Language models of protein sequences at the scale of evolution enable accurate structure prediction

Zeming Lin[1,2,*], Halil Akin[1,*], Roshan Rao[1,*], Brian Hie[1,3,*], Zhongkai Zhu[1], Wenting Lu[1], Allan dos Santos Costa[4], Maryam Fazel-Zarandi[1], Tom Sercu[1,†], Sal Candido[1,†], Alexander Rives[1,†,‡]

[1] Meta AI, FAIR Team
[2] New York University. Work performed as a visiting researcher at Meta AI.
[3] Stanford University. Work performed as a visiting researcher at Meta AI.
[4] Massachusetts Institute of Technology. Work performed during internship at Meta AI.

* Equal contribution
† Research and engineering leadership
‡ Corresponding author, arives@fb.com

Abstract

Large language models have recently been shown to develop emergent capabilities with scale, going beyond simple pattern matching to perform higher level reasoning and generate lifelike images and text. While language models trained on protein sequences have been studied at a smaller scale, little is known about what they learn about biology as they are scaled up. In this work we train models up to 15 billion parameters, the largest language models of proteins to be evaluated to date. We find that as models are scaled they learn information enabling the prediction of the three-dimensional structure of a protein at the resolution of individual atoms. We present ESMFold for high accuracy end-to-end atomic level structure prediction directly from the individual sequence of a protein. ESMFold has similar accuracy to AlphaFold2 and RoseTTAFold for sequences with low perplexity that are well understood by the language model. ESMFold inference is an order of magnitude faster than AlphaFold2, enabling exploration of the structural space of metagenomic proteins in practical timescales.

Introduction

In linguistics, the distributional hypothesis proposes that meaning can be inferred from text by the way it constrains the patterns of words (*1*). An analogous idea has been critical for inference from sequences in biology. Because the structure and function of a protein constrains the mutations to its sequence that are selected through evolution (*2–4*), it should also be possible to infer biological structure and function from sequence patterns (*5–9*), which would provide insight into some of the most foundational problems in biology (*10*). However, learning sufficient information from sequence alone to model the complexity and diversity of biological structures and functions remains a considerable challenge.

In natural language processing and artificial intelligence, general purpose language models have shown that their performance on complex tasks improves as compute, data, and model size increases. At certain scales, language models exhibit useful capabilities which emerge as a result of scaling a simple training process to large corpuses of data, e.g. few-shot language translation, commonsense reasoning, and mathematical reasoning (*11–14*). To this end, we study protein structure as learned by language models

trained purely on protein sequence data with a simple language modeling objective (*15–22*). Previous work has shown that protein language models can capture some functional (*23*) and structural properties of proteins, including secondary structure, tertiary contacts, backbone structure, and antibody structure (*15*, *19*, *24*, *25*).

Here, we report that large protein language models learn sufficient information to enable accurate, atomic-level predictions of protein structure. First, we introduce ESM-2, in variants up to 15 billion parameters, the largest language model of protein sequences to date. Next, we introduce ESMFold, which uses the information and representations learned by ESM-2 to perform end-to-end 3D structure prediction using only a single sequence as input, allowing us to quantify the emergence of protein structure as the language model is scaled from millions to billions of parameters. Notably, we find that as the size of the language model increases, we also observe consistent improvements in structure prediction accuracy.

While recent models, AlphaFold2 (*26*) and RoseTTAFold (*27*), have achieved breakthrough success in the problem of atomic-resolution structure prediction, they also rely on the use of multiple sequence alignments (MSAs) and templates of similar protein structures to achieve optimal performance. In contrast, by leveraging the internal representations of the language model, ESMFold generates structure predictions using only a single sequence as input, resulting in considerably faster structure prediction. ESMFold also produces more accurate atomic-level predictions than AlphaFold2 or RoseTTAFold when they are artificially given a single sequence as input, and obtains competitive performance to RoseTTAFold given full MSAs as input. Moreover, we find that ESMFold produces comparable predictions to state-of-the-art models for low perplexity sequences, and more generally that structure prediction accuracy correlates with language-model perplexity, indicating that when a language model better understands the sequence, it also better understands the structure.

Since ESMFold's prediction speed is an order of magnitude faster than existing atomic resolution structure predictors, ESMFold can help address the gap between the rapid growth of protein sequence databases, which increasingly contain billions of sequences (*28–30*), and the much slower growth of databases of protein structures and functions. Using ESMFold, we rapidly compute 1 million predicted structures representing a diverse subset of metagenomic sequence space, most of which have no annotated structure or function (*29*). A large fraction of ESMFold's high-confidence predictions has low similarity to any known experimental structures, which suggests the structural novelty of many metagenomic proteins. Notably, many high-confidence structures also have low sequence similarity to any entry in UniRef90, indicating generalization of the model's predictions beyond its training dataset and enabling structure-based insight into protein function that would be difficult to obtain from sequence information alone. By leveraging the unprecedented view into the language of protein sequences provided by ESM-2, ESMFold promises to augment our understanding of vast databases of poorly understood protein sequences.

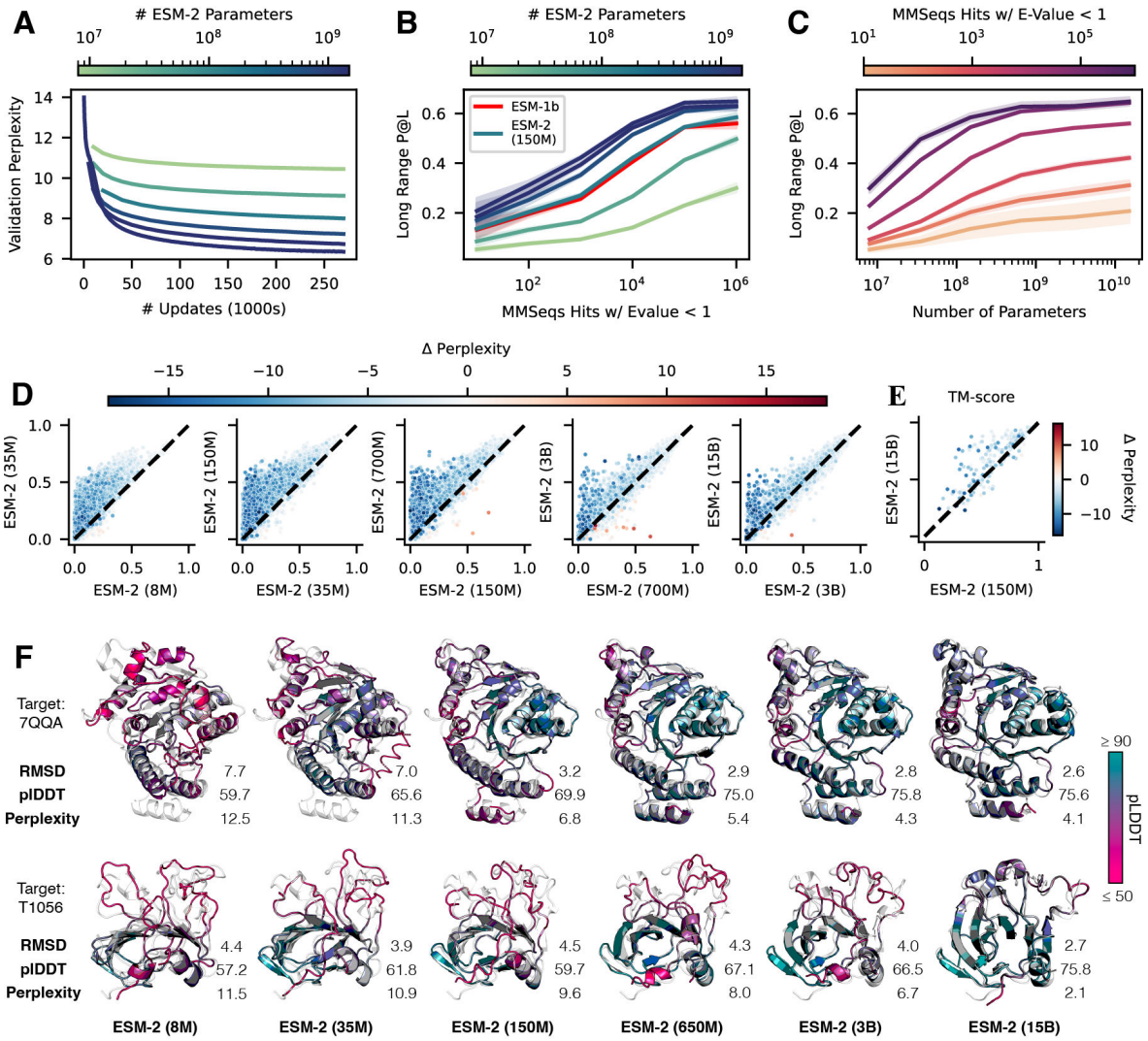

Figure 1: Emergence of structure when scaling language models to 15 Billion parameters. **(A)** Training curves for ESM-2 models at all scales, through 270,000 updates. **(B, C, D)** Unsupervised contact prediction performance (long range precision @ L) for different scales of the ESM-2 model. **(B)** Performance is binned by the number of MMseqs hits when searching the training set. Larger models perform better at all levels, and the 150M parameter ESM-2 model performs comparably with the 650M parameter ESM-1b model. **(C)** Trajectory of improvement as model scale increases for sequences with different numbers of MMseqs hits is shown. The largest improvement is seen for sequences with O(10^4) MMseqs hits. **(D)** Left-to-right shows models from 8M to 15B parameters, consecutively comparing the smaller model (x-axis) against the next larger model (y-axis) in terms of unsupervised contact precision. Points correspond to PDB proteins and are colored by the change in pseudo-perplexity for the sequence between the smaller and larger model. Sequences with large changes in contact prediction performance also exhibit large changes in language model understanding measured by pseudo-perplexity. **(E)** TM-score on combined CASP14 and CAMEO test sets. Models are structure module trained on 150M parameter (x-axis) and 15B parameter (y-axis) ESM-2. Points are colored by the change in pseudo-perplexity between the models. **(F)** Left-to-right structure predictions on CAMEO structure 7QQA and CASP target 1056 at all ESM-2 model scales, colored by pLDDT (pink = low, teal = high). For 7QQA, prediction accuracy improves suddenly at the 150M parameter threshold, and slowly thereafter. For T1056, prediction accuracy improves suddenly at the 15B parameter threshold.

Training and evaluating 15B parameter protein language models.

