CONVOLUTIONS ARE COMPETITIVE WITH TRANSFORMERS FOR PROTEIN SEQUENCE PRETRAINING

Kevin K. Yang
Microsoft Research New England
yang.kevin@microsoft.com

Alex X. Lu
Microsoft Research New England
lualex@microsoft.com

Nicolo Fusi
Microsoft Research New England
fusi@microsoft.com

ABSTRACT

Pretrained protein sequence language models largely rely on the transformer architecture. However, transformer run-time and memory requirements scale quadratically with sequence length. We investigate the potential of a convolution-based architecture for protein sequence masked language model pretraining and subsequent finetuning. CNNs are competitive on the pretraining task with transformers across several orders of magnitude in parameter size while scaling linearly with sequence length. More importantly, CNNs are competitive with and occasionally superior to transformers across an extensive set of downstream evaluations, including structure prediction, zero-shot mutation effect prediction, and out-of-domain generalization. We also demonstrate strong performance on sequences longer than the positional embeddings allowed in the current state-of-the-art transformer protein masked language models. Finally, we close with a call to disentangle the effects of pretraining task and model architecture when studying pretrained protein sequence models.

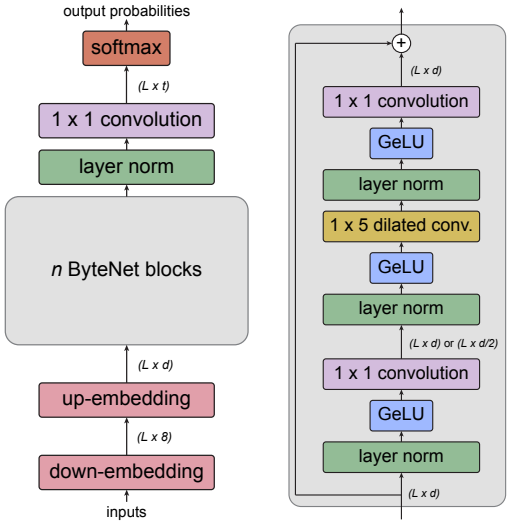
1 INTRODUCTION AND BACKGROUND

Large pretrained protein language models, largely relying on the attention-based transformer (Vaswani et al., 2017) architecture, have advanced the ability of machine-learning methods to predict protein structure and function from sequence, especially when labeled training data is sparse. Most modern self-supervised protein sequence pretraining combine a transformer model with either an autoregressive likelihood (Madani et al., 2020; 2021; Ferruz et al., 2022) or with the masked language modeling task introduced for natural language by BERT (bidirectional encoder representations from transformers) (Devlin et al., 2018), as in ESM (Rives et al., 2021), the transformers benchmarked in TAPE (Rao et al., 2019), and ProtTrans Elnaggar et al. (2021).

The primary drawback of transformers is that the compute and memory required by the attention layers scale quadratically with input sequence length. In addition, because transformer attention is invariant to input position, transformer sequence models usually use a positional encoding. These encodings can be difficult to extend past the maximum length seen during training. As a result, most pretrained protein transformer models limit the input length during pretraining; for example, ESM has a maximum input length of 1022 residues. Of the 42 million cluster representatives in the March 2020 release of UniRef50 (Suzek et al., 2015), 1.1 million, or 2.6%, are longer than 1022 residues. This includes many proteins of interest, such as the SARS-Cov-2 spike gylcoprotein and *Streptococcus pyogenes* CRISPR-associated endonuclease Cas9. Work in natural language processing (Kalchbrenner et al., 2016; Wu et al., 2019; Tay et al., 2021) hints that pretrained attention-free convolutional neural networks (CNNs) can be competitive with pretrained transformers while scaling linearly with sequence length. On protein sequences, Shin et al. (2021) train autoregressive convolutional models on protein families, but do not attempt to train a single model over the breadth of known protein diversity. Lu et al. (2020) use a small convolutional encoder for a noise-contrastive pretraining task on proteins, but do not give it global context or make the model autoencoding.

Bileschi et al. (2022) use a similar convolutional architecture to our model to learn functional annotations for unaligned protein sequences. However, their task is not autoencoding, and they do not consider performance on downstream tasks.

