six-layer Transformer (42.6 M parameters) on

Table 1. Evaluation of language models for generalization to held-out UniRef50 clusters

	Model		Params	Training	ECE
(a)	Oracle				1
	Uniform Random				25
(b)	*n*-gram	4-gram		UR50/S	17.18
(c)	LSTM	Small	28.4 M	UR50/S	14.42
	LSTM	Large	113.4 M	UR50/S	13.54
(d)	Transformer	6-layer	42.6 M	UR50/S	11.79
	Transformer	12-layer	85.1 M	UR50/S	10.45
(e)	Transformer	34-layer	669.2 M	UR100	10.32
	Transformer	34-layer	669.2 M	UR50/S	8.54
	Transformer	34-layer	669.2 M	UR50/D	8.46
(f)	Transformer	10% data	669.2 M	UR50/S	10.99
	Transformer	1% data	669.2 M	UR50/S	15.01
	Transformer	0.1% data	669.2 M	UR50/S	17.50

(a) ECE ranges from 25 for a random model to 1 for a perfect model. (b) Best *n*-gram model across range of context sizes and Laplace smoothing settings. (c) State-of-the-art LSTM bidirectional language models (9). (d) Transformer model baselines with 6 and 12 layers. Small Transformer models have better performance than LSTMs despite having fewer parameters. (e) Transformers that are 34-layer models are trained on datasets of differing sequence diversity. Increasing the diversity of the training set improves generalization. High-capacity Transformer models outperform LSTMs and smaller Transformers. (f) Transformers that are 34-layer models are trained on reduced fractions of data. Increasing training data improves generalization.

the UR50/S dataset. Both Transformer models have better ECE values (10.45 and 11.79, respectively) than the small and large LSTM models, despite the large LSTM having more parameters. These results show the Transformer enables higher fidelity modeling of protein sequences for a comparable number of parameters.

We train high-capacity 34-layer Transformers (approximately 670 M parameters) across the three datasets of differing diversity. The high-capacity Transformer model trained on the UR50/S dataset outperforms the smaller Transformers, indicating an improvement in language modeling with increasing model capacity. Transformers trained on the two high-diversity datasets, UR50/S and UR50/D, improve generalization over the UR100 low-diversity dataset. The best Transformer trained on the most diverse and dense dataset reaches an ECE of 8.46, meaning that intuitively, the model is choosing among ∼8.46 amino acids for each prediction.

We also train a series of 34-layer Transformer models on 0.1, 1, and 10% of the UR50/S dataset, seeing the expected relationship between increased data and improved generalization performance. Underfitting is observed even for the largest models trained on 100% of UR50/S, suggesting potential for additional improvements with higher capacity models.

ESM-1b Transformer. Finally, we perform a systematic optimization of model hyperparameters on 100 M parameter models to identify a robust set of hyperparameters. The hyperparameter search is described in detail in *SI Appendix*, section B. We scale the hyperparameters identified by this search to train a model with ∼650 M parameters (33 layers) on the UR50/S dataset, resulting in the ESM-1b Transformer.

Multiscale Organization in Sequence Representations

The variation observed in large protein sequence datasets is influenced by processes at many scales, including properties that affect fitness directly, such as activity, stability, structure, binding, and other properties under selection (25, 36) as well as by contributions from phylogenetic bias (37), experimental and selection biases (38, 39), and sources of noise such as random genetic drift (40).

Unsupervised learning may encode underlying factors that, while unobserved, are useful for explaining the variation in sequences seen by the model during pretraining. We investigate the representation space of the network at multiple scales from biochemical to evolutionary homology to look for signatures of biological organization.

Neural networks contain inductive biases that impart structure to representations. Randomly initialized networks can produce features that perform well without any learning (41). To understand how the process of learning shapes the representations, it is necessary to compare representations before and after they have been trained. Furthermore, a basic level of intrinsic organization is expected in the sequence data itself as a result of biases in amino acid composition. To disentangle the role of frequency bias in the data we also compare against a baseline that maps each sequence to a vector of normalized amino acid counts.

Learning Encodes Biochemical Properties. The Transformer neural network represents the identity of each amino acid in its input and output embeddings. The input embeddings project the input amino acid tokens into the first Transformer block. The output embeddings project the final hidden representations back to logarithmic probabilities. The interchangeability of amino acids within a given structural or functional context in a protein depends on their biochemical properties (36). Self-supervision can be expected to capture these patterns to build a representation space that reflects biochemical knowledge.

To investigate if the network has learned to encode physicochemical properties in its representations, we project the weight matrix of the final embedding layer of the network into two dimensions with t-distributed stochastic neighbor embedding (t-SNE) (42). In Fig. 1, the structure of the embedding space reflects biochemical interchangeability with distinct clustering of hydrophobic and polar residues, aromatic amino acids, and organization by molecular weight and charge.

Biological Variations Are Encoded in Representation Space. Each protein can be represented as a single vector by averaging across the hidden representation at each position in its sequence. Protein embeddings represent sequences as points in a high dimensional space. Each sequence is represented as a single point, and sequences assigned to similar representations by the network are mapped to nearby points. We investigate how homologous genes are represented in this space.

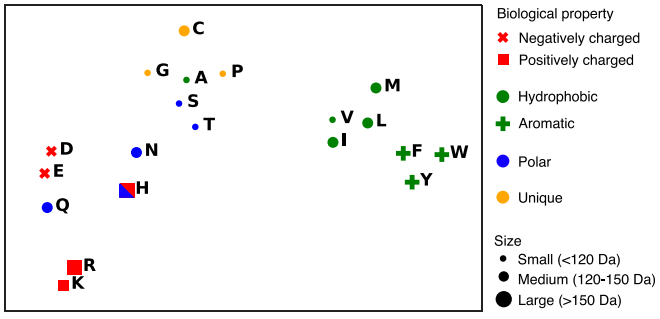

Fig. 1. Biochemical properties of amino acids are represented in the Transformer model's output embeddings, visualized here with t-SNE. Through unsupervised learning, residues are clustered into hydrophobic, polar, and aromatic groups and reflect overall organization by molecular weight and charge. Visualization of 36-layer Transformer trained on UniParc.

The structure and function of orthologous genes are likely to be retained despite divergence of their sequences (43). We find in Fig. 2*A* that training shapes the representation space so that orthologous genes are clustered. Fig. 2*A* shows a two-dimensional projection of the model's representation space using t-SNE. Prior to training, the organization of orthologous proteins in the model's representation space is diffuse. Orthologous genes are clustered in the learned representation space.

We examine whether unsupervised learning encodes biological variations into the structure of the representation space. We apply principal component analysis (PCA) to recover principal directions of variation in the representations, selecting four orthologous genes across four species to look for directions of variation. Fig. 2*B* indicates that linear dimensionality reduction recovers species and orthology as primary axes of variation in the representation space after training. This form of structure is absent from the representations prior to training.

To quantitatively investigate the structure of the representation space, we assess nearest neighbor recovery under vector similarity queries. If biological properties are encoded along independent directions in the representation space, then proteins corresponding with a unique biological variation are related by linear vector arithmetic. In *SI Appendix*, Fig. S1, we find that learning improves recovery of target proteins under queries encoded as linear transformations along the species or gene axes.

Learning Encodes Remote Homology. Remotely homologous proteins have underlying structural similarity despite divergence of their sequences. If structural homology is encoded in the metric structure of the representation space, then the distance between proteins reflects their degree of structural relatedness.

We investigate whether the representation space enables detection of remote homology at the superfamily (proteins that belong to different families but are in the same superfamily) and fold (proteins that belong to different superfamilies but have the same fold) level. We construct a dataset to evaluate remote homology detection using SCOPe (Structural Classification of Proteins—extended) (44). Following standard practices (45), we exclude Rossmann-like folds (c.2 to c.5, c.27 and 28, c.30 and 31) and four- to eight-bladed β-propellers (b.66 to b.70).

An unsupervised classifier on distance from the query measures the density of homologous proteins in the neighborhood of a query sequence. For each domain, a vector similarity query is performed against all other domains, ranking them by distance to the query domain. For evaluation at the fold level, any domain with the same fold is a positive, any domain with a different fold is a negative, and domains belonging to the same superfamily are excluded. For evaluation at the superfamily level, any domain with the same superfamily is a positive, any domain with a different superfamily is a negative, and domains belonging to the same family are excluded. We report the area under the ROC curve (AUC) for the classifier and Hit-10 (46), which gives the probability of recovering a remote homolog in the 10 highest ranked results.