The ESM-2 language models are the most performant language models of proteins developed to date. Relative to our previous generation model ESM-1b we improve model architecture, training parameters, and increase computational resources and data. Addition of relative positional embeddings enables generalization to arbitrary length sequences. These modifications lead to a significantly better model. We observe that the ESM-2 model with 150M parameters performs better than the ESM-1b model with 650M parameters. On structure prediction benchmarks it also outperforms other recent protein language models (**Table 1, table S3**). This performance increase is consistent with scaling laws established in the large language modeling field (*31*). For context, the 15B parameter ESM-2 model is only one order of magnitude smaller than the largest state-of-the-art language models of text that have been trained such as Chinchilla (70 billion parameters), GPT3 and OPT-175B (both 175 billion parameters), and PALM (540 billion parameters) (*11*, *14*, *32*, *33*).

ESM-2 is trained on protein sequences from the UniRef database (*28*). Given an input protein (represented as a character sequence of amino acids), 15% of amino acids are masked and ESM-2 is tasked with predicting these missing positions (*34*). Although this training objective only directly involves predicting missing amino acids, achieving a high degree of success requires the model to learn complex internal representations of its input. In natural language processing, these representations contain information about parts of speech, dependency parsing, semantic relatedness and textual entailment (*34–36*). In biology, these representations learn secondary structure prediction, binding site prediction, and contact prediction (*15*, *37*, *38*).

As we increase the scale of ESM-2, we observe large improvements in the fidelity of language modeling. Language model performance is evaluated using perplexity, which measures the model's performance on the task of predicting amino acids from their context in a sequence. Perplexity ranges from 1 for a perfect model to 20 for a model that makes predictions at random. Intuitively, perplexity describes the number of amino acids the model is uncertain between when it makes a prediction. We hold out ~500k UniRef50 clusters from the ESM-2 training set for evaluation. **Fig. 1A** shows perplexity as a function of the number of updates for the ESM-2 family models. After 270k training steps the 8M parameter model has a perplexity of 10.45, and the 15B model reaches a perplexity of 6.37.

Next, we look at the emergence of protein structure as the model scales. ESM-2 is a transformer-based language model, and uses an attention mechanism to learn interaction patterns between pairs of amino acids in the input sequence. Standard methods in computational biology use correlations between mutations in amino acids at different sites to learn protein contact maps, since amino acids that are in contact in the three dimensional structure are not free to evolve independently (*5–8*, *39*). More recent work has shown the pairwise interaction patterns learned by protein language models also correspond to protein contact maps because the correlations between different sites is useful for predicting missing amino acids (*37*, *38*). Importantly, this correspondence emerges solely from the language modeling objective, without direct supervision on protein structures.

Throughout **Fig. 1** we compare different ESM-2 parameter scales at 270,000 training updates, and find that scaling leads to large improvements in the unsupervised learning of structure (**Fig. 1B**). We also look at the accuracy of the predicted contacts as a function of the number of evolutionarily related sequences in the language model's training set. Proteins with more related sequences in the training set have steeper

learning trajectories with respect to model scale (**Fig. 1C**). In other words, improvement on sequences with high evolutionary depth saturates at lower model scales, and improvement on sequences with low evolutionary depth continues as models increase in size.

For individual proteins, we often observe non-linear improvements as a function of scale. **Fig. 1D** plots the distribution of long range contact precision as we scale from one model to the next. At each level of scale we see the overall distribution shift toward better performance. Also at each transition, there is a subset of proteins that undergo significant improvement. In **Fig. 1D** these are concentrated in the upper left of each plot, far from the diagonal. There is a link between contact accuracy and perplexity, with proteins undergoing large changes in contact accuracy also undergoing large changes in perplexity. This link indicates that the language modeling objective is directly correlated with the materialization of the structure in the attention maps.

Given that the folded structure, in the form of the contact pattern, develops in the attention patterns of the model with scale, we investigate whether information also develops that enables prediction of the structure at the full atom resolution. We develop a method to project the atom level structure from the internal representations of ESM-2 using the equivariant structure module of Alphafold. Using the same network architecture of the structure module alongside different versions of ESM-2 during training, we can quantitatively compare the language models on how much information their representations contain that is useful for making a full atom prediction. To train the full atom projection we use supervision from a dataset of experimentally determined protein structure from PDB (*40*). We evaluate on temporally held out CAMEO (*41*) and CASP14 (*42*) test sets consisting respectively of 194 and 51 proteins.

Atomic resolution predictions can be extracted from the representations of the ESM-2 language models and improve with scale. The 15 billion parameter ESM-2 model achieves a TM-score (*43*) of 71.3 on the CAMEO test set and 53.9 on the CASP14 test set, 6.4 points higher than the 150 million parameter ESM-2 model on both (**Fig. 1E**). Similarly to the results with contact maps, at each increase in scale a subset of proteins see large changes in accuracy. For example, the protein 7QQA sees an improvement in RMSD from 7.0 to 3.2 when scale is increased from 35M to 150M parameters, and the CASP target T1056 sees an improvement in RMSD from 4.0 to 2.6 when scale is increased from 3B to 15B parameters (**Fig. 1F**). Before and after these jumps, changes in RMSD are much smaller. Across all models (**table S3**) there is a correlation of -0.99 between validation perplexity and CASP14 TM-score, and -1.00 between validation perplexity and CAMEO TM-score indicating a strong link between language model understanding of a sequence measured by perplexity and the atomic resolution structure prediction. Additionally there are strong correlations between the low resolution picture of the structure that can be extracted from the attention maps and the atomic resolution prediction (0.96 between long range contact precision and CASP14 TM-score, and 0.99 between long range contact precision and CAMEO TM-score).

Model	# Params	Validation Perplexity	LR P@L	CASP14	CAMEO
ESM-2	8M	10.33	0.17	0.37	0.48
	35M	8.95	0.30	0.41	0.56
	150M	7.75	0.44	0.49	0.65
	650M	6.95	0.52	0.51	0.70
	3B	6.49	**0.54**	0.52	**0.72**
	15B	**6.37**	**0.54**	**0.55**	**0.72**
ESM-1b[1]	650M	–	0.41	0.42	0.64
Prot-T5-XL-UR50 (*19*)	3B	–	0.48	0.50	0.69
Prot-T5-XL-BFD (*19*)	3B	–	0.36	0.46	0.63
CARP (*44*)	640M	–	–	0.42	0.59

Table 1: Evaluation Metrics for converged ESM-2 models compared with baselines. Comparisons of the final structure predictions using models trained out to 500k updates (except the 15B parameter model which is trained to 270k updates). All numbers are reported with only a structure module trained on top of various language models. Despite the shorter training time, the 15B parameter ESM-2 model has lowest validation perplexity and highest TM-score on CASP14. The 150M parameter ESM-2 model outperforms the 650M parameter ESM-1b model on structure-based tasks, while the 650M parameter ESM-2 model is comparable with the 3B parameter Prot-T5-XL-UniRef50 model, suggesting ESM-2 models are far more parameter efficient.

[1] ESM-1b evaluated only on sequences of length < 1024, due to constraints with position embedding.

End-to-end single-sequence structure prediction with ESMFold

Predicting protein structure from its amino acid sequence is a long-standing grand challenge in the natural sciences (*10*). AlphaFold2 (*26*), arguably the most successful evolutionary-based approach to the problem, presented a breakthrough achievement by training an end-to-end neural network on inputs of sequence, aligned sequences of evolutionary homologs, and optional structural templates. These advances built on earlier work learning using deep learning on sets of aligned sequences to prediction structure (*45*, *46*). Here, we use the information in the large protein language model to train an atomic-resolution structure prediction model that requires only a single sequence input, which we refer to as ESMFold.

A key difference between ESMFold and AlphaFold2 is the use of language model representations to remove the need for explicit homologous sequences (in the form of an MSA) as input. Language model representations are provided as input to ESMFold's folding trunk (**Fig. 2A**), which simplifies the Evoformer in AlphaFold2 by replacing the computationally expensive network modules that process the MSA with a transformer module which processes a sequence (*47*). This simplification means that ESMFold is substantially faster than the MSA-based model. The output of the folding trunk is in turn processed by a structure module, which outputs the final atomic-level structure and predicted confidences (**Fig. 2A**). Additional architectural details can be found in **Methods**. We train ESMFold on a diverse subset of PDB chains, further augmented with a dataset of 12M structures predicted by AlphaFold2 (*48*).

We compare ESMFold to AlphaFold2 and RoseTTAFold on held out CAMEO (April 2022 to June 2022) and CASP14 test sets consisting of structures released after our training data cutoff date (May 2020). As an ablation, we also remove MSA and template information from AlphaFold2 and RoseTTAFold and compare against "single-sequence" versions of these methods. ESMFold achieves an average TM-score of 82.8 on CAMEO and 67.8 on CASP14, significantly higher than single-sequence versions of AlphaFold2 and RoseTTAFold (**Fig. 2, B** and **C**). With the full pipeline, including MSAs and templates, AlphaFold2 achieves 88.3 and 84.7 on CAMEO and CASP14 respectively. ESMFold achieves competitive accuracy with RoseTTAfold on CAMEO, which averages a TM-score of 82.0.

Because the language model is a critical component of ESMFold, we test how well differences in the language model correspond to changes in structure prediction performance. In particular, the performance of ESMFold on both test sets is well correlated with the perplexity of the language model. On the CAMEO test set, language model perplexity has a Pearson correlation of -0.55 with the TM-score between the predicted and experimental structures; on CASP14, the correlation is -0.67 (**Fig. 2, B** and **C**). The relationship between perplexity and structure prediction suggests that improving the language model is key to improving single-sequence structure prediction accuracy, consistent with observations from the scaling analysis (**Fig. 1, D** and **E**). Additionally, this makes it possible to predict the performance of ESMFold from how well the language model understands the input sequence as quantified by perplexity.

We further conduct ablation studies to understand how different model components (**Fig. 2A**) affect the performance of ESMFold. With a much smaller folding trunk of 8 blocks, performance degrades to 0.74 lDDT (*baseline*). Without the language model, the single sequence performance on the CAMEO test set degrades substantially, to 0.58 lDDT. When removing the folding trunk entirely (i.e. only using the language model and the structure module), performance degrades to 0.66 lDDT. Any one ablation of (a) only 1 block of a structure module, (b) turning off recycling, (c) not using AlphaFold2 predicted structures as distillation targets, or (d) not using triangular updates all have similar performance

degradation (change in lDDT of -0.01 to -0.04). This leads us to conclude that the language model makes the largest contribution to atomic-resolution structure prediction performance on unseen proteins.

ESMFold matches AlphaFold2 performance on a majority of proteins (**Fig. 2C**). We find that this is true even on large proteins - T1076 is an example with 0.98 TM-score and 540 residues despite being trained on a maximum crop size of 384 (**Fig. 2D**). Parts of structure with low accuracy do not differ significantly between ESMFold and AlphaFold, suggesting that language models are learning information similar to what is provided by MSAs. We also observe that ESMFold is able to make good predictions for components of homo- and heterodimeric protein-protein complexes (**Fig. 2D**).