In order to investigate the efficacy of pretrained convolutions for proteins, we train protein sequence convolutional masked language models on the March 2020 release of UniRef50, which we refer to as CARP (**C**onvolutional **A**utoencoding **R**epresentations of **P**roteins). These are competitive with transformers across several orders of magnitude in parameter size on the pretraining task. The largest CARP, with approximately 640M learnable parameters (CARP-640M) is competitive with the current state-of-the-art transformer protein sequence masked language model, ESM (Rives et al., 2021; Meier et al., 2021) on a variety of downstream prediction tasks, including structure prediction, zero-shot mutation effect prediction, and out-of-domain generalization on biologically-relevant protein engineering datasets. Because CARP scales linearly in computation with the input sequence and does not rely on an input positional embedding, it is straightforward to apply it to sequences longer than the longest sequences allowed in training, which we demonstrate with zero-shot predictions of mutation and deletion effects in CRISPR-CAS9. These empirical results demonstrate a need to deepen our understanding of protein sequence pretraining by disentangling the effects of architecture and the pretraining task.



(a) CARP (b) ByteNet block

Figure 1: The CARP architecture.

2 Convolutional protein sequence mask language models

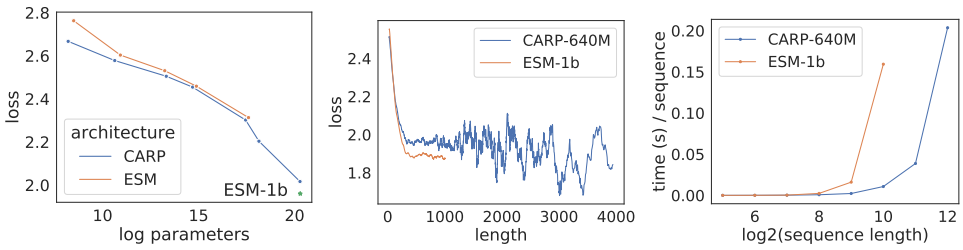
We pretrain CARP using the masked language model (MLM) objective described in Rives et al. (2021). Each sequence is corrupted by changing some tokens to a special mask token or another amino acid token, and the model is tasked with reconstructing the original sequence. Specifically, 15% of tokens from each sequence are randomly selected for supervision. For those 15% of tokens, 80% are replaced by the mask token, 10% are replaced by a randomly-chosen amino acid, and 10% remain unchanged. The model is trained to minimize the batch average cross entropy loss between its predictions for the selected tokens and the true tokens at those locations. We train on the cluster representatives from the March 2020 release of UniRef50, with approximately 83k sequences held out for validation and another 210k sequences held out for testing, leaving 41.5 million sequences for training.

CARP combines the ByteNet encoder dilated CNN architecture from Kalchbrenner et al. (2016) with simple input embedding and output decoding layers, as shown in Figure 1a. CARP begins with an embedding layer, which maps an input sequence of L tokens $x \in \mathbb{D}^L$ to an 8-dimensional intermediate embedding, followed by a linear mapping into the model dimension d: $e_0 \in \mathbb{R}^{L \times d}$.

This passes through a stack of n ByteNet dilated CNN blocks Figure 1b with residual connections in between followed by a final layer norm to produce the encoder representation $e_n \in \mathbb{R}^{L \times d}$, and finally a linear decoder maps this to the $L \times t$ logits, where t is the number of possible tokens. The 1×5 convolution layer in every ByteNet block is dilated and padded to preserve sequence length. Dilation increases the CNN perceptive field exponentially with the number of layers in order to obtain global context for long input sequences. The CNN dilation rate doubles every layer up to a maximum rate r (for our experiments $r = 128$). The scheme is repeated multiple times in the network, always starting from a dilation rate of 1.

We varied the number of parameters in CARP from approximately 3000 to 640 million by setting the model dimension d, setting the encoder hidden dimension h_e to either d or $\frac{d}{2}$, and setting the number of layers. For comparison, we also trained transformer models with comparable numbers of parameters using the ESM-1b architecture described in Rives et al. (2021) on our UniRef50 dataset. Our largest model, CARP-640M, has a test loss of 2.02, comparable to ESM-1b, which has 650 million parameters and a loss of 1.96 on its test set. Note that ESM-1b was trained and tested on an earlier version of UniRef50 with different train/test splits than CARP or our ESM models. (Throughout, ESM-1b refers specifically to the 650-million parameter transformer trained on the March 2018 UniRef50 release and described in Rives et al. (2021), while ESM refers to our small transformer masked language models based off of the ESM-1b architecture. Likewise, CARP refers to any ByteNet masked language model, while CARP-X refers to the model with approximately X parameters.)