Table 2 indicates that vector nearest neighbor queries using the representations can detect remote homologs that are distant at the fold level with similar performance to HHblits (15), a state-of-the-art hidden Markov model (HMM) based method. At the superfamily level, where sequence similarity is higher, HMM performance is better, but Transformer embeddings are close. Fast vector nearest neighbor finding methods allow billions of sequences to be searched for similarity to a query protein within milliseconds (47).

Learning Encodes Alignment within a Protein Family. An MSA identifies corresponding sites across a family of related sequences (23). These correspondences give a picture of evolutionary variation at different sites within the sequence family. The model receives as input individual sequences and is given no access to the family of related sequences except via learning. We investigate whether the final hidden representations of a sequence encode information about the family it belongs to.

Family information could appear in the network through assignment of similar representations to positions in different sequences that are aligned in the family's MSA. Using the collection of MSAs of structurally related sequences in Pfam (48), we compare the distribution of cosine similarities of representations between pairs of residues that are aligned in the family's MSA to a

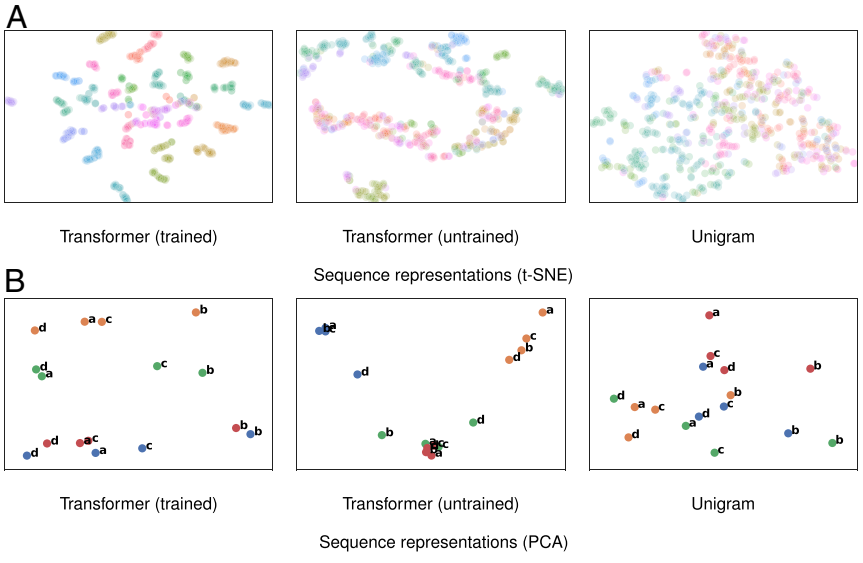

Fig. 2. Protein sequence representations encode and organize biological variations. (*A*) Each point represents a gene, and each gene is colored by the orthologous group it belongs to (dimensionality is reduced by t-SNE). Orthologous groups of genes are densely clustered in the trained representation space. By contrast, the untrained representation space and unigram representations do not reflect strong organization by evolutionary relationships. (*B*) Genes corresponding to a common biological variation are related linearly in the trained representation space. Genes are colored by their orthologous group, and their species are indicated by a character label. PCA recovers a species axis (horizontal) and orthology axis (vertical) in the trained representation space but not in the untrained or unigram spaces. Representations are from the 36-layer Transformer model trained on UniParc.

Table 2. Remote homology detection

		Hit-10		AUC	
	Pretraining	Fold	SF	Fold	SF
HHblits*		0.584	0.965	0.831	0.951
LSTM (S)	UR50/S	0.558	0.760	0.801	0.863
LSTM (L)	UR50/S	0.574	0.813	0.805	0.880
Transformer-6	UR50/S	0.653	0.878	0.768	0.901
Transformer-12	UR50/S	0.639	0.915	0.778	0.942
Transformer-34	(None)	0.481	0.527	0.755	0.807
Transformer-34	UR100	0.599	0.841	0.753	0.876
Transformer-34	UR50/D	0.617	0.932	0.822	0.932
Transformer-34	UR50/S	0.639	0.931	0.825	0.933
ESM-1b	UR50/S	0.532	0.913	0.770	0.880

Structural homology at the fold and superfamily (SF) level is encoded in the metric structure of the representation space. Results for unsupervised classifier based on distance between vector sequence embeddings. Hit-10 reports the probability that a remote homolog is included in the 10 nearest neighbors of the query sequence. AUC is reported for classification by distance from the query in representation space. Transformer models have higher performance than LSTMs and similar performance to HMMs at the fold level.
*HHblits (15) is a state-of-the-art HMM-based method for remote homology detection, using three iterations of sequence search.

background distribution of cosine similarities between unaligned pairs of residues. A large difference between the aligned and unaligned distributions implies that the representations use shared features for related sites within all the sequences of the family.

Fig. 3*A* depicts the distribution of cosine similarity values between aligned and unaligned positions within a representative family for the trained model and baselines. Unsupervised learning produces a marked shift between the distributions of aligned and unaligned pairs. Fig. 3 *B* and *C* indicate that these trends hold under the constraints that the residue pairs (1) share the same amino acid identity or (2) have different amino acid identities.

We estimate differences between the aligned and unaligned distributions across 128 Pfam families using AUC as a metric of discriminative power between aligned and unaligned pairs. *SI Appendix*, Table S1 shows a quantitative improvement in average AUC after unsupervised training, supporting the idea that self-supervision encodes information about the MSA of a sequence into its representation of the sequence.

Prediction of Secondary Structure and Tertiary Contacts

There is reason to believe that unsupervised learning will cause the model's representations to contain structural information. The underlying structure of a protein is a hidden variable that influences the patterns observed in sequence data. For example, local sequence variation depends on secondary structure (16), and tertiary structure introduces higher order dependencies in the choices of amino acids at different sites within a protein (49, 50). While the model cannot observe protein structure directly, it observes patterns in the sequences of its training data that are determined by structure. In principle, the network could compress sequence variations by capturing commonality in structural elements across the data, thereby encoding structural information into the representations.

Linear Projections. We begin by identifying information about protein structure that is linearly encoded within the representations. The use of linear projections ensures that the information originates in the Transformer representations, enabling a direct inspection of the structural content of representations. By comparing representations of the Transformer before and after pretraining, we can identify the information that emerges as a result of the unsupervised learning.

We perform a fivefold cross validation experiment to study generalization of structural information at the family, superfamily, and fold level. For each of the three levels, we construct a dataset of 15,297 protein structures using the SCOPe database. We partition the structures into five parts, splitting by family, superfamily, and fold accordingly. Fivefold cross validation is performed independently for each of the levels of structural holdout.

To detect information about secondary structure, we fit a logistic regression to the hidden representations using the eight-class secondary structure labels. To detect information about tertiary structure, we fit two separate linear projections to the hidden representations of pairs of positions in the sequence, taking their dot product to regress a binary variable indicating whether the positions are in contact in the protein's three-dimensional structure. The neural representations are compared to 1) projections of the sequence profile and 2) unsupervised contacts predicted by the CCMpred implementation (51) of direct coupling analysis. MSAs for the baselines are generated from the UniClust30 (52) database using three iterations of search by HHblits. For secondary structure, we report eight-class accuracies. For contact precision, we report top-L long-range precision, that is, the precision of the L (length of the protein) highest ranked predictions for contacts with sequence separation of at least 24 residues.