Language models enable more efficient predictions of protein structure

A notable advantage of ESMFold is its computational efficiency. The use of MSAs and templates in AlphaFold2 and RoseTTAFold creates two bottlenecks which ESMFold avoids. First, potentially expensive CPU-based search is required to retrieve and align the MSAs and templates. This process can average up to 30 minutes per input sequence with the standard AlphaFold2 pipeline[2], although runtime can be reduced to the order of seconds with newer approximate search methods like MMseqs (*49*). The main speed benefits come from improvements in model architecture. Instead of a two-dimensional sequence embedding state, AlphaFold2 and RoseTTAFold operate on three-dimensional internal states corresponding to the MSA using axial attention, which is expensive even when using a GPU.

By contrast, ESMFold is a fully end-to-end sequence to structure predictor and can be run entirely on GPU without access to any external databases. On a single NVIDIA V100 GPU, ESMFold makes a prediction on a protein with 384 residues in 14.2 seconds, 6X faster than a single AlphaFold2 model. On shorter sequences we see a ~60X improvement (**fig. S1**). Note that this excludes the CPU time for MSA and template search, as well as the 5X from the default ensemble of models. ESMFold can be run reasonably quickly on CPU, and an Apple M1 Macbook Pro makes the same prediction in just over 5 minutes (**fig. S1**).

[2] Average computed across all CASP14 sequences.

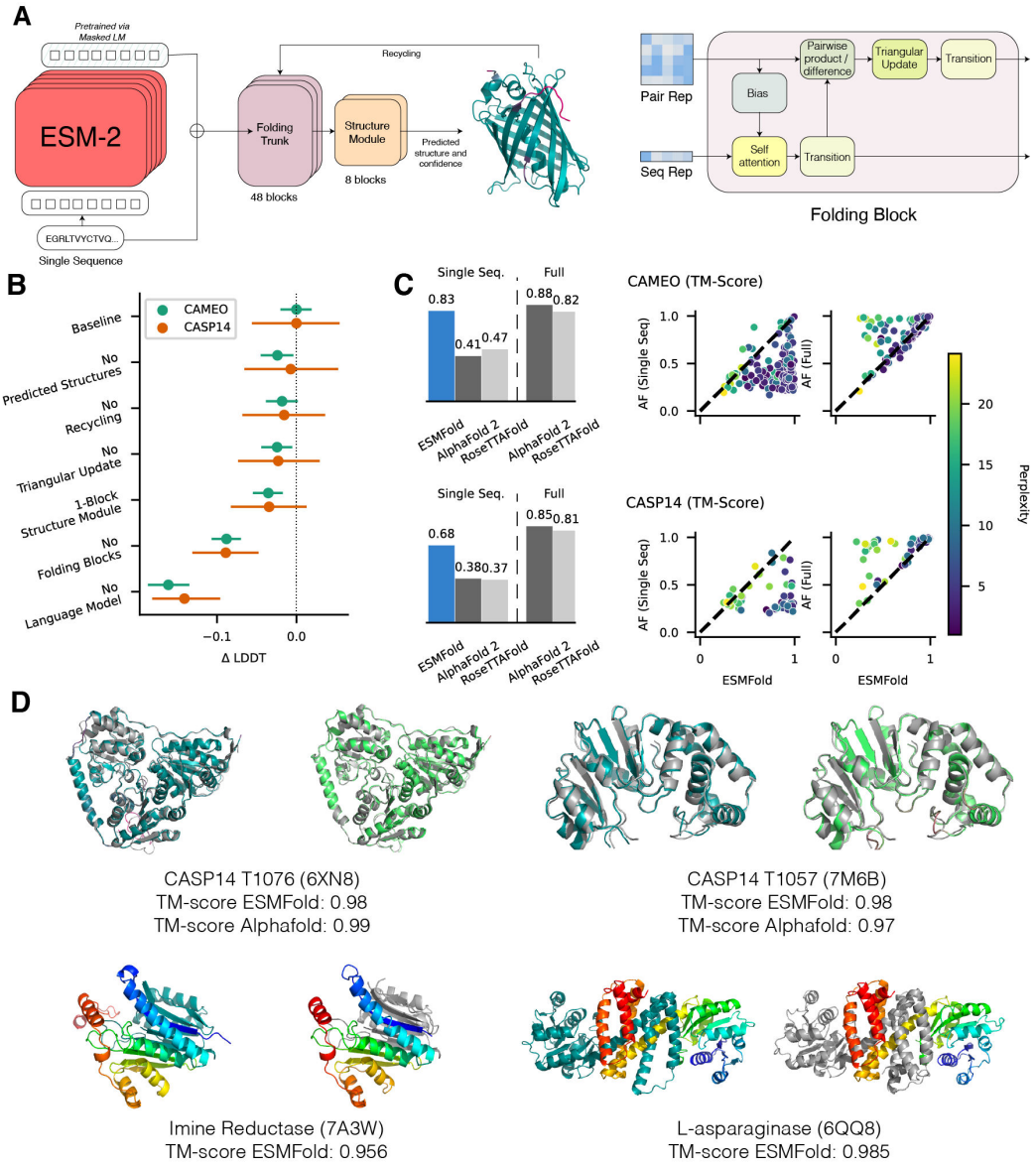

Figure 2: ESMFold enables accurate structure prediction from a single sequence.
(A) ESMFold model architecture. Arrows show the information flow in the network from the language model to the folding trunk to the structure module which outputs 3D coordinates and confidences. The folding trunk is a simplified single-sequence version of the EvoFormer described in AlphaFold2. **(B)** Effect of various ablations on ESMFold test-time performance. Language models are by far the biggest contributor. **(C)** ESMFold outperforms both RoseTTAFold and AlphaFold2 when given a single sequence as input, and is competitive with RoseTTAFold even when given full MSAs on CAMEO. Scatter-plots show ESMFold (x-axis) against AlphaFold2 (y-axis) performance, colored by perplexity. Proteins with low perplexity under our model score similarly to AlphaFold2. **(D)** *Top* shows test-set predictions of ESMFold in teal, ground truth in gray, and AlphaFold2 predictions in green. Pink shows low predicted lDDT for both ESMFold and AlphaFold2. *Bottom* shows complex predictions; chain B is colored teal for ESMFold (left) and gray for ground truth (right); chain A is colored rainbow from blue (N-terminal) to red (C-Terminal).

Exploring metagenomic structural space

The order of magnitude improvement in speed is a unique advantage of ESMFold over AlphaFold2 (**fig. S1**), enabling us to construct large sets of predicted structures in much shorter timescales than existing methods. This is especially important considering the scale of available sequence data. For example, the initial version of the AlphaFold2 Protein Structure Database (*50*) was released with ~360K predicted structures, and as of July 2022 contains ~995K predictions, which is orders of magnitude smaller than many databases of protein sequences.

Because rapid structure prediction may enable new insight into large collections of poorly studied or annotated proteins, we use ESMFold to predict the structures of 1 million randomly-sampled, non-redundant metagenomic protein sequences from the MGnify database (*29*) (**Methods**; **Fig. 3 and fig. S3**). Of these 1 million sequences, ESMFold assigns good confidence (mean pLDDT > 0.7) to ~29% of structures (N = 291,265) and high confidence (mean pLDDT > 0.9) to ~5.9% (N = 59,080). (**Fig. 3A**). We are able to compute these predictions in less than a day leveraging ~1600 GPU hours of compute, indicating that ESMFold can produce structure predictions for a much larger set of metagenomic sequences within a computationally tractable timeline.

Of the high-confidence structures, we identify ~12% (N = 7,271) with low structural similarity (TM-score < 0.5) to any known protein chain in the PDB (**Fig. 3B**), most of which also agree well with the predictions from full AlphaFold2 with MSA: 80% of corresponding ESMFold-AlphaFold2 predictions have TM-score greater than 0.7, and 50% have TM-score greater than 0.87 (**fig. S4**). We also find that ~2% (N = 1,143) of the high-confidence structures have low sequence similarity to any sequence in UniRef90 (**Methods**). Interestingly, we find 317 high-confidence structures that have both low structural and sequence similarity to proteins in these databases (for example, MGYP000706186022; **Fig. 3C** and **fig. S2**; **table S4**), indicating that ESMFold can identify regions of the protein landscape that are distant from existing knowledge.

Many high-confidence structures with low similarity to UniRef90 sequences, on the other hand, do have similar structures in the PDB, which can enable structure-based functional insight that would be difficult to obtain from sequence information alone. For example, MGnify sequence MGYP000936678158 has no significant matches to any entry in UniRef90, nor any significant matches via a jackhmmer (*51*) reference proteome search, but has a predicted structure conserved across many nucleases (PDB 5YET_B, TM-score 0.68; PDB 3HR4_A, TM-score 0.67) (**Fig. 3D** and **table S4**); similarly, MGnify sequence MGYP004000959047 has no significant UniRef90 or jackhmmer reference proteome matches but its predicted structure has high similarity to experimental structures of lipid binding domains (PDB 6BYM_A, TM-score 0.80; PDB 5YQP_B, TM-score 0.78) (**Fig. 3E** and **table S4**). These results illustrate how ESMFold's efficient structural prediction capabilities can augment our ability to explore large and poorly-understood regions of the metagenomic protein universe.