As shown in Figure 2a, CARP's performance on the pretraining task is comparable to ESM's across several orders of magnitude of variation in the number of parameters when using the same pretraining dataset. Figure 2b shows MLM loss by length for CARP-640M and ESM-1b on their respective test sets, smoothed with a window of 30 in the length dimension. For both models, the pretraining loss improves quickly until the sequence length reaches about 500, and then slowly thereafter. The maximum input length for ESM-1b is 1022, but we calculate losses for CARP-640M for sequences with up to 4096 residues. As expected and shown in Figure 2c, CARP-640M scales much better with input sequence length than ESM-1b. These results show that convolutions can perform protein sequence masked language modeling comparably to tranformers without suffering from a quadratic dependence between runtime and sequence length.



(a) MLM loss vs size (b) loss vs length (smoothed) (c) runtime vs length

Figure 2: Comparisons between CARP and the ESM-1b transformer.

3 Downstream Tasks

One goal of protein MLMs is to encode information in their output representation or model weights that improves performance on downstream prediction tasks. Downstream evaluation can be zero-shot (without access to labels for further training), the pretrained model can be frozen and a small neural network decoder can be trained to predict labels from the pretrained model's output representations (pretrain-freeze), or the new decoder and pretrained model can be finetuned together (pretrain-finetune). We use the output from the final layer norm in Figure 1a as the output representation. Unless otherwise noted, the new decoder consists of a learned attention that converts the output from $L \times d$ to d followed by a 2-layer neural network with hidden size d. For tasks with labels, we evaluate both pretrain-freeze and pretrain-finetune and compare to ESM-1b or ESM-1v. Where relevant, we also compare the CARP architecture with randomly-initialized weights (naive-freeze

and naive-finetune), linear ridge regression, and the small CNN described in Dallago et al. (2021) and Shanehsazzadeh et al. (2020).

3.1 Protein structure

One of the most striking successes of protein MLMs is their ability to encode structural information without access to structural labels during pretraining. We evaluate CARP-640M's ability to encode structural information through 2 tasks:

1. **Remote homology detection** asks a model to detect structural similarity across distantly-related sequences. We evaluate accuracy on the fold-level holdout set from TAPE.
2. **3-class secondary structure prediction** asks a model to predict whether each residue in a protein is part of a helix, strand, or other. We use the training and validation sets from TAPE and evaluate accuracy on the CB513 test set. For this task, we train a neural network consisting of two CNN layers, an LSTM, and a linear head on top of the pretrained model, as described in Rives et al. (2021).

As shown in Table 1, pretraining improves performance for structure prediction tasks, and CARP-640M is competitive with ESM-1b on both tasks. These results show that pretrained convolutions learn structural information.

Table 1: Structure prediction tasks. Values for ESM-1b are taken from Rives et al. (2021).

		Task	
Method	Model	remote homology	secondary structure
pretrain-freeze	CARP-640M	0.25	**0.83**
	ESM-1b	-	**0.82**
pretrain-finetune	CARP-640M	0.29	**0.83**
	ESM-1b	**0.33**	-
naive-freeze	CARP-640M	0.09	0.66
	ESM-1b	-	-
naive-finetune	CARP-640M	0.09	0.71
	ESM-1b	-	-

3.2 Zero-shot mutation effect prediction

Large language models can predict experimental measurements of protein function without further training on sequence-fitness measurements or sets of evolutionarily-related sequences Hie et al. (2022); Meier et al. (2021). Following Meier et al. (2021), we score CARP-640M on 41 deep mutational scanning datasets originally compiled by Riesselman et al. (2018). These datasets measure the effects of thousands of mutations or combinations of mutations to a parent sequence. We score sequences by masking every mutated position and computing the log odds ratio between the mutated and wild-type residues at each mutated position, assuming an additive model when a sequence contains multiple mutations:

$$\sum_{p \in P} \log p(x_P^{\mathrm{mt}} | x_{\setminus P}^{\mathrm{wt}}) - \log p(x_P^{\mathrm{wt}} | x_{\setminus P}^{\mathrm{wt}}) \tag{1}$$

where P indicates the mutated positions.