Table 3 shows results of the cross validation experiments. Prior to pretraining, minimal information about secondary structure and contacts can be detected. After pretraining, projections

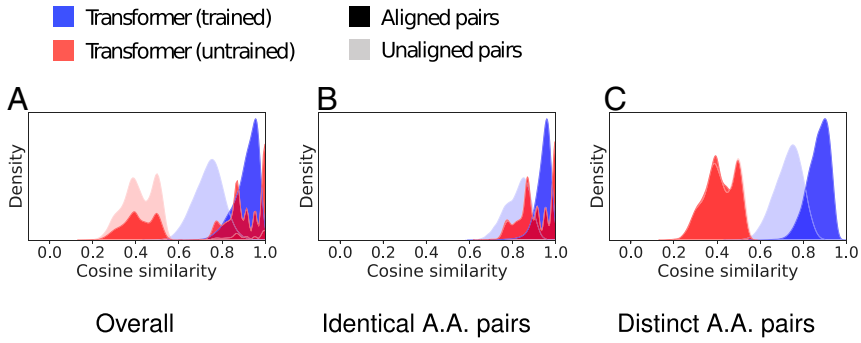

Fig. 3. Final representations from trained models implicitly align sequences. Cosine similarity distributions are depicted for the final representations of residues from sequences within Pfam family PF01010. The differences between the aligned (dark blue) and unaligned (light blue) distributions imply that the trained Transformer representations are a powerful discriminator between aligned and unaligned positions in the sequences. In contrast, representations prior to training do not separate the aligned (dark red) and unaligned positions (light red). (*A*) Overall distribution; distribution under constraint that residue pairs have (*B*) same amino acid identity; or (*C*) different amino acid identities. AUCs across 128 Pfam families are reported in *SI Appendix*, Table S1.

recover information about secondary structure and long-range contacts that generalizes across families, superfamilies, and folds. Secondary structure prediction eight-class accuracy distributions (*SI Appendix*, Fig. S2) and long-range contact prediction top-L precision distributions (*SI Appendix*, Fig. S3) demonstrate that pretraining produces an increase in structural information across the entire distribution of test domains. Table 3 shows that projections of the Transformer representations recover more structure than projections of the sequence profile. For long-range contacts, projections of the best Transformer models have higher precision than contacts predicted by CCMpred across all levels of structural generalization. As the level of structural split becomes more remote, there is little degradation for secondary structure, with performance at the family level similar to the fold level. For long-range contacts, while generalization is reduced at the fold level in comparison to the family level, the best models still capture more structure than the unsupervised baseline. Training with higher diversity sequences (UR50 datasets) improves learning of both secondary structure and long-range contacts, with a more pronounced effect on long-range contacts.

Fig. 4 visualizes three-class secondary structure projections for two domains belonging to held-out folds. Prior to pretraining, projections produce an incoherent prediction of secondary structure. After pretraining, projections recover a coherent prediction with most errors occurring at the boundaries of secondary structure regions. Fig. 5 compares a projected contact map to predictions from CCMpred. Transformer projections recover complex contact patterns, including long-range contacts. Further visualizations of projected contacts for eight randomly selected test proteins are shown in *SI Appendix*, Fig. S7.

Deep Neural Network. We train deep neural networks to predict secondary structure and contacts from the representations. We choose state-of-the-art neural architectures for both tasks. These downstream models are trained with a supervised loss to predict either the secondary structure or contact map from the pretrained representations. The architecture of the downstream model is kept fixed across experiments with different representations and baselines to enable comparison.

To predict the secondary structure, we replace the linear layer with a deep neural network using the model architecture introduced by the NetSurf method (53). For tertiary structure, we predict the binary contact map from the hidden representation of the sequence. We use a dilated convolutional residual network

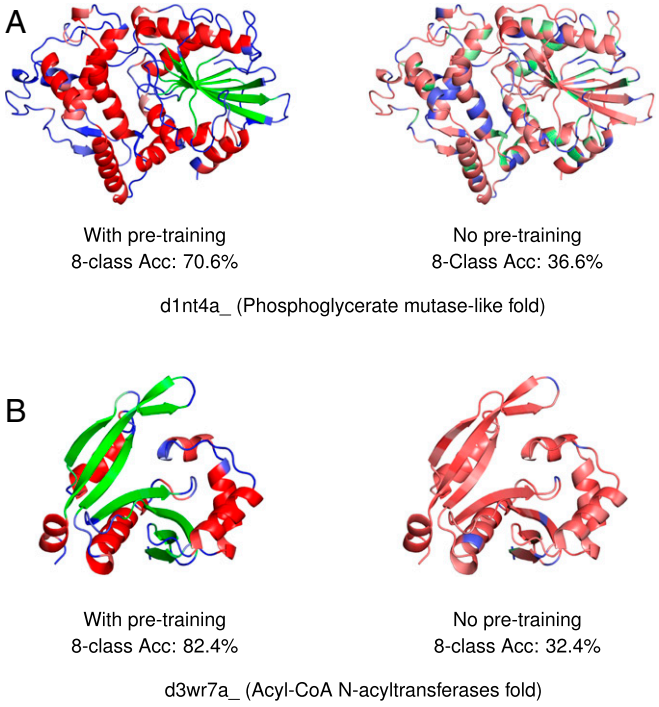

Fig. 4. Secondary structure (linear projections). Example predictions for held-out folds. Unsupervised pretraining encodes secondary structure into representations. Following pretraining, linear projections recover secondary structure (left column). Without pretraining, little information is recovered (right column). (*A*) d1nt4a_ Phosphoglycerate mutase-like fold; (*B*) d3wr7a_ Acyl-CoA N-acyltransferases fold. Colors indicate secondary structure class identified by the projection: helix (red), strand (green), and coil (blue). Color intensities indicate confidence. Representations from ESM-1b Transformer are used.

similar to recent state-of-the-art methods for tertiary structure prediction (54–56).

Table 4 compares the representations for secondary structure prediction. We evaluate models on the CB513 test set (57) and the CASP13 domains (58). For comparison, we also reimplement the NetSurf method. The models are trained on the NetSurf training dataset, which applies a 25% sequence identity holdout

Table 3. Linear projections

		SSP			Contact		
Model	Pretraining	Family	Superfamily	Fold	Family	Superfamily	Fold
HMM Profile	—	47.9 ± 1.1	47.8 ± 1.3	48.0 ± 1.6	14.8 ± 0.6	14.5 ± 1.4	14.6 ± 1.6
CCMpred	—	—	—	—	32.7 ± 1.0	32.5 ± 2.3	32.6 ± 0.4
Transformer-6	UR50/S	62.4 ± 0.7	61.8 ± 0.7	61.8 ± 1.0	24.4 ± 2.2	19.4 ± 3.1	18.6 ± 0.5
Transformer-12	UR50/S	65.5 ± 0.7	65.0 ± 1.0	65.2 ± 1.0	32.8 ± 2.9	25.8 ± 3.7	25.2 ± 0.9
Transformer-34	(None)	41.5 ± 0.9	41.3 ± 0.8	41.3 ± 1.3	8.7 ± 0.3	8.4 ± 0.7	8.4 ± 0.3
Transformer-34	UR100	65.4 ± 0.7	64.9 ± 0.9	64.9 ± 0.9	28.3 ± 2.5	22.5 ± 3.3	22.0 ± 0.8
Transformer-34	UR50/S	69.4 ± 0.6	68.8 ± 0.9	68.9 ± 0.9	43.9 ± 2.8	36.4 ± 4.2	35.3 ± 1.7
Transformer-34	UR50/D	69.5 ± 0.6	68.9 ± 0.9	69.0 ± 0.9	43.9 ± 2.8	37.1 ± 4.6	36.3 ± 2.0
ESM-1b	UR50/S	71.1 ± 0.5	70.5 ± 0.9	70.6 ± 0.9	49.2 ± 2.5	43.5 ± 4.8	42.8 ± 2.3

Fivefold cross validation experiment for generalization at the family, superfamily, and fold level. Eight-class accuracy (secondary structure), top-L long-range precision (contacts), and mean and SD across test sets for the five partitions. Minimal information about structure is present in representations prior to training. Information about secondary and tertiary structure emerges in representations as a result of unsupervised learning on sequences with the language modeling objective. Increasing diversity of sequences improves learning of structure (higher diversity UR50 datasets improve over UR100). Learned representations enable linear projections to generalize to held-out folds, outperforming projections of the sequence profile and contacts identified by the CCMpred (51) implementation of direct coupling analysis.