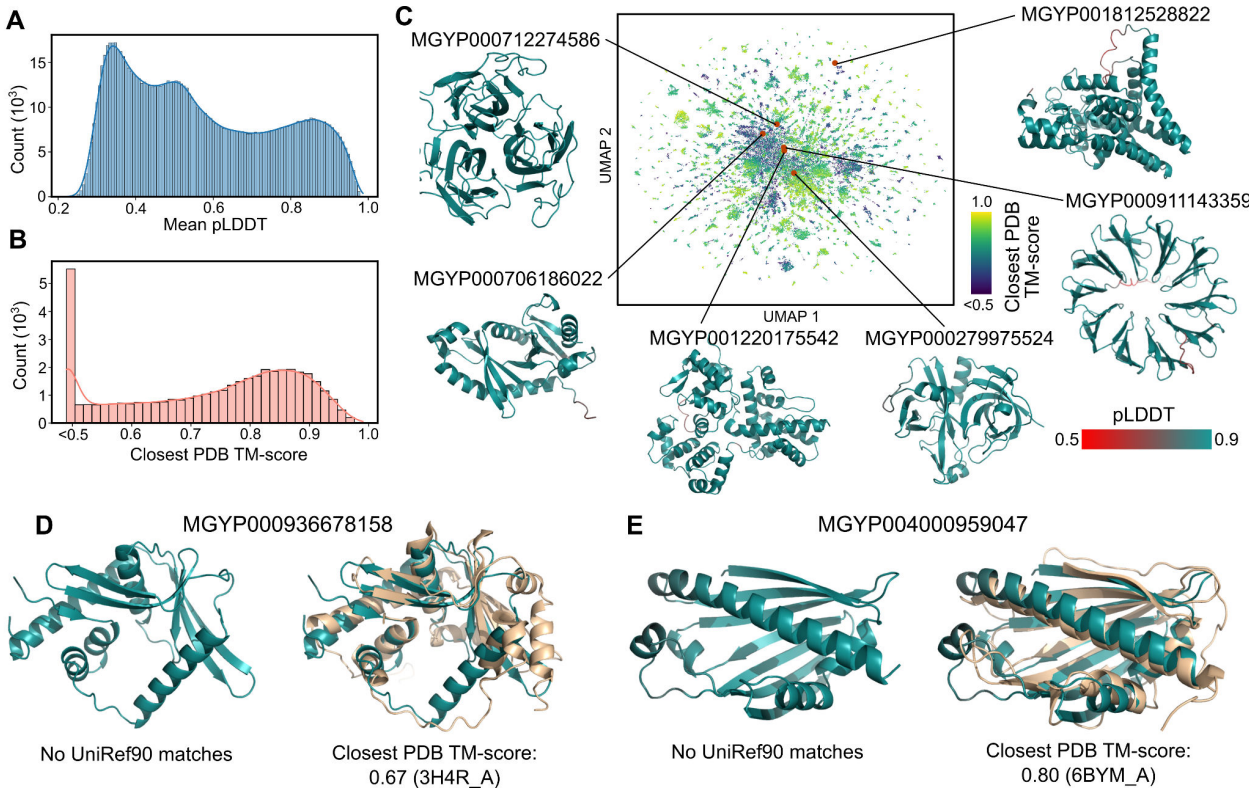

Figure 3: Exploring metagenomic structural space

(**A**) The distribution of mean pLDDT values computed for each of 1 million ESMFold-predicted structures from the MGnify database. (**B**) The distribution of the TM-score to the most similar PDB structure for each of ~59K highest confidence (mean pLDDT > 0.9) structures. Values were obtained by a Foldseek (*52*) search that does not report structures below a TM-score cutoff of 0.5. (**C**) High-confidence protein structures are visualized in two dimensions using the UMAP algorithm and colored according to distance from nearest PDB structure, where regions with low similarity to known structures are colored in dark blue. Example protein structures and their locations within the sequence landscape are provided; see also **figure S2** and **table S4**. (**D, E**) Examples of two ESMFold-predicted structures that have good agreement with experimental structures in the PDB but that have low sequence identity to any sequence in UniRef90, potentially enabling structure-based functional insight when sequence information is inadequate. (**D**) The predicted structure of MGYP000936678158 aligns to an experimental structure from a bacterial nuclease (light brown, PDB: 3H4R), while (**E**) the predicted structure of MGYP004000959047 aligns to an experimental sterol binding domain (light brown, PDB: 6BYM).

Conclusions

We find that language models trained with an unsupervised learning objective across a large database of evolutionarily diverse protein sequences enable atomic resolution prediction of protein structure. Scaling language models up to 15B parameters enables systematic study of the effect of scale on the learning of protein structure. We see non-linear improvements in protein structure predictions as a function of model scale, and observe a strong link between how well the language model understands a sequence (as measured by perplexity) and the structure prediction that emerges.

The ESM-2 family of models are the largest protein language models trained to date, with just an order of magnitude fewer parameters than the largest models of text that have recently been developed. ESM-2 has substantial improvements over prior models and even at 150M parameters ESM-2 captures a more accurate picture of structure than ESM-1 generation language models at 650M parameters. ESM-2 compares favorably to other recent, significant protein language models.

We show that the biggest driver of performance for ESMFold is the language model. With the strong link between language modeling perplexity and accuracy of structure prediction, we find that comparable predictions to state-of-the-art models can be obtained when the sequence is well understood by ESM-2.

ESMFold obtains accurate atomic resolution structure predictions with up to an order of magnitude improvement in inference time over AlphaFold2. In practice the speed advantages are even greater as ESMFold removes the need to search for evolutionarily related sequences to construct an MSA. The time for this search can be substantial, although new faster methods (*49*, *53*) can reduce this.

The inference time advantage makes it possible to efficiently map the structural space of large metagenomics sequence databases. Alongside structure-based tools for identifying remote homology and conservation, rapid and accurate structure prediction with ESMFold can help to play a role in structural and functional analysis of large collections of novel sequences. Obtaining millions of predicted structures within practical timescales can help reveal new insights into the breadth and diversity of natural proteins, and enable the discovery of new protein structures and functions.

Acknowledgements

We would like to thank Justas Dauparas, Laurens van der Maaten, Sergey Ovchinnikov, Ammar Rizvi, Jon Shepard, Joe Spisak, Sainbayar Sukhbaatar and Robert Verkuil for technical help, feedback, program support and discussions that helped shape this project.

References

1. Z. S. Harris, Distributional Structure. WORD. **10**, 146–162 (1954).
2. D. Altschuh, T. Vernet, P. Berti, D. Moras, K. Nagai, Coordinated amino acid changes in homologous protein families. *Protein Eng. Des. Sel.* **2**, 193–199 (1988).
3. C. Yanofsky, V. Horn, D. Thorpe, Protein Structure Relationships Revealed By Mutational Analysis. *Science*. **146**, 1593–4 (1964).
4. U. Göbel, C. Sander, R. Schneider, A. Valencia, Correlated mutations and residue contacts in proteins. *Proteins Struct. Funct. Bioinforma.* **18**, 309–317 (1994).
5. A. S. Lapedes, B. G. Giraud, L. Liu, G. D. Stormo, Correlated Mutations in Models of Protein Sequences: Phylogenetic and Structural Effects. *Lect. Notes-Monogr. Ser.* **33**, 236–256 (1999).
6. J. Thomas, N. Ramakrishnan, C. Bailey-Kellogg, Graphical models of residue coupling in protein families. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **5** (2008), pp. 183–197.
7. 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).
8. F. Morcos, A. Pagnani, B. Lunt, A. Bertolino, D. S. Marks, C. Sander, R. Zecchina, J. N. Onuchic, T. Hwa, M. Weigt, Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proc. Natl. Acad. Sci. U. S. A.* **108**, E1293–E1301 (2011).
9. T. A. Hopf, J. B. Ingraham, F. J. Poelwijk, C. P. I. Schärfe, M. Springer, C. Sander, D. S. Marks, Mutation effects predicted from sequence co-variation. *Nat. Biotechnol.* **35**, 128–135 (2017).
10. C. B. Anfinsen, Principles that Govern the Folding of Protein Chains. *Science*. **181**, 223–230 (1973).
11. T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language Models are Few-Shot Learners. *CoRR*. **abs/2005.14165** (2020) (available at https://arxiv.org/abs/2005.14165).
12. J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, Q. V. Le, FINETUNED LANGUAGE MODELS ARE ZERO-SHOT LEARNERS, 46 (2022).
13. J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, D. Zhou, Chain of Thought Prompting Elicits Reasoning in Large Language Models (2022), (available at http://arxiv.org/abs/2201.11903).
14. A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, N. Fiedel, PaLM: Scaling Language Modeling with Pathways (2022), , doi:10.48550/arXiv.2204.02311.
15. A. Rives, J. Meier, T. Sercu, S. Goyal, Z. Lin, J. Liu, D. Guo, M. Ott, C. L. Zitnick, J. Ma, R. Fergus, Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc. Natl. Acad. Sci.* **118** (2021), doi:10.1073/pnas.2016239118.
16. R. Rao, N. Bhattacharya, N. Thomas, Y. Duan, X. Chen, J. Canny, P. Abbeel, Y. S. Song, "Evaluating Protein Transfer Learning with TAPE" in *Neural Information Processing Systems* (Cold Spring Harbor Laboratory, 2019; https://doi.org/10.1101/676825 http://arxiv.org/abs/1906.08230).
17. T. Bepler, B. Berger, "Learning protein sequence embeddings using information from structure" in *International Conference on Learning Representations* (2019; https://openreview.net/forum?id=SygLehCqtm).
18. T. Bepler, B. Berger, Learning the protein language: Evolution, structure, and function. *Cell Syst.* **12**, 654-669.e3 (2021).

19. A. Elnaggar, M. Heinzinger, C. Dallago, G. Rihawi, Y. Wang, L. Jones, T. Gibbs, T. Feher, C. Angerer, D. Bhowmik, B. Rost, ProtTrans: Towards Cracking the Language of Lifes Code Through Self-Supervised Deep Learning and High Performance Computing. *IEEE Trans. Pattern Anal. Mach. Intell.* **14** (2021), doi:10.1109/TPAMI.2021.3095381.
20. A. Madani, B. McCann, N. Naik, N. S. Keskar, N. Anand, R. R. Eguchi, P.-S. Huang, R. Socher, ProGen: Language Modeling for Protein Generation. *bioRxiv* (2020) (available at http://arxiv.org/abs/2004.03497).
21. E. C. Alley, G. Khimulya, S. Biswas, M. AlQuraishi, G. M. Church, Unified rational protein engineering with sequence-only deep representation learning. *Nat. Methods*. **12**, 1315–1322 (2019).
22. M. Heinzinger, A. Elnaggar, Y. Wang, C. Dallago, D. Nechaev, F. Matthes, B. Rost, Modeling the language of life – Deep Learning Protein Sequences. *bioRxiv*, 614313 (2019).
23. B. Hie, E. D. Zhong, B. Berger, B. Bryson, Learning the language of viral evolution and escape. *Science*. **371**, 284–288 (2021).
24. R. Chowdhury, N. Bouatta, S. Biswas, C. Rochereau, G. M. Church, P. K. Sorger, M. AlQuraishi, "Single-sequence protein structure prediction using language models from deep learning" (preprint, Bioinformatics, 2021), , doi:10.1101/2021.08.02.454840.
25. J. A. Ruffolo, J. J. Gray, Fast, accurate antibody structure prediction from deep learning on massive set of natural antibodies. *Biophys. J.* **121**, 155a–156a (2022).
26. J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli, D. Hassabis, Highly accurate protein structure prediction with AlphaFold. *Nature*. **596**, 583–589 (2021).
27. M. Baek, F. DiMaio, I. Anishchenko, J. Dauparas, S. Ovchinnikov, G. R. Lee, J. Wang, Q. Cong, L. N. Kinch, R. D. Schaeffer, C. Millán, H. Park, C. Adams, C. R. Glassman, A. DeGiovanni, J. H. Pereira, A. V. Rodrigues, A. A. van Dijk, A. C. Ebrecht, D. J. Opperman, T. Sagmeister, C. Buhlheller, T. Pavkov-Keller, M. K. Rathinaswamy, U. Dalwadi, C. K. Yip, J. E. Burke, K. C. Garcia, N. V. Grishin, P. D. Adams, R. J. Read, D. Baker, Accurate prediction of protein structures and interactions using a three-track neural network. *Science*. **373**, 871–876 (2021).
28. B. E. Suzek, Y. Wang, H. Huang, P. B. McGarvey, C. H. Wu, U. Consortium, UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics*. **31**, 926–932 (2014).
29. A. L. Mitchell, A. Almeida, M. Beracochea, M. Boland, J. Burgin, G. Cochrane, M. R. Crusoe, V. Kale, S. C. Potter, L. J. Richardson, E. Sakharova, M. Scheremetjew, A. Korobeynikov, A. Shlemov, O. Kunyavskaya, A. Lapidus, R. D. Finn, MGnify: the microbiome analysis resource in 2020. *Nucleic Acids Res.*, gkz1035 (2019).
30. M. Steinegger, M. Mirdita, J. Söding, Protein-level assembly increases protein sequence recovery from metagenomic samples manyfold. *Nat. Methods*. **16**, 603–606 (2019).
31. J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, D. Amodei, Scaling laws for neural language models. *ArXiv Prepr. ArXiv200108361* (2020).
32. J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. de L. Casas, L. A. Hendricks, J. Welbl, A. Clark, T. Hennigan, E. Noland, K. Millican, G. van den Driessche, B. Damoc, A. Guy, S. Osindero, K. Simonyan, E. Elsen, J. W. Rae, O. Vinyals, L. Sifre, Training Compute-Optimal Large Language Models (2022), doi:10.48550/arXiv.2203.15556.
33. S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. Diab, X. Li, X. V. Lin, T. Mihaylov, M. Ott, S. Shleifer, K. Shuster, D. Simig, P. S. Koura, A. Sridhar, T. Wang, L. Zettlemoyer, OPT: Open Pre-trained Transformer Language Models (2022), doi:10.48550/arXiv.2205.01068.
34. J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" in *Proceedings of the 2019 Conference of the North {A}merican Chapter of the Association for Computational Linguistics: Human Language*