Figure 3 compares zero-shot performance for CARP-640M, ESM-1b, ESM-1v, position-specific scoring matrices (PSSM), and ProtBert-BFD Elnaggar et al. (2021). ESM-1v uses the ESM-1b transformer architecture but is trained on UniRef90 instead of UniRef50, and the results are reported for an ensemble of 5 models. ProtBERT-BFD is a 420 million-parameter transformer masked language model trained on 2 billion protein sequences from BFD (Steinegger & Söding, 2018; Steinegger et al., 2019). Averaged across the 41 datasets, CARP-640M has a Spearman correlation of 0.49, compared to 0.46 for ESM-1b, 0.51 for ESM-1v, 0.46 for PSSM, and 0.43 for ProtBERT-BFD. CARP-640M outperforms ESM-1b on 23 out of 41 datasets, ESM-1v on 18 out of 41 datasets, PSSM

on 26 out of 41 datasets, and ProtBERT-BFD on 25 out of 41 datasets. Meier et al. (2021) found that using the full UniProt sequences instead of only the sequence of the mutated domain results in better zero-shot predictions. However, this is not always possible with ESM-1x, as some UniProt sequences for these proteins are longer than 1022 residues. As a further proof of concept, we made zero-shot predictions for the effects of mutations in Cas9 from *Streptococcus pyogenes* (Spencer & Zhang, 2017), which is 1368 residues long, and obtain a Spearman correlation of 0.26. These results show that pretrained convolutions can make zero-shot predictions of protein mutation effects on fitness, including on sequences longer than allowed by ESM-1x.

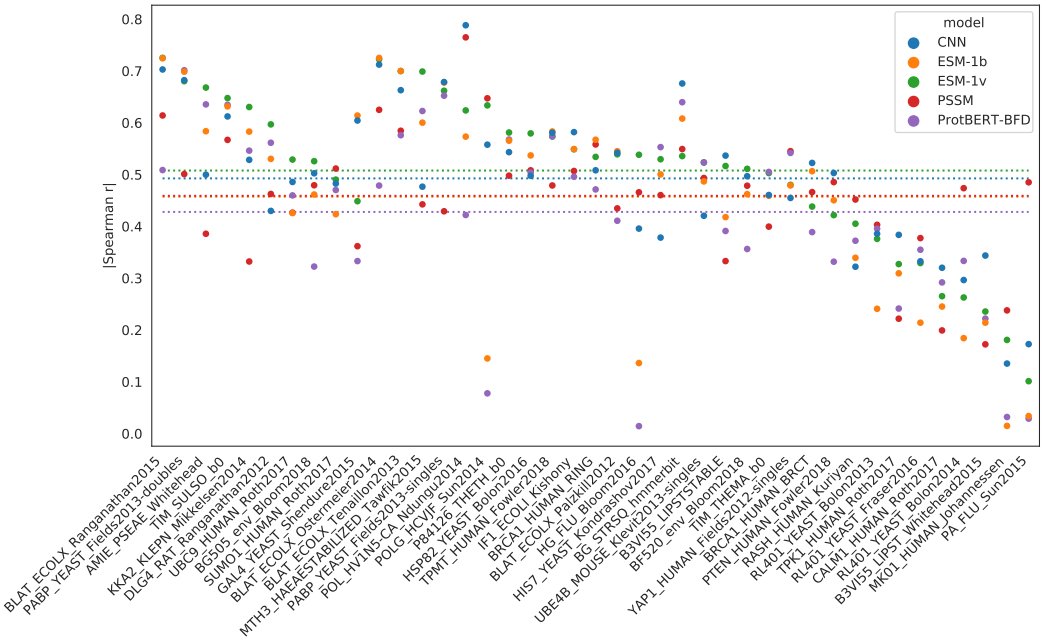


Figure 3: Zero-shot protein fitness prediction. Comparison across 41 deep mutational scanning datasets from DeepSequence. Points are Spearman correlation on each dataset. Horizontal lines show the average Spearman correlation across the datasets. Values for ESM-1b, ESM-1v, PSSM, and ProtBERT-BFD are taken from Meier et al. (2021).

