SINGLE LAYERS OF ATTENTION SUFFICE TO PREDICT PROTEIN CONTACTS

Anonymous authors
Paper under double-blind review

ABSTRACT

The established approach to protein contact prediction frames the task as one of graph selection, extracting contacts by estimating the parameters of a Potts model. Another approach has recently appeared which leverages large pretrained Transformers, producing contacts by combining attention maps from various heads. In this work, we provide evidence that these approaches are not as different as they initially seem by providing a theoretical connection between attention and Potts models. To do so, we introduce a simplified attention model called *factored attention*. On the one hand, factored attention is a direct simplification of multihead scaled dot-product attention in the Transformer. On the other hand, factored attention defines a valid Pairwise Markov Random Field and includes Potts models as a sparse special case. Examining factored attention allows us to explore the relative merits of each model class when learning contacts from an aligned protein family. We go on to empirically assess the performance of factored attention by training on a wide range of alignments of individual protein families. We further compare to a large pretrained Transformer trained on a corpus of unaligned protein sequences. We find that a single layer of attention is comparable to state-of-the-art Potts model at contact prediction. Taken together, these results provide motivation for training Transformers on large protein datasets.

1 INTRODUCTION

Inferring protein structure from sequence is a longstanding problem in computational biochemistry. State of the art supervised models (Senior et al., 2020; Yang et al., 2019; Wang et al., 2017) take as input protein features extracted via unsupervised methods, and depend critically on the quality of this featurization. Markov Random Field (MRF) models are the predominant unsupervised method for modeling interactions between amino acids, and are trained to maximize a pseudolikelihood loss on a family of evolutionarily related proteins (Balakrishnan et al., 2011; Seemayer et al., 2014).

Inspired by the success of BERT (Devlin et al., 2018), GPT (Brown et al., 2020) and related unsupervised models in NLP, a line of work has emerged that learns features of proteins to compete with MRF models (Rives et al., 2020; Elnaggar et al., 2020; Rao et al., 2019; Madani et al., 2020; Nambiar et al., 2020). This new approach trains Transformer models on large datasets of protein sequences. There are three major differences between this and the MRF-based approach. First, the MRF approach trains a separate model for each cluster of protein sequences by querying a database for highly related sequences, while the Transformer approach trains a single model on an entire database. Second, the MRF approach takes as input aligned sequences, which are the output of a separate alignment algorithm (Durbin et al., 1998). The Transformer approach, on the other hand, uses *unaligned* sequences and must therefore learn an alignment in an unsupervised manner. Third, the MRF approach uses a simple linear model of interactions, while the Transformer approach uses a nonlinear deep neural network. The multiple layers of the Transformer prevent the use of pseudolikelihood for training, so Transformers are trained via masked language modeling.

Our work draws theoretical connections between the loss function and architecture of the MRF approach and the Transformer approach. We introduce two simplified models of attention as tools for exploring the relationship between MRFs and Transformers. We call these models *single-layer* and *factored* attention. Single-layer attention is a single layer of the Transformer attention operation. Factored attention is a further simplification of single-layer attention which processes the position

and sequence of inputs separately, and defines an MRF. We show that leveraging the evolutionary information encoded in a Multiple Sequence Alignment (MSA) to encode the position of input sequences enables simplified attention models to learn good-quality contacts, suggestive of their capacity to compete with MRF approaches.

To further highlight connections to the Transformer, we adapt the approach of Vig et al. (2020) to extract contacts directly from large pretrained Transformers. We compare contacts extracted from the ProtBERT-BFD (Elnaggar et al., 2020) Transformer to those extracted from MRF and simplified attention models. We show that each model class produces good contacts and that ProtBERT-BFD is competitive with state-of-the-art MRF models for contact prediction as well. Taken together, our results provide theoretical and empirical motivation for learning contacts via pretrained Transformers.

2 BACKGROUND

Proteins are polymers composed of amino acids and are commonly represented as strings.[1] Along with this 1D sequence representation, each protein folds into a 3D physical structure. Interactions between positions in a protein are defined by physical distance, rather than proximity in sequence and can be represented via a 2D symmetric matrix C, where entry C_{ij} is a measure of the 3D distance between positions i and j in the sequence. $C_{ij} = 1$ if i and j are within 8Å of one another, and 0 otherwise. C is commonly referred to as a contact map.