Technologies, Volume 1 (Long and Short Papers) (Association for Computational Linguistics, Minneapolis, Minnesota, 2019; http://arxiv.org/abs/1810.04805), pp. 4171–4186.

35. A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, S. Bowman, "GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding" in *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP* (Association for Computational Linguistics, Brussels, Belgium, 2018; https://aclanthology.org/W18-5446), pp. 353–355.
36. A. Rogers, O. Kovaleva, A. Rumshisky, A Primer in BERTology: What We Know About How BERT Works. *Trans. Assoc. Comput. Linguist.* **8**, 842–866 (2020).
37. J. Vig, A. Madani, L. R. Varshney, C. Xiong, R. Socher, N. Rajani, "BERTology Meets Biology: Interpreting Attention in Protein Language Models" in (2020; https://openreview.net/forum?id=YWtLZvLmud7).
38. R. Rao, J. Meier, T. Sercu, S. Ovchinnikov, A. Rives, "Transformer protein language models are unsupervised structure learners" in *International Conference on Learning Representations* (Cold Spring Harbor Laboratory, 2021), p. 2020.12.15.422761.
39. S. Balakrishnan, H. Kamisetty, J. G. Carbonell, S.-I. Lee, C. J. Langmead, Learning generative models for protein fold families. *Proteins Struct. Funct. Bioinforma.* **79**, 1061–1078 (2011).
40. S. K. Burley, H. M. Berman, C. Bhikadiya, C. Bi, L. Chen, L. Di Costanzo, C. Christie, K. Dalenberg, J. M. Duarte, S. Dutta, Z. Feng, S. Ghosh, D. S. Goodsell, R. K. Green, V. Guranoví, D. Guzenko, B. P. Hudson, T. Kalro, Y. Liang, R. Lowe, H. Namkoong, E. Peisach, I. Periskova, A. Prlí, C. Randle, A. Rose, P. Rose, R. Sala, M. Sekharan, C. Shao, L. Tan, Y.-P. Tao, Y. Valasatava, M. Voigt, J. Westbrook, J. Woo, H. Yang, J. Young, M. Zhuravleva, C. Zardecki, RCSB Protein Data Bank: biological macromolecular structures enabling research and education in fundamental biology, biomedicine, biotechnology and energy. *Nucleic Acids Res.* **47** (2019), doi:10.1093/nar/gky1004.
41. J. Haas, A. Barbato, D. Behringer, G. Studer, S. Roth, M. Bertoni, K. Mostaguir, R. Gumienny, T. Schwede, Continuous Automated Model EvaluatiOn (CAMEO) complementing the critical assessment of structure prediction in CASP12. *Proteins Struct. Funct. Bioinforma.* **86**, 387–398 (2018).
42. A. Kryshtafovych, T. Schwede, M. Topf, K. Fidelis, J. Moult, Critical assessment of methods of protein structure prediction (CASP)—Round XIV. *Proteins Struct. Funct. Bioinforma.* **89**, 1607–1617 (2021).
43. Y. Zhang, J. Skolnick, Scoring function for automated assessment of protein structure template quality. *Proteins*. **57**, 702–710 (2004).
44. K. K. Yang, A. X. Lu, N. Fusi, Convolutions are competitive with transformers for protein sequence pretraining (2022), doi:10.1101/2022.05.19.492714.
45. 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**, 1–34 (2017).
46. Y. Liu, P. Palmedo, Q. Ye, B. Berger, J. Peng, Enhancing Evolutionary Couplings with Deep Convolutional Neural Networks. *Cell Syst.* **6**, 65–74 (2018).
47. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, "Attention Is All You Need" in *Advances in Neural Information Processing Systems* (2017; https://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf), pp. 5998–6008.
48. C. Hsu, R. Verkuil, J. Liu, Z. Lin, B. Hie, T. Sercu, A. Lerer, A. Rives, "Learning inverse folding from millions of predicted structures" in *Proceedings of the 39th International Conference on Machine Learning* (PMLR, 2022; https://proceedings.mlr.press/v162/hsu22a.html), pp. 8946–8970.
49. M. Hauser, M. Steinegger, J. Söding, MMseqs software suite for fast and deep clustering and searching of large protein sequence sets. *Bioinformatics*. **32**, 1323–1330 (2016).
50. M. Varadi, S. Anyango, M. Deshpande, S. Nair, C. Natassia, G. Yordanova, D. Yuan, O. Stroe, G. Wood, A. Laydon, A. Žídek, T. Green, K. Tunyasuvunakool, S. Petersen, J. Jumper, E. Clancy, R. Green, A. Vora, M. Lutfi, M. Figurnov, A. Cowie, N. Hobbs, P. Kohli, G. Kleywegt, E. Birney, D. Hassabis, S. Velankar, AlphaFold Protein Structure Database: massively expanding the structural

coverage of protein-sequence space with high-accuracy models. *Nucleic Acids Res.* **50**, D439–D444 (2022).
51. S. C. Potter, A. Luciani, S. R. Eddy, Y. Park, R. Lopez, R. D. Finn, HMMER web server: 2018 update. *Nucleic Acids Res.* **2**, W200–W204 (2018).
52. M. van Kempen, S. S. Kim, C. Tumescheit, M. Mirdita, C. L. M. Gilchrist, J. Söding, M. Steinegger, Foldseek: fast and accurate protein structure search (2022), doi:10.1101/2022.02.07.479398.
53. M. Mirdita, K. Schütze, Y. Moriwaki, L. Heo, S. Ovchinnikov, M. Steinegger, ColabFold: making protein folding accessible to all. *Nat. Methods*. **19**, 679–682 (2022).
54. J. Meier, R. Rao, R. Verkuil, J. Liu, T. Sercu, A. Rives, "Language models enable zero-shot prediction of the effects of mutations on protein function" (preprint, Synthetic Biology, 2021), , doi:10.1101/2021.07.09.450648.
55. R. Rao, J. Liu, R. Verkuil, J. Meier, J. Canny, P. Abbeel, T. Sercu, A. Rives, "MSA Transformer" in *Proceedings of the 38th International Conference on Machine Learning* (PMLR, 2021; https://proceedings.mlr.press/v139/rao21a.html), pp. 8844–8856.
56. S. D. Dunn, L. M. Wahl, G. B. Gloor, Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics*. **24**, 333–340 (2008).
57. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, É. Duchesnay, Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
58. J. Yang, I. Anishchenko, H. Park, Z. Peng, S. Ovchinnikov, D. Baker, Improved protein structure prediction using predicted interresidue orientations. *Proc. Natl. Acad. Sci.* **117**, 1496–1503 (2020).
59. J. Su, Y. Lu, S. Pan, B. Wen, Y. Liu, RoFormer: Enhanced Transformer with Rotary Position Embedding (2021), (available at http://arxiv.org/abs/2104.09864).
60. Y. You, J. Li, S. Reddi, J. Hseu, S. Kumar, S. Bhojanapalli, X. Song, J. Demmel, K. Keutzer, C.-J. Hsieh, LARGE BATCH OPTIMIZATION FOR DEEP LEARNING: TRAINING BERT IN 76 MINUTES, 38 (2020).
61. S. Rajbhandari, J. Rasley, O. Ruwase, Y. He, "ZeRO: Memory optimizations Toward Training Trillion Parameter Models" in *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis* (2020), pp. 1–16.
62. J. Ho, N. Kalchbrenner, D. Weissenborn, T. Salimans, Axial Attention in Multidimensional Transformers. *arXiv* (2019) (available at http://arxiv.org/abs/1912.12180).
63. L. McInnes, J. Healy, J. Melville, UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction (2018), doi:10.48550/ARXIV.1802.03426.
64. F. A. Wolf, P. Angerer, F. J. Theis, SCANPY: large-scale single-cell gene expression data analysis. *Genome Biol.* **19**, 15 (2018).