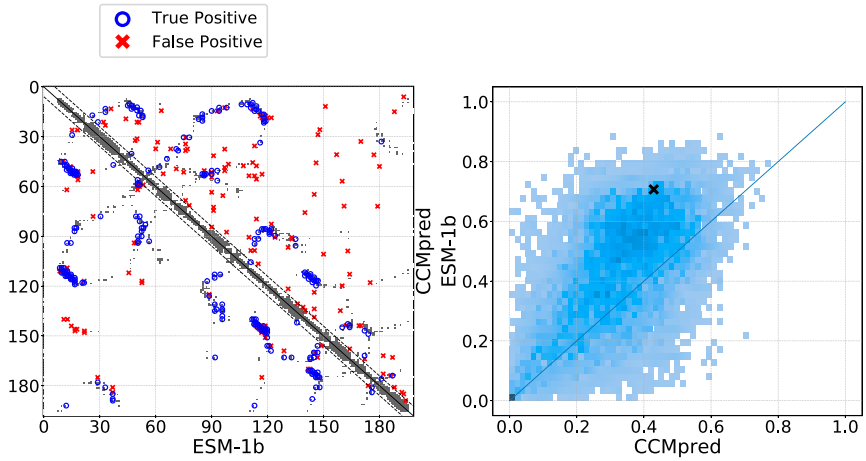

Fig. 5. Residue–residue contacts (linear projections). (*Left*) Top-L predictions for fold level held-out example d1n3ya_, with vWA-like fold. True positives in blue, false positives in red, superimposed on ground truth contact map in gray. ESM-1b Transformer projections below the diagonal, CCMpred predictions above the diagonal. (*Right*) Precision distribution (top-L long-range) comparing ESM-1b projections with CCMpred across all domains in the five test partitions with structural holdout at the fold level. Visualized domain marked by ×.

with CB513 and a temporal holdout with CASP13. The Transformer features are compared before and after unsupervised pretraining to features from the LSTM baselines. They are also compared to the HMM profiles used by NetSurf. The best Transformer features (71.6%) match the performance of the HMM profiles (71.2%) and exceed the published performance of RaptorX (70.6%) on the same benchmark (53), implying that protein language models can produce features that are directly competitive with sequence profiles for secondary structure prediction.

Table 5 shows the performance of the various representations for predicting top-L long-range contacts across a panel of benchmarks using the RaptorX train set (59). For comparison, we train the same architecture using features from RaptorX (54, 59). The Test (59) and CASP11 (60) test sets evaluate with sequence identity holdout at 25%, the CASP12 (61) test set implements a temporal holdout with the structural training data, and the CASP13 (58) experiment implements a full temporal holdout of both the pretraining and training data. For contact prediction, the best features from representation learning do not achieve comparable performance to the state-of-the-art RaptorX features (50.2 versus 59.4, respectively, on the RaptorX test set).

In the secondary structure benchmarks, Transformer representations produce higher accuracy than the LSTM baselines with comparable numbers of parameters. For contact prediction, Transformer representations yield higher precision than LSTMs, with even the smallest Transformer representations exceeding LSTMs with more parameters. Diversity in the pretraining data also has a strong effect, with the high-diversity datasets providing significant improvements over the low-diversity dataset. Relative performance of the representations is consistent across all four of the contact benchmarks using different holdout methodology.

Relationship between Language Modeling and Structure Learning. To investigate the relationship between the language modeling objective and information about structure in the model, linear projections for secondary structure and contacts are fit using the representations from Transformer models taken from checkpoints across their pretraining trajectories. We use the Transformers trained on UR50/S. We fit the projections and evaluate with the train and test split implemented by the first partition of the fold level structural holdout dataset. For each model, Fig. 6

Table 4. Eight-class secondary structure prediction accuracy on the CB513 and CASP13 test sets

Model	Pretraining	CB513	CASP13
HMM Profile		71.2 ± 0.1	72.3 ± 0.9
LSTM (S)	UR50/S	60.4 ± 0.1	63.2 ± 0.6
LSTM (L)	UR50/S	62.4 ± 0.2	64.1 ± 0.7
Transformer-6	UR50/S	62.0 ± 0.2	64.2 ± 1.2
Transformer-12	UR50/S	65.4 ± 0.1	67.2 ± 0.3
Transformer-34	(None)	56.8 ± 0.3	60.0 ± 0.5
Transformer-34	UR100	64.3 ± 0.2	66.5 ± 0.3
Transformer-34	UR50/S	69.1 ± 0.2	70.7 ± 0.8
Transformer-34	UR50/D	69.2 ± 0.1	70.9 ± 0.5
ESM-1B	UR50/S	71.6 ± 0.1	72.5 ± 0.2

A fixed neural architecture is trained to predict the secondary structure label from the language model representation of the input sequence. The Transformer has higher performance than the comparable LSTM baselines. Pretraining with the high-diversity UR50 datasets increases accuracy significantly. Features from ESM-1b Transformer are competitive with HMM profiles for supervised secondary structure prediction. Mean and SD across five random training seeds for the downstream model are shown.

Table 5. Top-L long-range contact precision

			CASP		
Model	Pretraining	Test	11	12	13
LSTM (S)	UR50/S	24.1	23.6	19.9	15.3
LSTM (L)	UR50/S	27.8	26.4	24.0	16.4
Transformer-6	UR50/S	30.2	29.9	25.3	19.8
Transformer-12	UR50/S	37.7	33.6	27.8	20.7
Transformer-34	(None)	16.3	17.7	14.8	13.3
Transformer-34	UR100	32.7	28.9	24.3	19.1
Transformer-34	UR50/S	50.2	42.8	34.7	30.1
Transformer-34	UR50/D	50.0	43.0	33.6	28.0
ESM-1b	UR50/S	56.9	47.4	42.7	35.9

A deep dilated convolutional residual network is trained to predict contacts using the representations from the pretrained language model. The pretrained Transformer representations outperform the LSTM representations in all cases. Pretraining on the high-diversity UR50 datasets boosts precision of representations over pretraining on UR100. High-capacity Transformers (34 layer) outperform lower capacity models (6/12 layer).

shows a linear relationship between the language modeling objective and information about structure, which is maintained over the course of pretraining. The linear fit is close to ideal for both secondary structure and contacts. A similar experiment is also performed for secondary structure with a deep neural network instead of linear projection by using the NetSurf training sequences and CB513 test set. A linear relationship between secondary structure accuracy and language modeling ECE is also observed for the deep prediction head (*SI Appendix*, Fig. S4). Thus, for a given model and pretraining dataset, language modeling fidelity measured by ECE is a good proxy for the structural content of the representations. Since performance on the language modeling objective improves with model capacity, this suggests further scale may improve results on structure prediction tasks.

Single versus Multifamily Pretraining. We compare training across evolutionary statistics to training on single protein families. We pretrain separate 12-layer Transformer models on the Pfam multiple sequence alignments of the three most common domains in nature longer than 100 amino acids: the ATP-binding domain of the ATP-binding cassette transporters, the protein kinase domain, and the response regulator receiver domain. We test the ability of models trained on one protein family to generalize secondary structure information within family and out of family by evaluating on sequences with ground truth labels from the family the model was trained on or from the alternate families. The models are evaluated using linear projections. In all cases, the model trained on within-family sequences has higher accuracy than models trained on out-of-family sequences (*SI Appendix*, Table S2), indicating poor generalization when training on single MSA families. More significantly, the model trained across the full UniParc sequence diversity has a higher accuracy than the single-family model accuracies, even on the same-family evaluation dataset. This suggests that the representations learned from the full dataset are generalizing information about secondary structure learned outside the sequence family.

Feature Combination

Features discovered by unsupervised protein language modeling can be combined with state-of-the-art features to improve them further. Current state-of-the-art methods use information derived from MSAs. We combine this information with features from the Transformer model.

We explore three approaches for incorporating information from representation learning. For each input sequence x, 1) *direct* uses the final hidden representation from the Transformer directly, 2) *avg* takes the average of the final hidden representation at each position across the sequences from the MSA of x, and 3) *cov* produces features for each pair of positions by using the uncentered covariance across sequences from the MSA of x after dimensionality reduction of the final hidden representations by PCA. Note that *direct* and *avg* produce features for each position in x, while *cov* produces features for each pair of positions.

Secondary Structure. Current state-of-the-art methods for secondary structure prediction have high accuracies for the eight-class prediction problem. We investigate whether performance can be improved by combining Transformer features with sequence profiles. Table 6 shows that combining the representations with profiles further boosts accuracy, resulting in state-of-the-art performance on secondary structure prediction.