3.3 Out-of-domain fitness prediction

Another motivation for pretrained protein sequence models is that they may be able to generalize after fine-tuning in ways that are helpful for protein engineering. For example, a protein engineer may want to train a model on single mutants and make predictions for sequences with multiple mutations, or train a model that is accurate for sequences with fitness greater than what is seen in the training set. Here, we evaluate CARP-640M on tasks from the GB1 (Wu et al., 2016) (Table 2 and AAV (Bryant et al., 2021) (Table 3 landscapes from FLIP (Dallago et al., 2021). We compare results to ESM-1b with frozen weights and the same decoder neural network, linear ridge regression, and a small CNN.

In general, pretraining improves CARP-640M's performance on these tasks, and fine-tuning the entire model outperforms freezing the pretrained weights. Comparisons to the baselines show that pretraining is most helpful when generalizing from single mutants to multiple. CARP-640M outperforms ESM-1b on generalizing from few mutations to more, but ESM-1b is better at generalizing from a low-fitness training to higher-fitness sequences. These results show that pretrained convolutions help generalization to types of sequence variation not seen during training.

3.4 In-domain property prediction

Finally, we consider fitness-prediction tasks that do not require difficult biological generalization (Table 4). We evaluate on three sequence-fitness regression tasks:

Table 2: Performance (Spearman correlation) on the FLIP AAV tasks. Values for models other than CARP-640M are taken from FLIP.

		Task		
Method	Model	1-vs-many	2-vs-many	low-vs-high
pretrain-freeze	CARP-640M	0.73	0.67	0.29
	ESM-1b	0.03	0.65	**0.39**
pretrain-finetune	CARP-640M	**0.76**	**0.79**	0.14
naive-freeze	CARP-640M	0.46	0.43	0.18
	ESM-1b	0.18	0.22	0.08
naive-finetune	CARP-640M	0.27	0.62	0.03
baseline	ridge	0.22	0.03	0.12
	CNN	0.48	0.74	0.34

Table 3: Performance (Spearman correlation) on the FLIP GB1 tasks. Values for models other than CARP-640M are taken from FLIP.

		Task			
Method	Model	1-vs-many	2-vs-many	3-vs-many	low-vs-high
pretrain-freeze	CARP-640M	**0.33**	0.45	0.69	0.12
	ESM-1b	0.28	0.55	0.79	**0.59**
pretrain-finetune	CARP-640M	**0.33**	**0.74**	**0.88**	0.42
naive-freeze	CARP-640M	-0.04	0.45	0.70	0.37
	ESM-1b	0.06	0.06	0.48	0.23
naive-finetune	CARP-640M	0.22	0.08	**0.88**	0.39
baseline	ridge	0.28	0.32	0.76	0.51
	CNN	0.17	0.32	0.83	0.51

1. **Fluorescence** requires the model to predict the effect of one or more mutations on the brightness of green fluorescent protein. The data was originally collected by Sarkisyan et al. (2016). We use the data splits provided in TAPE.
2. **Stability** requires the model to predict a small protein's resistance to protease degradation. The data was originally collected by Rocklin et al. (2017). We use the data splits provided in TAPE.
3. **Meltome-mixed** requires the model to predict the melting temperature of a variety of proteins from across the domains of life. The data was originally collected by Jarzab et al. (2020). We use the cluster representatives and data splits provided in FLIP.

in addition to two intrinsically-disordered region (IDR) function classification tasks taken from Zarin et al. (2021). For the IDR datasets, we use MMseqs2 (Steinegger & Söding, 2017) to cluster sequences to 50% identity and then randomly assign clusters to training, validation, or testing.

1. **Cdc28 binding** requires the model to predict whether an IDR is a target of Cdc28.
2. **Mitochondria targeting** requires the model to predict whether an IDR targets its protein for transport into the mitochondria.

In general, while pretraining improves CARP-640M's performance on these tasks, neither of the large pretrained models consistently out-perform the baseline models on these tasks. However, CARP-640M is generally comparable to ESM-1b, showing that once again convolutions are comparable to attention.

4 Discussion

We have shown that convolutions can be comparable to or superior to transformers on both the MLM pretraining task and a variety of downstream protein sequence modeling tasks, and that convolutions, like transformers, benefit from pretraining. Unlike transformers, convolutions scale linearly with input

Table 4: Performance on property prediction tasks. For fluorescence, stability, and meltome-mixed, values reported are Spearman correlation. For the IDR tasks, values reported are area under the reciever operating curve. Values for ESM-1b on fluorescence and stability are taken from Rives et al. (2021). Values for baselines on fluorescence and stability are taken from FLIP.