Multiple Sequence Alignments The most common method to determine a protein's structure from its sequence involves inferring interactions between positions in the sequence using statistical patterns in the sequence's evolutionary relatives. A set of related sequences is referred to as a protein *family*, and they are stored in a data structure which preserves evolutionary relationships. This data structure is known as a Multiple Sequence Alignment (MSA), (See Figure 7 for schematic). Gaps in aligned sequences correspond to insertions from an alignment algorithm (Johnson et al., 2010; Remmert et al., 2012), which seeks to minimize a Levenshtein edit distance-like metric among all the related sequences. These insertions ensure that amino acids with similar structure and function line up for all members of the family. With such an alignment, sequence position carries significant evolutionary, structural, and functional information. See Appendix A.1 for more information.

Coevolutionary Analysis of Protein Families The observation that statistical patterns in MSAs can be used to predict couplings has been used by multiple groups to infer structure and function from protein families (Korber et al., 1993; Göbel et al., 1994; Lapedes et al., 1999; Lockless & Ranganathan, 1999; Weigt et al., 2009; Fatakia et al., 2009; Fodor & Aldrich, 2004). Suppose we view the amino acid at position i in a sequence as a random variable x_i. Then we expect the mutual information $MI(x_i; x_j)$ between two non-interacting positions to be approximately zero. On the other hand, if $MI(x_i; x_j) > 0$ it suggests an interaction between positions which in turn suggests that the positions may be in contact or otherwise functionally related. The main challenge in estimating contacts from mutual information is to differentiate coupling that arises from direct contact from coupling that arises for other reasons such as transitivity ((i, k) coupled, (k, j) coupled, but (i, j) not in contact) and evolutionary relatedness of sequences (Weigt et al., 2009). The current standard approach is to model interactions between positions via a pairwise Markov Random Field (MRF) whose parameters are fit via regularized pseudolikelihood maximization (Balakrishnan et al., 2011).

Attention-Based Protein Models Multiple groups have proposed using attention-based architectures for protein modeling. Elnaggar et al. (2020); Rives et al. (2020); Rao et al. (2019); Madani et al. (2020) train large unsupervised attention-based models on millions of billions of protein sequences. These groups have shown that pretraining can produce good representations, but often still fall short of MRF based approaches for structure prediction. In cases where the protein structure is known, Ingraham et al. (2019) and Du et al. (2019) have used sparse attention based on the physical structure of the protein to directly encode priors about interaction. Our work compares a Transformer trained by Elnaggar et al. (2020) (ProtBERT-BFD) to single-layer attention models.

[1] Typically an alphabet of size 20 is used. Our schematic in Figure 1 shows proteins with an alphabet of size 3.

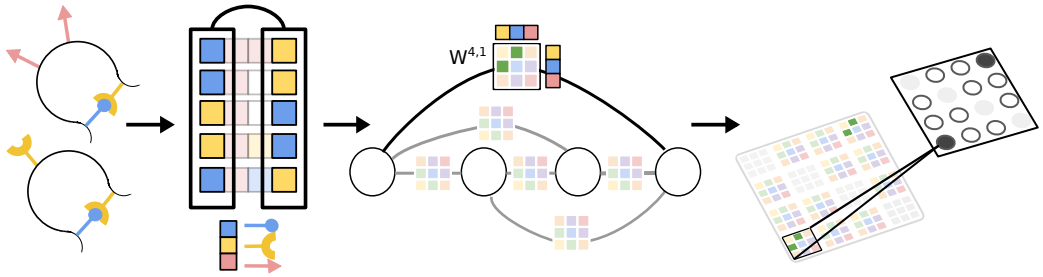


Figure 1: Training and interpretation of a Potts model on a single protein family. In this case, the proteins are all loops formed by one blue and one yellow amino acid locking together. The MSA for this family aligns these critical yellow and blue amino acids. For the trained MRF on this MSA, the weight matrix $W^{4,1}$ has the highest values due to the evolutionary constraint that blue and yellow covary for those positions. To extract predicted contacts from this Potts model, we take the Frobenius norm and see that the corners of the 4×4 contact map have highest confidence. In this case, the predicted contacts recapitulate the true ones.

3 Methods

Throughout this section, $x = (x_1, \ldots, x_L)$ is a sequence of length L from an alphabet of size A. This sequence is part of an MSA of length L with N total sequences. For each model, further details are provided in Appendix A.2.

3.1 Potts Models

A Potts model is a fully-connected Pairwise MRF with L nodes in which a node represents a position in the MSA. each edge (i, j) is parametrized with a matrix $W^{ij} \in \mathbb{R}^{A \times A}$. These matrices are organized into a 4D tensor which, together with a first-order potential $b \in \mathbb{R}^{L \times A}$, forms the parameters of a Potts model. Note that $W^{ij} = W^{ji}$.