We establish a baseline of performance by reimplementing the Klausen et al. (53) method using the same features, resulting in an accuracy of 71.2% (versus published performance of 72.1%) on the CB513 test set. Then, we add the Transformer features using the *direct* and *avg* combination methods; these achieve 0.9 and 2.5% absolute improvement in accuracy respectively. This suggests that the Transformer features contain information not present in the MSA-derived features.

Residue–Residue Contacts. Deep neural networks have enabled recent breakthroughs in the prediction of protein contacts and tertiary structure (54, 56). State-of-the-art neural networks for tertiary structure and contact prediction use deep residual architectures with two-dimensional convolutions over pairwise feature maps to output a contact prediction or distance potential for each pair of residues (54, 56, 59).

A variety of input features, training datasets, and supervision signals are used in state-of-the-art methods. To make a controlled comparison, we fix a standard architecture, training dataset, multiple sequence alignments, and set of base input features for all experiments, to which we add pretrained features from the Transformer model. For the base features, we use the RaptorX feature set, which includes position specific scoring matrix

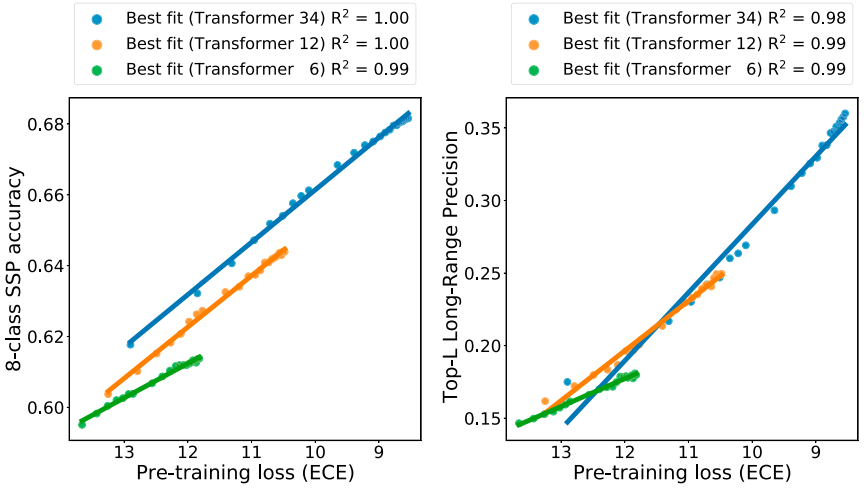

Fig. 6. Relationship between the language modeling objective and structure learning. Eight-class secondary structure prediction accuracy (*Left*) and contact prediction top-L long-range precision (*Right*) both as a function of pretraining ECE. Performance is evaluated on held-out folds. Linear projections are fit using model checkpoints over the course of pretraining on UR50/S. The linear relationship for each model indicates that for a given model and pretraining dataset, the language modeling ECE is a good proxy for the structural content of the representations. Improvement of the model's ECE leads to an increase in information about structure. This establishes a link between the language modeling objective and unsupervised structure learning.

Table 6. Feature combination (secondary structure prediction)

Features	CB513	CASP13
RaptorX	70.6	
NetSurf	72.1	74
(a) NetSurf (reimpl.)	71.2 ± 0.1	72.3 ± 0.9
(b) +direct	72.1 ± 0.1	72.2 ± 0.5
(c) +avg	73.7 ± 0.2	75.1 ± 0.4

Eight-class accuracy. The language model improves state-of-the-art features for secondary structure prediction. Features from a reimplementation of NetSurf (53) are combined with 34-layer Transformer (UR50/S) embeddings using a two-layer bidirectional LSTM architecture. (a) Performance of NetSurf features alone. (b) *Direct* adds the Transformer representation of the input sequence. (c) *Avg* adds the average of Transformer features for each position in the MSA of the input sequence. Results exceed those for state-of-the-art methods RaptorX (70.6%) and NetSurf (72.1%) on the CB513 test set and for NetSurf (74.0%) on the CASP13 evaluation set used here. Mean and SD across five random training seeds for the downstream model are shown.

(PSSM), three-state secondary structure prediction, one-hot embedding of sequence, average product correction-corrected Potts model couplings, mutual information, pairwise contact potential, and predicted accessibility. RaptorX was the winning method for contact prediction in the CASP12 and CASP13 competitions (54). The training and evaluation sets are the same as used in the previous section.

Table 7 indicates that addition of Transformer features from the 34-layer model trained on UR50/S consistently produces an improvement across the test sets. The table shows top-L long-range precisions reporting mean and SD over five different model seeds. *Direct* gives a modest improvement on some test sets. *Avg* improves over direct, and *cov* provides further gains. For example, *cov* produces an absolute improvement of 3.9% on the RaptorX test set and a 1.8% improvement on the CASP13 test set evaluated with temporal holdouts on both fine-tuning and pretraining data. Additional results and metrics for contact prediction are reported in *SI Appendix*, Table S3.

Prediction of Mutational Effects

The mutational fitness landscape provides deep insight into biology. Coupling next-generation sequencing with a mutagenesis screen allows parallel readout of tens of thousands of variants of a single protein (62). The detail and coverage of these experiments provides a view into the mutational fitness landscape of individual proteins, giving quantitative relationships between sequence and protein function. We adapt the Transformer protein language model to predict the quantitative effect of mutations.

First, we investigate intraprotein variant effect prediction, where a limited sampling of mutations is used to predict the effect of unobserved mutations. This setting has utility in protein engineering applications (63). We evaluate the representations on two deep mutational scanning datasets used by recent state-of-the-art methods for variant effect prediction, Envision (64) and DeepSequence (26). Collectively, the data includes over 700,000 variant effect measurements from over 100 large-scale experimental mutagenesis datasets.

Fine-tuning the Transformer yields a mutational effect predictor that is comparable to the results of Envision. Envision (64) relies on protein structural and evolutionary features to generalize. We assess whether the Transformer can achieve similar generalization results without direct access to structural features. The same methodology for partitioning data for training and evaluation is used as in Gray et al. (64) to allow a comparison of the results. We use the 34-layer Transformer trained on UR50/S. Fig. 7 shows the fine-tuned Transformer exceeds the performance of Envision on 10 of the 12 proteins. For each protein, a fraction $p = 0.8$ of the data are used for training, and the remaining data are used for testing. We report mean and SDs for fivefold cross validation in *SI Appendix*, Table S5. Results varying the fraction of data that is used for training are reported in *SI Appendix*, Fig. S5.

We also evaluate using the same fivefold cross validation methodology on the deep mutational scanning experiments assembled for DeepSequence (26). The fine-tuned Transformer model outperforms the fine-tuned LSTM baselines. While not directly comparable, we also include the performance of the original DeepSequence method, which is unsupervised and represents state of the art for this dataset.

Generalization to a New Fitness Landscape. We analyze the Transformer's ability to generalize to the fitness landscape of a new protein. Following the protocol introduced in Envision, we use a leave-one-out analysis: to evaluate performance on a given protein, we train on data from the remaining $n - 1$ proteins and test on the held-out protein. *SI Appendix*, Fig. S6 shows that the Transformer's predictions from raw sequences perform better than Envision on five of the nine tasks.

Related Work

Contemporaneously with the preprint of this work, Rives et al. (65), two related preprints Alley et al. (66) and Heinzinger et al. (67) also proposed protein language modeling, albeit at a smaller scale. These works, along with Rao et al. (68), were evaluated on a variety of downstream tasks. Rives et al. (65) first proposed protein language modeling with Transformers. Alley et al. (66) and Heinzinger et al. (67) train LSTMs on UniRef50. Rao et al. (68) trained a 12-layer Transformer model (38 M parameters) on Pfam (48). The baselines in this paper are comparable to these models. The large Transformer models trained in this paper are considerably larger than in these related works.