		Task				
Method	Model	fluorescence	stability	meltome-mixed	Cdc28	mito.
pretrain-freeze	CARP-640M	0.60	0.63	0.54	0.83	0.87
	ESM-1b	-	-	**0.68**	**0.91**	**0.91**
pretrain-finetune	CARP-640M	**0.68**	**0.72**	0.53	0.90	0.88
	ESM-1b	**0.68**	0.71	-	0.87	0.87
naive-freeze	CARP-640M	0.62	0.34	0.29	0.83	0.85
	ESM-1b	-	-	0.44	0.87	0.84
naive-finetune	CARP-640M	0.52	0.65	0.30	0.82	0.89
	ESM-1b	-	-	-	0.81	0.82
baseline	ridge	**0.68**	0.48	0.17	0.52	0.53
	CNN	0.67	0.51	0.34	0.87	0.85

sequence length, which becomes important when modeling long protein sequences. Work in natural language processing has also shown that convolutions can require fewer FLOPs of compute than transformers, even for short sequences (Tay et al., 2021). In addition, while we use standard dilated convolutions, there are more efficient convolution variants designed for sequence modeling (Wu et al., 2019) that may further improve model speed.

However, convolutions may not be competitive with transformers on tasks where a cross- or self-attention inductive bias is explicitly needed or desired for interpretability. For example, it is possible to extract structural contact maps from pretrained transformer self-attention matrices (Rao et al., 2020), and self-attention matrices contain information about binding sites (Vig et al., 2020) – convolutions lack an obvious equivalent. In addition, it is more natural to extend attention-based models to predict protein-protein interaction sites. The transformer's quadratic dependence on sequence length can also be ameliorated with approximate attention methods (Child et al., 2019; Beltagy et al., 2020; Kitaev et al., 2020; Tay et al., 2020a; Wang et al., 2020; Zaheer et al., 2020; Katharopoulos et al., 2020; Choromanski et al., 2020b), but the choice of approximation matters for performance and the best method is not always clear *a priori* (Tay et al., 2020b). On proteins, Choromanski et al. (2020b) and Choromanski et al. (2020a) show that Performer approximate attention can perform well for autoregressive and masked protein language models, respectively, while ProteinBERT combines a fast global attention mechanism with masked language and functional annotation prediction pretraining (Brandes et al., 2021).

Currently, pretrained protein language models are tightly-coupled to the transformer architecture, and the effects of the pretraining task can be conflated with the effects of the pretrained architecture. Our pretrained convolutional models may provide complementary inductive biases those found in pretrained transformer models, making them useful alternatives for practitioners. Finally, we hope that this work is the first step in investigating the independent and interaction effects of pretraining and architecture for protein sequence modeling. While we evaluate the effects of masked language model pretraining, transformers have also been used for autoregressive language model pretraining (Madani et al., 2020) and pairwise masked language modeling (He et al., 2021), and combining structural information (Mansoor et al., 2021; Zhang et al., 2022) or functional annotations (Brandes et al., 2021) offers further directions for protein pretraining tasks.

REFERENCES

Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

Maxwell L Bileschi, David Belanger, Drew H Bryant, Theo Sanderson, Brandon Carter, D Sculley, Alex Bateman, Mark A DePristo, and Lucy J Colwell. Using deep learning to annotate the protein universe. *Nature Biotechnology*, pp. 1–6, 2022.

Nadav Brandes, Dan Ofer, Yam Peleg, Nadav Rappoport, and Michal Linial. ProteinBERT: A universal deep-learning model of protein sequence and function. *bioRxiv*, 2021.

Drew H Bryant, Ali Bashir, Sam Sinai, Nina K Jain, Pierce J Ogden, Patrick F Riley, George M Church, Lucy J Colwell, and Eric D Kelsic. Deep diversification of an AAV capsid protein by machine learning. *Nature Biotechnology*, 39(6):691–696, 2021.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, David Belanger, Lucy Colwell, et al. Masked language modeling for proteins via linearly scalable long-context transformers. *arXiv preprint arXiv:2006.03555*, 2020a.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*, 2020b.