Supplementary Figures

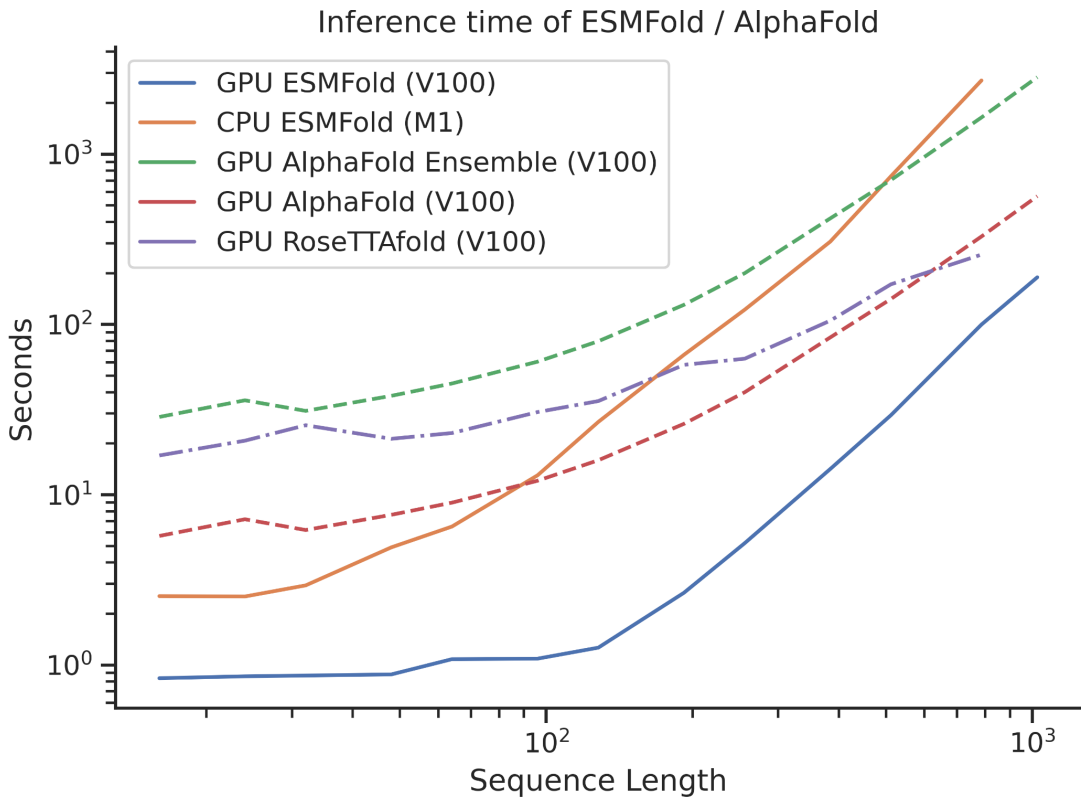

Figure S1: ESMFold vs AlphaFold2 and RoseTTAfold timing experiments

We test the speed of ESMFold vs AlphaFold2 on sequence lengths up to 1024. At low sequence lengths, ESMFold is dominated by language model performance, while the $O(N^3)$ computation of pairwise representations takes over at high sequence lengths. Most of the speed advantage of ESMFold comes from not needing to process the MSA branch. We see an over 60x speed advantage for shorter protein sequences, and a reasonable speed advantage for longer protein sequences. We also do not count Jax graph compilation times or MSA search times for AlphaFold2 - meaning in practice there is a larger performance advantage in the cold start case. We also use an optimized Colabfold 1.3.0 (*53*) to do speed comparison. No significant optimization has been performed on ESMFold, and we suspect that further gains can be made by optimizing ESMFold as well. For RoseTTAfold, the speed of the SE(3) Transformer dominates, especially at low sequence lengths. The number of SE(3) max-iterations are artificially limited to 20 (default 200) and no MSAs are used as input for these measurements. Additionally, this only measures the network forward time, and does not include the time taken to compute sidechains with PyRosetta or search for MSAs. These comparisons are much more favorable towards AlphaFold2 and RoseTTAfold.

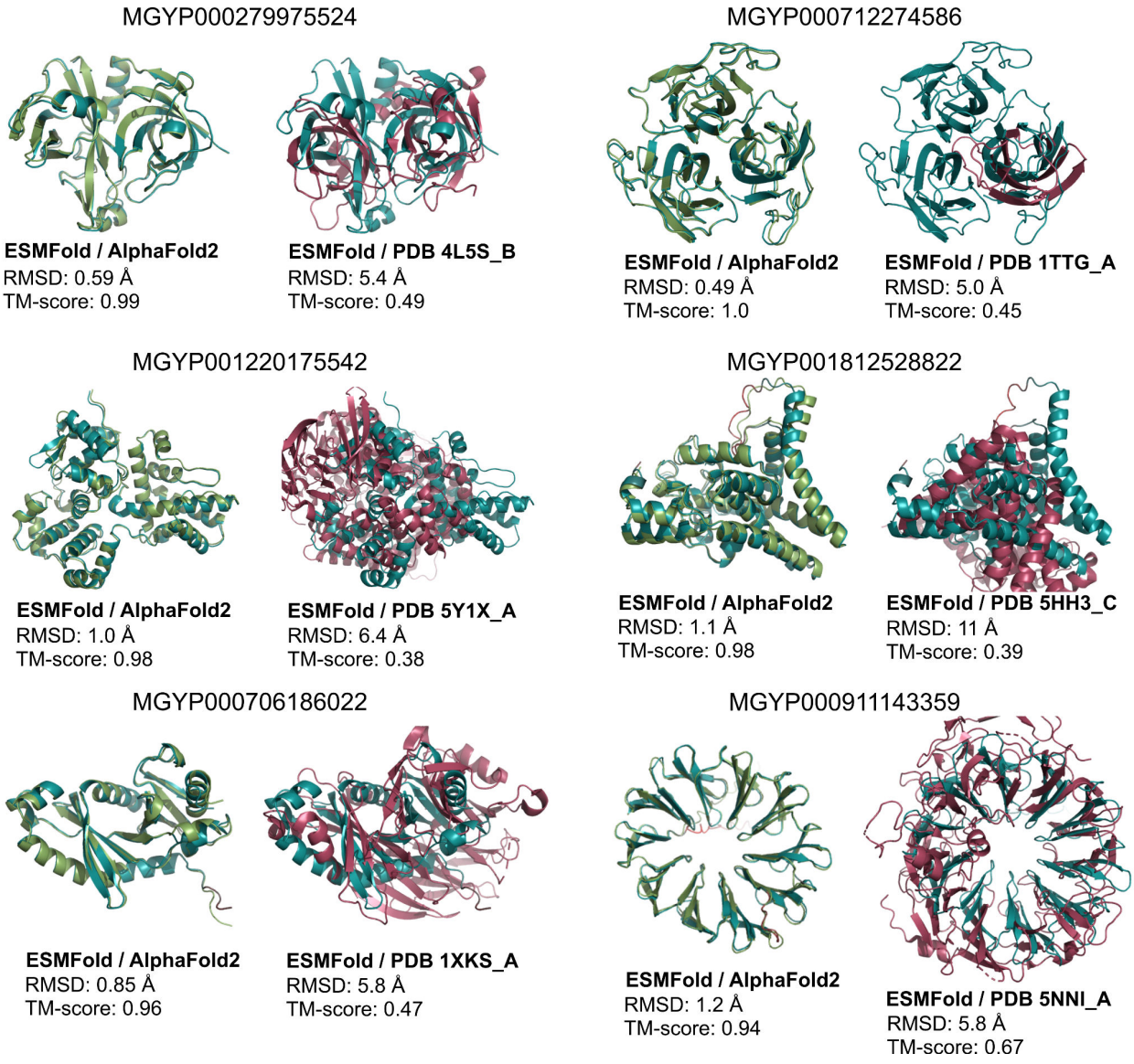

Figure S2: Highlighted ESMFold structure predictions, comparison to AlphaFold2, and comparison to closest PDB structure, related to Figure 3.

Example predicted structures from six different metagenomic sequences; also see **table S4**. Left of each subfigure: The prediction is displayed with the AlphaFold2 prediction (light green). Right of each subfigure: The prediction is displayed with the Foldseek-determined nearest PDB structure according to TM-score.

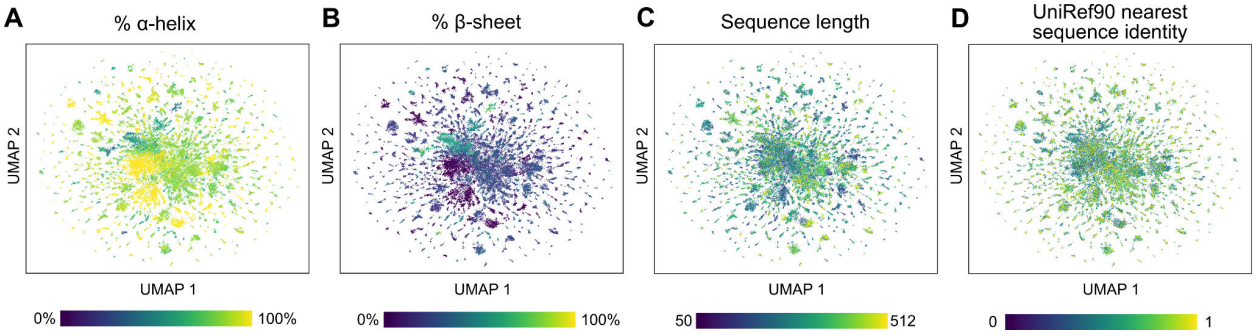

Figure S3: Sequence landscape UMAP visualizations, related to Figure 3.

Additional UMAP plots in which MGnify sequences are plotted according to the same coordinates as in **Figure 3C**, but colored by secondary structure percentage (**A, B**), sequence length (**C**), or the sequence identity to the most similar entry in UniRef90 according to a blastp search (**D**).

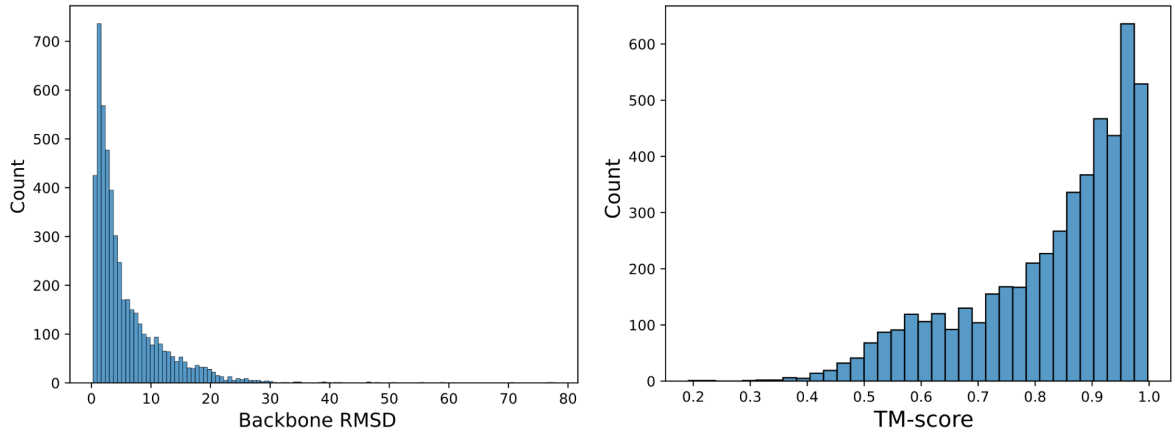

Figure S4: Comparison to AlphaFold2 of structurally remote ESMFold predictions, related to Figure 3.

Distributions of backbone RMSDs (left) and TM-scores (right) of ESMFold-AlphaFold2 predictions of the same sequence, where the ESMFold prediction has both high confidence (mean pLDDT > 0.9) and relatively low structural similarity to the PDB (Foldseek closest PDB TM-score < 0.5).