We benchmark against related work in Table 8. Heinzinger et al. (67), Alley et al. (66), and Rao et al. (68) evaluate models on differing downstream tasks and test sets. We retrieve the weights for the above models, evaluating them directly in our codebase against the panel of test sets used in this paper for remote homology, secondary structure prediction, and contact prediction, with the same training data and model architectures. This allows a direct comparison between the representations. Table 8 shows that high-capacity Transformers have strong performances for secondary structure and contact predictions significantly exceeding Alley et al. (66), Heinzinger et al. (67), and Rao et al. (68). The small Transformer models trained as baselines also

Table 7. Feature combination (contact prediction)

	Test	CASP11	CASP12	CASP13
No. domains	500	105	55	34
(a) RaptorX	59.4 ± 0.2	53.8 ± 0.3	51.1 ± 0.2	43.4 ± 0.4
(b) +direct	61.7 ± 0.4	55.0 ± 0.1	51.5 ± 0.5	43.7 ± 0.4
(c) +avg	62.9 ± 0.4	56.6 ± 0.4	52.4 ± 0.5	44.8 ± 0.8
(d) +cov	63.3 ± 0.2	56.8 ± 0.2	53.0 ± 0.3	45.2 ± 0.5

Top-L long-range contact precision. The language model improves state-of-the-art features for contact prediction. A deep ResNet with fixed architecture is trained on each feature set to predict binary contacts. (a) Performance of state-of-the-art RaptorX (54) features, including PSSM, predicted secondary structure, predicted accessibility, pairwise average product correction-corrected Potts model couplings and mutual information, and a pairwise contact potential. (b) Adds Transformer representation of the input sequence to the feature set. (c) Adds the average Transformer representation at each position of the MSA. (d) Adds the uncentered covariance over the MSA of a low-dimensional projection of the Transformer features. Features are from the 34-layer Transformer pretrained on UR50/S. Mean and SD across five random training seeds for the downstream model are shown.

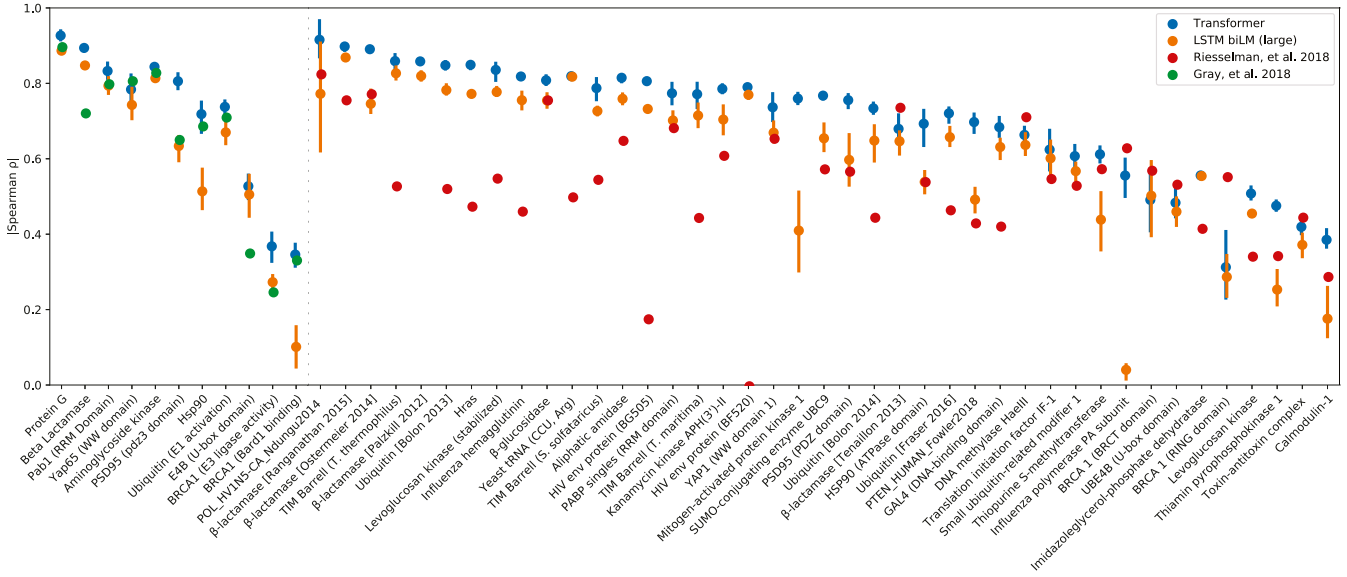

Fig. 7. Representation learning enables state-of-the-art supervised prediction of the quantitative effect of mutations. (*Left*) Envision dataset (65). (*Right*) DeepSequence dataset (26). Transformer representations (34-layer, UR50/S) are compared to the LSTM bidirectional language model (large model, UR50/S). The result of fivefold cross validation is reported for each protein. For each partition, supervised fine-tuning is performed on 80% of the mutational data for the protein, and results are evaluated on the remaining 20%. Transformer representations outperform baseline LSTM representations on both datasets. State-of-the-art methods are also shown for each dataset. Gray et al. (65) is a supervised method using structural, evolutionary, and biochemical features, trained with the same protocol as used for the Transformer. Riesselman et al. (26) is an unsupervised method trained on the MSA of each protein. Mean and SD across the five partitions for Transformer model and LSTM baseline.

have higher performances than the methods with comparable parameter numbers.

Protein sequence embeddings have been the subject of recent investigation for protein engineering (69). Bepler and Berger (70) pretrained LSTMs on protein sequences, adding supervision from contacts to produce embeddings. Subsequent to our preprint, related works have built on its exploration of protein sequence modeling, exploring generative models (71, 72), internal representations of Transformers (73), and applications of representation learning and generative modeling such as classification (74, 75), mutational effect prediction (80), and design of sequences (76–78).

Discussion

One of the goals for artificial intelligence in biology could be the creation of controllable predictive and generative models that can read and generate biology in its native language. Accordingly, research will be necessary into methods that can learn intrinsic biological properties directly from protein sequences, which can be transferred to prediction and generation.

We investigated deep learning across evolution at the scale of the largest protein sequence databases, training contextual language models across 86 billion amino acids from 250 million sequences. The space of representations learned from sequences by high-capacity networks reflects biological structure at multiple levels, including that of amino acids, proteins, and evolutionary homology. Information about secondary and tertiary structure is internalized and represented within the network. Knowledge of intrinsic biological properties emerges without supervision—no learning signal other than sequences is given during pretraining.

We find that networks that have been trained across evolutionary data generalize: information can be extracted from representations by linear projections, deep neural networks, or by adapting the model using supervision. Fine-tuning produces results that match state of the art on variant activity prediction. Predictions are made directly from the sequence, using features that have been automatically learned by the language model rather than selected by domain knowledge.

We find that pretraining discovers information that is not present in current state-of-the-art features. The learned features can be combined with features used by state-of-the-art structure prediction methods to improve results. Empirically, we find that features discovered by larger models perform better on downstream tasks. The Transformer outperforms LSTMs with similar capacity across benchmarks. Increasing diversity of the training data results in significant improvements to the representations.

While the protein language models we study are of comparable scale to those used in the text domain, our experiments have not yet reached the limit of scale. We observed that even the highest capacity models we trained (with ∼650 to 700 M parameters) under-fit the sequence datasets because of insufficient model capacity. The relationship we find between language modeling fidelity and the information about structure encoded into the representations suggests that higher capacity models will yield better representations. These findings imply potential for further

Table 8. Comparison to related protein language models

Model	Pretraining	Params	RH	SSP	Contact
UniRep* (66)		18 M	0.527	58.4	21.9
SeqVec* (67)		93 M	0.545	62.1	29.0
Tape* (68)		38 M	0.581	58.0	23.2
LSTM (S)	UR50/S	28.4 M	0.558	60.4	24.1
LSTM (L)	UR50/S	113.4 M	0.574	62.4	27.8
Transformer-6	UR50/S	42.6 M	0.653	62.0	30.2
Transformer-12	UR50/S	85.1 M	0.639	65.4	37.7
Transformer-34	UR100	669.2 M	0.599	64.3	32.7
Transformer-34	UR50/S	669.2 M	0.639	69.2	50.2
ESM-1b	UR50/S	652.4 M	0.532	71.6	56.9

RH, remote homology at the fold level, using Hit-10 metric on SCOP. SSP, secondary structure Q8 accuracy on CB513. Contact, Top-L long-range contact precision on RaptorX test set from Wang et al. (59). Results for additional test sets are in *SI Appendix*, Table S6.

*The pretraining datasets for related work have differences from ours. See Alley et al. (66), Heinzinger et al. (67), and Rao et al. (68) for details.

model scale and data diversity, incorporating sequences from metagenomics.