Christian Dallago, Jody Mou, Kadina E Johnston, Bruce Wittmann, Nick Bhattacharya, Samuel Goldman, Ali Madani, and Kevin K Yang. FLIP: Benchmark tasks in fitness landscape inference for proteins. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv*, 2018.

Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, Debsindhu Bhowmik, and Burkhard Rost. ProtTrans: Towards cracking the language of life's code through Self-Supervised learning. May 2021.

Noelia Ferruz, Steffen Schmidt, and Birte Höcker. A deep unsupervised language model for protein design. *bioRxiv*, 2022.

Liang He, Shizhuo Zhang, Lijun Wu, Huanhuan Xia, Fusong Ju, He Zhang, Siyuan Liu, Yingce Xia, Jianwei Zhu, Pan Deng, et al. Pre-training co-evolutionary protein representation via a pairwise masked language model. *arXiv preprint arXiv:2110.15527*, 2021.

Brian L. Hie, Kevin K. Yang, and Peter S. Kim. Evolutionary velocity with protein language models predicts evolutionary dynamics of diverse proteins. *Cell Systems*, 2 2022. ISSN 24054712. doi: 10.1016/j.cels.2022.01.003.

Anna Jarzab, Nils Kurzawa, Thomas Hopf, Matthias Moerch, Jana Zecha, Niels Leijten, Yangyang Bian, Eva Musiol, Melanie Maschberger, Gabriele Stoehr, et al. Meltome atlas—thermal proteome stability across the tree of life. *Nature methods*, 17(5):495–503, 2020.

Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aaron van den Oord, Alex Graves, and Koray Kavukcuoglu. Neural machine translation in linear time. *arXiv preprint arXiv:1610.10099*, 2016.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning*, pp. 5156–5165. PMLR, 2020.

Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*, 2020.

Amy X Lu, Haoran Zhang, Marzyeh Ghassemi, and Alan Moses. Self-Supervised contrastive learning of protein representations by mutual information maximization. November 2020.

Ali Madani, Bryan McCann, Nikhil Naik, Nitish Shirish Keskar, Namrata Anand, Raphael R Eguchi, Po-Ssu Huang, and Richard Socher. ProGen: Language modeling for protein generation. *arXiv*, 2020.

Ali Madani, Ben Krause, Eric R Greene, Subu Subramanian, Benjamin P Mohr, James M Holton, Jose Luis Olmos, Caiming Xiong, Zachary Z Sun, Richard Socher, et al. Deep neural language modeling enables functional protein generation across families. *bioRxiv*, 2021.

Sanaa Mansoor, Minkyung Baek, Umesh Madan, and Eric Horvitz. Toward more general embeddings for protein design: Harnessing joint representations of sequence and structure. *bioRxiv*, 2021.

Joshua Meier, Roshan Rao, Robert Verkuil, Jason Liu, Tom Sercu, and Alexander Rives. Language models enable zero-shot prediction of the effects of mutations on protein function. In M. Ranzato, A. Beygelzimer, K. Nguyen, P.S. Liang, J.W. Vaughan, and Y. Dauphin (eds.), *Advances in Neural Information Processing Systems 34*, 2021.

Roshan Rao, Nicholas Bhattacharya, Neil Thomas, Yan Duan, Peter Chen, John Canny, Pieter Abbeel, and Yun Song. Evaluating protein transfer learning with TAPE. In *Advances in Neural Information Processing Systems*, pp. 9686–9698, 2019.

Roshan Rao, Joshua Meier, Tom Sercu, Sergey Ovchinnikov, and Alexander Rives. Transformer protein language models are unsupervised structure learners. In *International Conference on Learning Representations*, 2020.

Adam J Riesselman, John B Ingraham, and Debora S Marks. Deep generative models of genetic variation capture the effects of mutations. *Nature Methods*, 15(10):816–822, 2018.

Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C Lawrence Zitnick, Jerry Ma, and Rob Fergus. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proc. Natl. Acad. Sci. U. S. A.*, 118(15), April 2021.

Gabriel J Rocklin, Tamuka M Chidyausiku, Inna Goreshnik, Alex Ford, Scott Houliston, Alexander Lemak, Lauren Carter, Rashmi Ravichandran, Vikram K Mulligan, Aaron Chevalier, et al. Global analysis of protein folding using massively parallel design, synthesis, and testing. *Science*, 357 (6347):168–175, 2017. doi: 10.1126/science.aan0693.