Supplementary Tables

	8M	35M	150M	650M	3B	15B
Dataset	UR50/D	UR50/D	UR50/D	UR50/D	UR50/D	UR50/D
Number of layers	6	12	30	33	36	48
Embedding dim	320	480	640	1280	2560	5120
Attention heads	20	20	20	20	40	40
Training steps	500K	500K	500K	500K	500K	270K
Learning rate	4e-4	4e-4	4e-4	4e-4	4e-4	1.6e-4
Weight decay	0.01	0.01	0.01	0.01	0.01	0.1
Clip norm	0	0	0	0	1.0	1.0
Distributed backend	DDP	DDP	DDP	DDP	FSDP	FSDP

Table S1: ESM-2 model parameters at different scales

	LR P@L	LR P@L/5	Validation Perplexity
Baseline	0.381	0.626	8.42
No RoPE	0.365	0.599	8.62
Older UniRef Data	0.368	0.599	7.98
No UR90 Sampling	0.387	0.631	8.40

Table S2: ESM-2 Architecture Ablations

Model	# Params	# Updates	Validation Perplexity	LR P@L	LR P@L/5	CASP14	CAMEO
ESM-2	8M	270k	10.45	0.16	0.28	0.37	0.48
	35M	270k	9.12	0.29	0.49	0.41	0.56
	150M	270k	8.00	0.42	0.68	0.47	0.63
	650M	270k	7.23	0.50	0.77	0.51	0.68
	3B	270k	6.73	0.53	0.80	0.51	0.71
	8M	500k	10.33	0.17	0.29	0.37	0.48
	35M	500k	8.95	0.30	0.51	0.41	0.56
	150M	500k	7.75	0.44	0.70	0.49	0.65
	650M	500k	6.95	0.52	0.79	0.51	0.70
	3B	500k	6.49	**0.54**	0.81	0.52	**0.72**
	15B	270k	**6.37**	**0.54**	**0.82**	**0.55**	**0.72**
ESM-1b[3]	650M	–	–	0.41	0.66	0.42	0.64
Prot-T5-XL (UR50) (*19*)	3B	–	–	0.48	0.72	0.50	0.69
Prot-T5-XL (BFD) (*19*)	3B	–	–	0.36	0.58	0.46	0.63
CARP (*44*)	640M	–	–	–	–	0.42	0.59

Table S3: Detailed language model comparison on structure prediction and unsupervised contact prediction.

Table S3 shows a comparison of ESM-2 language models with other language models on structure prediction and unsupervised contact prediction. ESM-2 language models are compared at different numbers of parameters and at different numbers of training updates. Training updates and validation perplexity are not reported for baseline models, since there is no straightforward comparison. For the number of training updates, different models use different batch sizes, so the number of sequences seen can vary even if the number of updates are the same. For validation perplexity, baseline models are not trained on the same dataset, and do not share a common heldout validation set with ESM-2. Unsupervised contact precision results, in the form of long range precision at L and at L / 5, do allow us to compare all transformer language models despite variance in training data. However, CARP, a convolution based language model, does not have attention maps with which to identify protein contacts. Despite this, supervised training manages to extract reasonable contact prediction results approximately on par with ESM-1b.

[3] ESM-1b evaluated only on sequences of length < 1024, due to constraints with position embedding.

MGnify ID	Mean pLDDT	Foldseek server closest TM-score	Foldseek server closest PDB	Closest blastp sequence identity (UniRef90)	Closest blastp sequence (UniRef90)
MGYP000712274586	0.96	0.45	1ttg_A	54%	UniRef90_A0A539E457 Uncharacterized protein (Acidimicrobiaceae bacterium)
MGYP000911143359	0.90	0.67	5nni_A	43%	UniRef90_A0A7Y5V7P8 Uncharacterized protein (Flavobacteriales bacterium)
MGYP001220175542	0.94	0.38	5y1x_A	98%	UniRef90_UPI0013011942 Helix-turn-helix domain-containing protein (Caenibacillus caldisaponilyticus)
MGYP001812528822	0.93	0.39	5hh3_C	50%	UniRef90_A0A545U581 Fatty acid desaturase (Exilibacterium tricleocarpae)
MGYP000706186022	0.92	0.47	1xks_A	29%	UniRef90_A0A6N6S1Z1 Uncharacterized protein (Candidatus brocadia)
MGYP000279975524	0.93	0.49	4l5s_B	38%	UniRef90_A0A1F4EWL6 Uncharacterized protein (Betaproteobacteria bacterium)
MGYP004000959047	0.90	0.80	6bym_A	No significant matches	NA
MGYP000936678158	0.95	0.68	5yet_B	No significant matches	NA

Table S4: Information on highlighted MGnify proteins, related to Figure 3.

MGnify sequence identifiers corresponding to predicted structures highlighted throughout this study, including the PDB chain and corresponding TM-score of the closest structure identified by the Foldseek webserver as well as the UniRef90 entry and sequence identity of the closest sequence identified by blastp (**Methods**).

Methods

1.1 Data

1.1.1 ESM

UniRef50, April 2021 version, is used for the training of ESM models. We partition the training dataset by randomly selecting 0.5% (≈250,000) sequences to form our validation set. The training set has sequences removed via the procedure described in (*54*). MMseqs search (–min-seq-id 0.5 –alignment-mode 3 –max-seqs 300 -s 7 -c 0.8 –cov-mode 0) is run using the train set as query database and the validation set as target database. All train sequences which match a validation sequence with 50% sequence identity under this search are removed from the train set.

We also filter out de-novo designed proteins from the pretraining dataset via two filters. First, we remove any sequence in UniRef50 and UniRef90 that was annotated as "artificial sequence" by a taxonomy search on the UniProt website, when 2021_04 was the most recent release (1,027 proteins). Second, we use jackhmmer to remove all hits around a manually curated set of 81 de-novo proteins. jackhmmer was run with `--num-iter 1 –max` flags, with each of the 81 de-novo proteins as a query and UniRef100 as a search database. All proteins returned by jackhmmer were removed from both UniRef50 and UniRef90 via their UniRef IDs (58,462 proteins). This filtering is performed to enable future work evaluating the generalization of language models to de-novo sequences.

To increase the amount of data and its diversity, we sampled a minibatch of UniRef50 sequences for each training update. We then replaced each sequence with a sequence sampled uniformly from the corresponding UniRef90 cluster. This allowed ESM-2 models to train on over 60M protein sequences.

1.1.2 Structure Training Sets

For training ESMFold, we closely follow the training procedure outlined in (*26*). We find all PDB chains until 2020-05-01 with resolution greater than or equal to 9Å and length greater than 20. All proteins where over 20% of the sequence is the same residue is not considered. We use MMseqs easy-cluster with default parameters to cluster resulting sequences at 40% sequence identity. Only individual chains are used during training, even when the chain is part of a protein complex.

At training time, we sample each cluster evenly, and then sample a random protein from each cluster. We also do the same rejection sampling technique, where protein chains are accepted with probability $\frac{1}{512}$ max(min(Nres, 512), 256). This means longer proteins are trained on more frequently.

The predicted structures dataset is the same generated from Hsu et al. 2022 (*48*). This dataset is a set of 13,477,259 structures predicted using AlphaFold2 on MSAs generated via the process in (*55*). We then filter this dataset by predicted lDDT greater than 70. Because of the way the dataset is constructed, we lose only 1.5% of the dataset with this filter. Additionally, during backpropagation, we do not backpropagate residues where predicted lDDT is less than 70. We found that this is necessary to obtain increased performance using predicted structures. We sample from predicted structures 75% of the time, and real structures 25% of the time during training.

1.1.3 Structure Validation and Test Sets

During method development (e.g. hyperparameter selection), we used a temporally held out validation set obtained from the Continuous Automated Model EvaluatiOn (CAMEO) server (*41*) by filtering from August 2021 to January 2022.

We report results by testing 3D structure prediction models on two test sets, both chosen to be temporally held out from our supervised training set. The first test is from CAMEO, consisting of all 194 test proteins from April 01, 2022 through June 25, 2022. Our second test set consists of 51 targets from the CASP14 competition (*42*). For both test sets, metrics are computed on all modeled residues in the PDB file. The full CASP14 target list is:
T1024,T1025,T1026,T1027,T1028,T1029,T1030,T1031,T1032,T1033,T1034,T1035,T1036s1,T1037,T1038,T1039,T1040,T1041,T1042,T1043,T1044,T1045s1,T1045s2,T1046s1,T1046s2,T1047s1,T1047s2,T1049,T1050,T1053,T1054,T1055,T1056,T1057,T1058,T1064,T1065s1,T1065s2,T1067,T1070,T1073,T1074,T1076,T1078,T1079,T1080,T1082,T1089,T1090,T1091,T1099.

These are all publicly available CASP14 targets as of July 2022.

No filtering is performed on these test sets, as ESMFold is able to make predictions on all sequences, including the length-2166 target T1044.

1.2 Language Models

1.2.1 Unsupervised Contact Prediction

The unsupervised contact prediction methodology used throughout this work is taken from Rao et al. 2021 (*38*). They show that transformer language models trained on large databases of protein sequences learn to predict protein contacts in their attention maps using little to no supervision. We exploit this fact as a very computationally inexpensive method for measuring a language model's knowledge of protein structure.

Rather than training an atomic-level structure predictor for each checkpoint of each language model (a process which can take several days for the largest language models), we use the logistic regression described in Rao et al. 2021 for contact prediction. The probability of a contact is defined as:

Let c_{ij} be a boolean random variable which is true if amino acids i, j are in contact. Suppose our transformer has L layers and K attention heads per layer. Let A_{kl} be the symmetrized and APC-corrected (*56*) attention map for the k-th attention head in the l-th layer of the transformer, and a_{ijkl} be the value of that attention map at position i, j. Then

$$p(c_{ij}) = (1 + \exp(-\beta_0 - \sum_{l=1}^{L}\sum_{k=1}^{K}\beta_{kl}\alpha^{kl}_{ij}))^{-1}$$

The parameters β are fit in scikit-learn (*57*) using L1-regularized logistic regression with $\lambda = 0.15$. The regression is fit using the 20 protein training set from Rao et al. 2021 (*38*), which was simply a random selection from the trRosetta (*58*) training set. We performed a variability analysis using 20 bootstrapped

samples of 20 training proteins from the total set of 14862 proteins. The average long range P@L was 0.4287 with a standard deviation of 0.0028. We also performed experiments using larger training sets, but observed no significant performance change. Given these results, we are confident that selecting a subset of 20 proteins for training provides a good estimate of contact precision performance.

Unsupervised contact prediction results are reported for the 14842 protein test set used in Rao et al. 2021, which is also derived from the trRosetta training set. For both training and test a contact is defined as two amino acids with C-alpha distance < 8Å.