Combining high-capacity generative models with gene synthesis and high throughput characterization can enable generative biology. The models we have trained can be used to generate new sequences (79). If neural networks can transfer knowledge learned from protein sequences to design functional proteins, this could be coupled with predictive models to jointly generate and optimize sequences for desired functions. The size of current sequence data and its projected growth point toward the possibility of a general purpose generative model that can condense the totality of sequence statistics, internalizing and integrating fundamental chemical and biological concepts including structure, function, activity, localization, binding, and dynamics, to generate new sequences that have not been seen before in nature but that are biologically active.

Data Availability. Pretrained models and datasets are available at https://github.com/facebookresearch/esm.

ACKNOWLEDGMENTS. We thank Tristan Bepler, Richard Bonneau, Yilun Du, Vladimir Gligorijevic, Anika Gupta, Omer Levy, Ian Peikon, Hetunandan Kamisetty, Laurens van der Maaten, Ethan Perez, Oded Regev, Neville Sanjana, and Emily Wang for feedback on the manuscript and insightful conversations. We thank Jinbo Xu for sharing RaptorX features and help with CASP13. We thank Michael Klausen for providing the NetSurf training code. A.R. was supported at New York University by NSF Grant #1339362.

1. C. Yanofsky, V. Horn, D. Thorpe, Protein structure relationships revealed by mutational analysis. *Science* **146**, 1593–1594 (1964).
2. D. Altschuh, A. M. Lesk, A. C. Bloomer, A. Klug, Correlation of co-ordinated amino acid substitutions with function in viruses related to tobacco mosaic virus. *J. Mol. Biol.* **193**, 693–707 (1987).
3. D. Altschuh, T. Vernet, P. Berti, D. Moras, K. Nagai, Coordinated amino acid changes in homologous protein families. *Protein Eng.* **2**, 193–199 (1988).
4. U. Göbel, C. Sander, R. Schneider, A. Valencia, Correlated mutations and residue contacts in proteins. *Proteins* **18**, 309–317 (1994).
5. Z. S. Harris, Distributional structure. *Word* **10**, 146–162 (1954).
6. J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv [Preprint] (2018). *arXiv*:1810.04805 (Accessed 6 August 2020).
7. R. Collobert, J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning" in *Proceedings of the 25th International Conference on Machine Learning*, W. Cohen, A. McCallum, S. Roweis, Eds. (ACM, New York, NY, 2008), pp. 160–167.
8. A. M. Dai, Q. V. Le, "Semi-supervised sequence learning" in *Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, R. Garnett, Eds. (Curran Associates, Inc., Red Hook, NY, 2015), pp. 3079–3087.
9. M. E. Peters *et al*., "Deep contextualized word representations" in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, M. Walker, J. Heng, A. Stent, Eds. (ACL, Stroudsburg, PA, 2018), pp. 2227–2237.
10. A. Baevski, S. Edunov, Y. Liu, L. Zettlemoyer, M. Auli, Cloze-driven pretraining of self-attention networks. arXiv [Preprint] (2019). *arXiv*:1903.07785 (Accessed 6 August 2020).
11. A. Radford *et al*., Language models are unsupervised multitask learners. OpenAI Blog [Preprint] (2019). https://openai.com/blog/better-language-models (Accessed 6 August 2020).
12. S. F. Altschul, W. Gish, W. Miller, E. W. Myers, D. J. Lipman, Basic local alignment search tool. *J. Mol. Biol.* **215**, 403–410 (1990).
13. S. F. Altschul, E. V. Koonin, Iterated profile searches with PSI-BLAST–A tool for discovery in protein databases. *Trends Biochem. Sci.* **23**, 444–447 (1998).
14. S. R. Eddy, Profile hidden Markov models. *Bioinformatics* **14**, 755–763 (1998).
15. M. Remmert, A. Biegert, A. Hauser, J. Söding, HHblits: Lightning-fast iterative protein sequence searching by HMM-HMM alignment. *Nat. Methods* **9**, 173–175 (2011).
16. M. Levitt, Conformational preferences of amino acids in globular proteins. *Biochemistry* **17**, 4277–4285 (1978).
17. A. S. Lapedes, B. G. Giraud, L. Liu, G. D. Stormo, Correlated mutations in models of protein sequences: Phylogenetic and structural effects. *Lecture Notes-Monograph Series*, 236–256 (1999).
18. J. Thomas, N. Ramakrishnan, C. Bailey-Kellogg, Graphical models of residue coupling in protein families. *IEEE/ACM Trans. Comput. Biol. Bioinformatics* **5**, 183–197 (2008).
19. M. Weigt, R. A. White, H. Szurmant, J. A. Hoch, T. Hwa, Identification of direct residue contacts in protein-protein interaction by message passing. *Proc. Natl. Acad. Sci. U.S.A.* **106**, 67–72 (2009).
20. F. Morcos *et al*., Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proc. Natl. Acad. Sci. U.S.A.* **108**, E1293–E1301 (2011).
21. D. T. Jones, D. W. Buchan, D. Cozzetto, M. Pontil, PSICOV: Precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments. *Bioinformatics* **28**, 184–190 (2012).
22. S. Balakrishnan, H. Kamisetty, J. G. Carbonell, S.-I. Lee, C. J. Langmead, Learning generative models for protein fold families. *Proteins* **79**, 1061–1078 (2011).
23. M. Ekeberg, C. Lövkvist, Y. Lan, M. Weigt, E. Aurell, Improved contact prediction in proteins: Using pseudolikelihoods to infer Potts models. *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.* **87**, 012707 (2013).
24. M. Figliuzzi, H. Jacquier, A. Schug, O. Tenaillon, M. Weigt, Coevolutionary landscape inference and the context-dependence of mutations in beta-lactamase TEM-1. *Mol. Biol. Evol.* **33**, 268–280 (2016).
25. T. A. Hopf *et al*., Mutation effects predicted from sequence co-variation. *Nat. Biotechnol.* **35**, 128–135 (2017).
26. A. J. Riesselman, J. B. Ingraham, D. S. Marks, Deep generative models of genetic variation capture the effects of mutations. *Nat. Methods* **15**, 816–822 (2018).
27. Y. Bengio, R. Ducharme, P. Vincent, C. Jauvin, A neural probabilistic language model. *J. Mach. Learn. Res.* **3**, 1137–1155 (2003).
28. A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, Improving language understanding by generative pre-training. OpenAI Blog [Preprint] (2018). https://openai.com/blog/language-unsupervised (Accessed 6 August 2020).
29. T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space. arXiv [Preprint] (2013). https://arxiv.org/abs/1301.3781 (Accessed 6 August 2020).
30. T. Mikolov *et al*., Subword language modeling with neural networks. The website of T. Mikolov [Preprint] (2012). http://www.fit.vutbr.cz/~imikolov/rnnlm/char.pdf (Accessed 14 March 2021).
31. Y. Kim, Y. Jernite, D. Sontag, A. M. Rush, "Character-aware neural language models" in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016*, D. Schuurmans, M. Wellman, Eds. (AAAI Press, Palo Alto, CA, 2016), pp. 2741–2749.
32. A. Vaswani *et al*., "Attention is all you need" in *Advances in Neural Information Processing Systems*, I. Guyon, Ed. *et al*. (Curran Associates, Inc., Red Hook, NY, 2017), pp. 5998–6008.
33. UniProt Consortium, The universal protein resource (UniProt). *Nucleic Acids Res.* **36**, D190–D195 (2008).
34. B. E. Suzek, Y. Wang, H. Huang, P. B. McGarvey, C. H. Wu; UniProt Consortium, UniRef clusters: A comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics* **31**, 926–932 (2015).
35. B. E. Suzek, H. Huang, P. McGarvey, R. Mazumder, C. H. Wu, UniRef: Comprehensive and non-redundant UniProt reference clusters. *Bioinformatics* **23**, 1282–1288 (2007).
36. S. Hormoz, Amino acid composition of proteins reduces deleterious impact of mutations. *Sci. Rep.* **3**, 2919 (2013).
37. T. Gabaldón, Evolution of proteins and proteomes: A phylogenetics approach. *Evol. Bioinform. Online* **1**, 51–61 (2007).
38. S.-W. Wang, A.-F. Bitbol, N. S. Wingreen, Revealing evolutionary constraints on proteins through sequence analysis. *PLoS Comput. Biol.* **15**, e1007010 (2019).
39. J. Overbaugh, C. R. Bangham, Selection forces and constraints on retroviral sequence variation. *Science* **292**, 1106–1109 (2001).
40. S. Sunyaev, F. A. Kondrashov, P. Bork, V. Ramensky, Impact of selection, mutation rate and genetic drift on human genetic variation. *Hum. Mol. Genet.* **12**, 3325–3330 (2003).
41. K. Jarrett, K. Kavukcuoglu, M. Ranzato, Y. LeCun, "What is the best multi-stage architecture for object recognition?" in *2009 IEEE 12th International Conference on Computer Vision* (Curran Associates, Inc., Red Hook, NY, 2009), pp. 2146–2153.
42. L. van der Maaten, G. Hinton, Visualizing data using t-SNE. *J. Mach. Learn. Res.* **9**, 2579–2605 (2008).
43. J. Huerta-Cepas *et al*., eggNOG 5.0: A hierarchical, functionally and phylogenetically annotated orthology resource based on 5090 organisms and 2502 viruses. *Nucleic Acids Res.* **47**, D309–D314 (2019).
44. N. K. Fox, S. E. Brenner, J.-M. Chandonia, SCOPe: Structural Classification of Proteins–extended, integrating SCOP and ASTRAL data and classification of new structures. *Nucleic Acids Res.* **42**, D304–D309 (2014).
45. J. Söding, M. Remmert, Protein sequence comparison and fold recognition: Progress and good-practice benchmarking. *Curr. Opin. Struct. Biol.* **21**, 404–411 (2011).
46. J. Ma, S. Wang, Z. Wang, J. Xu, MRFalign: Protein homology detection through alignment of Markov random fields. *PLOS Comp. Biol.* **10**, e1003500 (2014).
47. J. Johnson, M. Douze, H. Jégou, Billion-scale similarity search with GPUs. arXiv [Preprint] (2017). *arXiv*:1702.08734 (Accessed 6 August 2020).
48. R. D. Finn *et al*., Pfam: The protein families database. *Nucleic Acids Res.* **42**, D222–D230 (2014).
49. D. S. Marks *et al*., Protein 3D structure computed from evolutionary sequence variation. *PLoS One* **6**, e28766 (2011).
50. I. Anishchenko, S. Ovchinnikov, H. Kamisetty, D. Baker, Origins of coevolution between residues distant in protein 3D structures. *Proc. Natl. Acad. Sci. U.S.A.* **114**, 9122–9127 (2017).
51. S. Seemayer, M. Gruber, J. Söding, CCMpred–Fast and precise prediction of protein residue-residue contacts from correlated mutations. *Bioinformatics* **30**, 3128–3130 (2014).
52. M. Mirdita *et al*., Uniclust databases of clustered and deeply annotated protein sequences and alignments. *Nucleic Acids Res.* **45**, D170–D176 (2017).