Karen S Sarkisyan, Dmitry A Bolotin, Margarita V Meer, Dinara R Usmanova, Alexander S Mishin, George V Sharonov, Dmitry N Ivankov, Nina G Bozhanova, Mikhail S Baranov, Onuralp Soylemez, Natalya S Bogatyreva, Peter K Vlasov, Evgeny S Egorov, Maria D Logacheva, Alexey S Kondrashov, Dmitry M Chudakov, Ekaterina V Putintseva, Ilgar Z Mamedov, Dan S Tawfik, Konstantin A Lukyanov, and Fyodor A Kondrashov. Local fitness landscape of the green fluorescent protein. *Nature*, 533(7603):397, 2016. doi: 10.1038/nature17995.

Amir Shanehsazzadeh, David Belanger, and David Dohan. Is transfer learning necessary for protein landscape prediction? *arXiv preprint arXiv:2011.03443*, 2020.

Jung-Eun Shin, Adam J Riesselman, Aaron W Kollasch, Conor McMahon, Elana Simon, Chris Sander, Aashish Manglik, Andrew C Kruse, and Debora S Marks. Protein design and variant prediction using autoregressive generative models. *Nature communications*, 12(1):1–11, 2021.

Jeffrey M Spencer and Xiaoliu Zhang. Deep mutational scanning of *S. pyogenes* Cas9 reveals important functional domains. *Scientific reports*, 7(1):1–14, 2017.

Martin Steinegger and Johannes Söding. MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature biotechnology*, 35(11):1026–1028, 2017.

Martin Steinegger and Johannes Söding. Clustering huge protein sequence sets in linear time. *Nature communications*, 9(1):1–8, 2018.

Martin Steinegger, Milot Mirdita, and Johannes Söding. Protein-level assembly increases protein sequence recovery from metagenomic samples manyfold. *Nature methods*, 16(7):603–606, 2019.

Baris E Suzek, Yuqi Wang, Hongzhan Huang, Peter B McGarvey, Cathy H Wu, and UniProt Consortium. UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics*, 31(6):926–932, 2015.

Yi Tay, Dara Bahri, Liu Yang, Donald Metzler, and Da-Cheng Juan. Sparse sinkhorn attention. In *International Conference on Machine Learning*, pp. 9438–9447. PMLR, 2020a.

Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. Long range arena: A benchmark for efficient transformers. *arXiv preprint arXiv:2011.04006*, 2020b.

Yi Tay, Mostafa Dehghani, Jai Gupta, Dara Bahri, Vamsi Aribandi, Zhen Qin, and Donald Metzler. Are pre-trained convolutions better than pre-trained transformers? *arXiv preprint arXiv:2105.03322*, 2021.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pp. 5998–6008, 2017.

Jesse Vig, Ali Madani, Lav R Varshney, Caiming Xiong, Richard Socher, and Nazneen Fatema Rajani. Bertology meets biology: Interpreting attention in protein language models. *arXiv preprint arXiv:2006.15222*, 2020.

Sinong Wang, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020.

Felix Wu, Angela Fan, Alexei Baevski, Yann N Dauphin, and Michael Auli. Pay less attention with lightweight and dynamic convolutions. *arXiv preprint arXiv:1901.10430*, 2019.

Nicholas C Wu, Lei Dai, C Anders Olson, James O Lloyd-Smith, and Ren Sun. Adaptation in protein fitness landscapes is facilitated by indirect paths. *Elife*, 5:e16965, 2016. doi: 10.7554/eLife.16965.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. Big bird: Transformers for longer sequences. *Advances in Neural Information Processing Systems*, 33:17283–17297, 2020.

Taraneh Zarin, Bob Strome, Gang Peng, Iva Pritišanac, Julie D Forman-Kay, and Alan M Moses. Identifying molecular features that are associated with biological function of intrinsically disordered protein regions. *Elife*, 10:e60220, 2021.

Zuobai Zhang, Minghao Xu, Arian Jamasb, Vijil Chenthamarakshan, Aurelie Lozano, Payel Das, and Jian Tang. Protein structure representation learning by geometric pretraining. 2022.