1.2.2 Perplexity Calculation

Perplexity is a measure of a language model's uncertainty of a sequence and is defined as the exponential of the negative log-likelihood of the sequence. Unfortunately, there is no efficient method of computing the log-likelihood of a sequence under a masked language model. Instead, there are two methods we can use for estimating perplexity.

First, let the mask M be a random variable denoting a set of tokens from input sequence x. Each token has a 15% probability of inclusion. If included the tokens have an 80% probability of being replaced with a mask token, a 10% probability of being replaced with a random token, and a 10% probability of being replaced with an unmasked token. Let $\hat{x}_{i \in M}$ denote the set of modified input tokens. The perplexity is then defined as

$$Perplexity(x) = \exp\left\{-\log p(x_{i \in M} | x_{j \notin M} \cup \hat{x}_{i \in M}\right\}$$

As the set M is a random variable, this expression is non-deterministic. This makes it a poor estimate of the perplexity of a single sequence. However, it requires only a single forward pass of the model to compute, so it is possible to efficiently obtain an estimate of the expectation of this expression over a large dataset. When reporting the perplexity over a large dataset (such as our UniRef validation set), this estimate is used.

The second perplexity calculation is the pseudo-perplexity, which is the exponential of the negative pseudo-log-likelihood of a sequence. This estimate provides a deterministic value for each sequence, but requires L forward passes to compute, where L is the length of the input sequence. It is defined as

$$PseudoPerplexity(x) = \exp\left\{-\frac{1}{L}\sum_{i=1}^{L} \log p(x_i | x_{j \neq i})\right\}$$

When reporting the perplexity for an individual sequence (e.g. on CASP14 or CAMEO), this estimate is used. For brevity, we refer to both of these estimates as the "perplexity," as they can be interpreted in a similar manner.

1.2.3 ESM-2 Architecture

We use a BERT (*34*) style encoder only transformer architecture (*47*) with modifications. We change the number of layers, number of attention heads, hidden size and feed forward hidden size as we scale the ESM model (**table S1**).

The original transformer paper uses absolute sinusoidal positional encoding to inform the model about token positions. These positional encodings are added to the input embeddings at the bottom of the encoder stack. In ESM-1b (*15*), we replaced this static sinusoidal encoding with a learned one. Both static and learned absolute encodings provide the model a very cheap way of adding positional information. However, absolute positional encoding methods don't extrapolate well beyond the context window they are trained on. In ESM-2, we used Rotary Position Embedding (RoPE) (*59*) to allow the model extrapolate beyond the context window it is trained on. RoPE slightly increases the computational cost of the model, since it multiplies every query and key vector inside the self attention with a sinusoidal embedding. In our experiments, we observed that this improves model quality for small models. However, we observed that the performance improvements start to disappear as the model size and training duration get bigger.

1.2.4 Training Details

In ESM-2, we have made multiple small modifications to our model with the goal of increasing the effective capacity of our models. ESM-1b had dropout both in hidden layers and attention which we removed completely to free up more capacity. In our experiments, we did not observe any significant performance regressions with this change.

We trained most of our models on a network with multiple nodes connected via a network interface. As the models get bigger, the amount of communication becomes the fundamental bottleneck for the training speed. Since BERT style models have been shown to be amenable to very large batch sizes (*60*), we increased our effective batch size to 2M tokens.

For model training optimization, we used Adam with $\beta_1 = 0.9$, $\beta_2 = 0.98$, $\epsilon = 10^{-8}$ and L_2 weight decay of 0.01 for all models except the 15 billion parameter model, where we used a weight decay of 0.1. The learning rate is warmed up over the first 2,000 steps to a peak value of 4e-4 (1.6e-4 for the 15B parameter model), and then linearly decayed to one tenth of its peak value over the 90% of training duration. We trained all models for 500K updates except the 15B model which we trained for 270K steps. All models used 2 million tokens as batch size except the 15B model where we used 3.2 million tokens batch size. In order to efficiently process large proteins, we cropped long proteins to random 1024 tokens. We used BOS and EOS tokens to signal the beginning and end of a real protein, to allow the model to separate a full sized protein from a cropped one.

We used standard distributed data parallelism for models up to 650M parameters and used sharded data parallelism (FSDP) (*61*) for the 2.8B and 15B parameter models. FSDP shards model weights and optimization parameters across multiple GPUs, allowing us to train models that can't fit into a single GPU memory.

1.2.5 ESM-2 Ablation Details

We ran ablation experiments using 150M parameter models trained for 100K steps. Ablations were performed for RoPE, new data version, and UniRef90 sampling (**table S2**).

Unsupervised contact prediction results show that both RoPE and newer data significantly improve the results. We do observe a slight regression when sampling from UniRef90 clusters, however we believe this difference is small and the UniRef90 cluster sampling is likely to help for the larger models.

1.3 ESMFold

The AlphaFold2 architecture is split into two major sections, the Evoformer and the structure module. The structure module processes the final representations into 3D coordinates for atomic-level structure predictions and requires no changes to be used with ESM-2. The Evoformer, however, builds separate MSA and residue-pairwise embedding spaces.

The major change that needs to be made in order to adapt the Evoformer block to language model features is to remove its dependence on MSAs. Since MSAs are two dimensional, the Evoformer employs axial attention (*62*) over the columns and rows of the MSA. The language model features are one dimensional, so we can replace the axial attention with a standard attention over this feature space. All other operations in the Evoformer block are kept the same. We call this simplified architecture the Folding block.

The second change involves the removal of templates. Template information is passed to the model as pairwise distances, input to the residue-pairwise embedding. We simply omit this information, passing instead the attention maps from the language model, as these have been shown to capture structural information well (*38*).

Our final architecture, which we call ESMFold, has 48 folding blocks. It was trained for an initial 125k steps on protein crops of size 256, and then fine-tuned with the structural violation loss for 25k steps, on crop sizes of 384. We use the Frame Aligned Point Error (FAPE) and distogram losses introduced in AlphaFold2, as well as heads for predicting lDDT and the pTM score. We omit the masked language modeling loss. Language model parameters are frozen for training ESMFold.

The folding block is as follows, and shown in **Figure 2A**:

```
Algorithm 1:
FoldingBlock(s, z)
b = Linear(z)
s = s + MultiHeadSelfAttention(s, bias=b)
s = s + MLP(s)
z = z + Linear(Concat([OuterProduct(s), OuterDifference(s)]))
z = z + TriangularMultiplicativeUpdateOutgoing(z)
z = z + TriangularMultiplicativeUpdateIncoming(z)
z = z + TriangularSelfAttentionOutgoing(z)
z = z + TriangularSelfAttentionIncoming(z)
z = z = MLP(z)
return s, z
```

The folding block is extremely similar to the Evoformer described in AlphaFold2. There are two major differences. Because we do not have an MSA, the modules responsible for processing the MSA are replaced with a simple transformer. The self-attention here still uses a bias derived from the pairwise representations. Secondly, the sequence representations communicate with pairwise representation via both an outer product and outer difference.

And ESMFold is as follows:

```
Algorithm 2:
esm_c_s: number of channels in ESM hidden representation
c_s = 1024
c_z = 128
ESMFold(sequence)
s = ESM_hiddens(sequence) # num_layers x Length x esm_c_s
s = (softmax(layer_weights) * s).sum(0)
s = MLP(s)
z = PairwiseRelativePositionalEncoding(Length)
for b in folding_blocks:
    s, z = b(s, z)
return StructureModule(s, z)
```

We use a learned weighted sum of ESM embeddings to produce the initial hidden state into the model. This is then fed through an MLP. The initial pairwise state is simply the pairwise relative positional encoding described in (*26*). We found that using the attention maps initially gives a boost in performance, but this disappears during training. For experiments that do not use any folding blocks, we use an MLP applied to the ESM attention maps as input, and add the pairwise relative positional encoding to the attention map scores. Finally, the StructureModule parses these results into coordinates.

The predicted lDDT head is output from the hidden representation of the StructureModule. The predicted TM head uses the pairwise representation z. Finally, we also predict the distogram, from the same representation.

To predict complexes shown in **Figure 2D**, we give a residue index break of 1000 to ESMFold and link chains with a 25-residue poly-glycine linker, which we remove before displaying. Note that this is using ESMFold out of distribution since single chains are used during training.

1.4 Metagenomics experiments

MGnify (*29*) version 2022 was clustered at 50% sequence similarity using parameters corresponding to high sensitivity (--min-seq-id 0.5 --kmer-per-seq 100 --cluster-mode 2 --cov-mode 1 -c 0.6). Of these cluster representatives, we filtered out any sequences with less than 50 residues or greater than 512 residues, from which we obtained a uniformly sampled set of 1 million sequences. We then used ESMFold to obtain structure predictions and corresponding pLDDT values for each of these sequences.

We further analyzed the high-confidence subset of the 1 million structures with mean pLDDT greater than 0.9, corresponding to ~59k structures. For each high-confidence structure, we used Foldseek (*52*) easy-search (--alignment-type 1) to identify similar structures in the PDB (as of April 12, 2022) based on TM-score, with a cutoff of 0.5 (lower than which Foldseek would return no structures) to enable more efficient search. For high-confidence structures that also have no structures with TM-score greater than 0.5 returned by Foldseek, we used full AlphaFold2 with MSAs to also obtain structure predictions (we picked the top of five relaxed models ranked by mean pLDDT). We then compute RMSD values of aligned backbone coordinates and all-atom TM-score between the ESMFold- and AlphaFold2-predicted structures. For each sequence corresponding to a high-confidence structure, we also used blastp version 2.10.0+ to search for similar sequences in UniRef90 to compute sequence identity. We defined sequences as having low sequence identity to UniRef90 if all of the returned hits have an E-value greater than 1 or if the closest entry in UniRef90 has sequence identity of less than 30%. For sequences with no significant matches in UniRef90, we also used the jackhmmer web server (https://www.ebi.ac.uk/Tools/hmmer/search/jackhmmer) (*51*) to manually query four reference proteomes for similar sequences.

To construct the landscape of MGnify sequences, we first used ESM-1b to embed each sequence as a 1280-dimensional vector. These embeddings were then visualized using the umap version 0.5.3, scanpy version 1.9.1, and anndata 0.8.0 Python packages (*63*, *64*), where dimensionality reduction was applied directly to the embedding vectors (use_rep='X' in scanpy.tl.umap) with default parameters (15-nearest-neighbors graph via approximate Euclidean distance, UMAP min_dist=0.5). Highlighted structure predictions with low similarity to known structures were manually selected and are summarized in **Figures 4** and **fig. S2**. For these structures, we performed an additional structural similarity search using the Foldseek webserver (https://search.foldseek.com/search) with default parameters to identify the closest structures in PDB100 211201 beyond the TM-score cutoff of 0.5.