53. M. S. Klausen *et al*., NetSurfP-2.0: Improved prediction of protein structural features by integrated deep learning. *Proteins* **87**, 520–527 (2019).
54. J. Xu, Distance-based protein folding powered by deep learning. arXiv [Preprint] (2018). *arXiv*:1811.03481 (Accessed 6 August 2020).
55. D. T. Jones, S. M. Kandathil, High precision in protein contact prediction using fully convolutional neural networks and minimal sequence features. *Bioinformatics* **34**, 3308–3315 (2018).
56. A. W. Senior *et al*., Improved protein structure prediction using potentials from deep learning. *Nature* **577**, 706–710 (2020).
57. J. A. Cuff, G. J. Barton, Evaluation and improvement of multiple sequence methods for protein secondary structure prediction. *Proteins* **34**, 508–519 (1999).
58. A. Kryshtafovych, T. Schwede, M. Topf, K. Fidelis, J. Moult, Critical assessment of methods of protein structure prediction (CASP)-Round XIII. *Proteins* **87**, 1011–1020 (2019).
59. S. Wang, S. Sun, Z. Li, R. Zhang, J. Xu, Accurate de novo prediction of protein contact map by ultra-deep learning model. *PLoS Comput. Biol.* **13**, e1005324 (2017).
60. J. Moult, K. Fidelis, A. Kryshtafovych, T. Schwede, A. Tramontano, Critical assessment of methods of protein structure prediction: Progress and new directions in round XI. *Proteins* **84** (suppl. 1), 4–14 (2016).
61. J. Moult, K. Fidelis, A. Kryshtafovych, T. Schwede, A. Tramontano, Critical assessment of methods of protein structure prediction (CASP)-Round XII. *Proteins* **86** (suppl. 1), 7–15 (2018).
62. D. M. Fowler, S. Fields, Deep mutational scanning: A new style of protein science. *Nat. Methods* **11**, 801–807 (2014).
63. K. K. Yang, Z. Wu, F. H. Arnold, Machine-learning-guided directed evolution for protein engineering. *Nat. Methods* **16**, 687–694 (2019).
64. V. E. Gray, R. J. Hause, J. Luebeck, J. Shendure, D. M. Fowler, Quantitative missense variant effect prediction using large-scale mutagenesis data. *Cell Syst.* **6**, 116–124.e3 (2018).
65. A. Rives *et al*., Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *bioRxiv* [Preprint] (2019). https://doi.org/10.1101/622803 (Accessed 6 August 2020).
66. E. C. Alley, G. Khimulya, S. Biswas, M. AlQuraishi, G. M. Church, Unified rational protein engineering with sequence-based deep representation learning. *Nat. Methods* **16**, 1315–1322 (2019).
67. M. Heinzinger *et al*., Modeling aspects of the language of life through transfer-learning protein sequences. *BMC Bioinformatics* **20**, 723 (2019).
68. R. Rao *et al*., "Evaluating protein transfer learning with TAPE" in *Advances in Neural Information Processing Systems*, H. Wallach, Ed. *et al*. (Curran Associates, Inc., Red Hook, NY, 2019), pp. 9686–9698.
69. K. K. Yang, Z. Wu, C. N. Bedbrook, F. H. Arnold, Learned protein embeddings for machine learning. *Bioinformatics* **34**, 2642–2648 (2018).
70. T. Bepler, B. Berger, "Learning protein sequence embeddings using information from structure" in International Conference on Learning Representations, (OpenReview.net, 2019).
71. A. J. Riesselman *et al*., Accelerating protein design using autoregressive generative models. *bioRxiv* [Preprint] (2019). https://doi.org/10.1101/757252 (Accessed 6 August 2020).
72. A. Madani *et al*., ProGen: Language modeling for protein generation. arXiv [Preprint] (2020). *arXiv*:2004.03497 (Accessed 6 August 2020).
73. J. Vig *et al*., BERTology meets biology: Interpreting attention in protein language models. arXiv [Preprint] (2020). *arXiv*:2006.15222 (Accessed 6 August 2020).
74. A. Elnaggar, M. Heinzinger, C. Dallago, B. Rost, End-to-end multitask learning, from protein language to protein features without alignments. *bioRxiv* [Preprint] (2019). https://doi.org/10.1101/864405 (Accessed 6 August 2020).
75. N. Strodthoff, P. Wagner, M. Wenzel, W. Samek, UDSMProt: Universal deep sequence models for protein classification. *Bioinformatics* **36**, 2401–2409 (2020).
76. D. Repecka *et al*., Expanding functional protein sequence space using generative adversarial networks. *bioRxiv* [Preprint] (2019). https://doi.org/10.1101/789719 (Accessed 6 August 2020).
77. A. Hawkins-Hooker *et al*., Generating functional protein variants with variational autoencoders. *bioRxiv* [Preprint] (2019). https://doi.org/10.1101/2020.04.07.029264 (Accessed 6 August 2020).
78. T. Amimeur *et al*., Designing feature-controlled humanoid antibody discovery libraries using generative adversarial networks. *bioRxiv* [Preprint] (2019). https://doi.org/10.1101/2020.04.12.024844 (Accessed 6 August 2020).
79. A. Wang, K. Cho, BERT has a mouth, and it must speak: BERT as a markov random field language model. arXiv [Preprint] (2019). *arXiv*:1902.04094 (Accessed 6 August 2020).
80. Y. Luo *et al*., Evolutionary context-integrated deep sequence modeling for protein engineering. *bioRxiv* [Preprint] (2020). https://doi.org/10.1101/2020.01.16.908509 (Accessed 6 August 2020).