Given an MSA, the parameters of this model are estimated through pseudolikelihood maximization (Kamisetty et al., 2013; Ekeberg et al., 2013). An individual full conditional $p_\theta(x_i|x_{\setminus i})$ is

$$p(x_i|x_{\setminus i}) = \text{softmax}\left(\sum_{j \neq i} W^{ij}(x_i, x_j) + b^i(x_i)\right). \tag{1}$$

Predicting contacts from Potts: We extract a predicted contact map $\widehat{C} \in \mathbb{R}^{L \times L}$ from the interaction tensor W. We follow Ovchinnikov et al. (2014) and set $\widehat{C}_{ij} = \|W^{ij}\|_F^2$. As described in the Appendix A.2.1, state-of-the-art Potts model implementations penalize the ℓ_2 norm of W and b during training. Empirically, this leads to artefacts in the outputs $\widehat{C}$ which are removed with the Average Product Correction (APC) (Dunn et al., 2008).

3.2 Attention Models

Single Attention Layer Given a discrete input x, the first layer of a Transformer produces a continuous representation with hidden size p, $y \in \mathbb{R}^{L \times p}$. Each position y_i is a function of all input positions $x_1, \ldots x_L$. Interactions between positions x_i and x_j are computed using dot-products of dense vectors Q_i and K_j. Multiple independent sets of such dot-products (referred to as *heads*) are combined to capture various types of interactions.

To compute the output y, we first augment x with positional information. This is done using a sequence embedding $E_{seq} \in \mathbb{R}^{A \times e}$ and a position embedding $E_{pos} \in \mathbb{R}^{L \times e}$, where the embedding size e is a hyperparameter. The augmented $\tilde{x} \in \mathbb{R}^{L \times e}$ is given by $\tilde{x}_i = E_{seq}(x_i) + E_{pos}(i)$.

Next, this layer computes interactions between all pairs of positions using dot products of vectors. Matrices $W_K, W_Q \in \mathbb{R}^{e \times d}$ are applied to obtain $Q = \tilde{x}W_Q$ and $K = \tilde{x}W_K$, called "queries" and

"keys," respectively. The inner product between rows of Q and K are interpreted as interaction affinities, yielding an interaction matrix $QK^T \in \mathbb{R}^{L\times L}$. To encourage sparsity and control the magnitude of the matrix, rescaling and row-wise softmax are applied to obtain the *attention map* $\mathcal{A} = \mathrm{softmax}(\alpha QK^T)$. Usually α is set to $1/\sqrt{d}$.

Finally, this attention map $\mathcal{A}$ is combined with a representation of the content at each position computed as $V = \tilde{x}W_V$, where the matrix $W_V \in \mathbb{R}^{e\times d}$. These are referred to as the "values". The output $y = \mathcal{A}V \in \mathbb{R}^{L\times d}$.

Many triples (W_Q, W_K, W_V) are used in order to capture distinct classes of interactions. Each triple is referred to as a head. For a model with H heads, the output representations y^h for each head h are concatenated, producing $y \in \mathbb{R}^{L\times d\cdot H}$. To train a single layer of attention with masked language modeling, we must produce logits $v \in \mathbb{R}^{L\times A}$. We use a dense layer $D \in \mathbb{R}^{H\cdot d\times e}$ which produces $D^h V_i^h \in \mathbb{R}^A$ for all positions i and heads h. Putting it all together, the full conditional analogous to Equation 1 is

$$p(x_i|x_{\setminus i}) = \mathrm{softmax}\left(\sum_{h=1}^{H}\sum_{j\neq i}\mathrm{softmax}\left(\frac{Q^h K^{h,T}}{\sqrt{d}}\right)_{ij}(D^h V^h)_j\right) \tag{2}$$

Predicting contacts from single attention layer: Each attention head computes an $L \times L$ attention matrix $\mathcal{A}^h = \mathrm{softmax}\left(\frac{Q^h K^{h,T}}{\sqrt{d}}\right)$. This can naturally be interpreted as a predicted contact map for the sequence x. Since Potts models produce contact maps summarizing an entire family, rather than a specific sequence, we use an alternate approach to contact extraction for single attention layers. Instead, we treat the positional embedding E_{pos} as an overall summary per-position information in the protein family. This leverages the fact that a single layer model is trained on a single family. Feeding in this modified input, we obtain attention maps $\mathcal{A}^h$. These H matrices are averaged to produce an overall contact map $\widehat{C} = \frac{1}{H}\sum_{h=1}^{H}\mathcal{A}^h$.

Factored Attention Factored attention simplifies the single layer of attention in multiple ways. First, only positional information is used to compute queries and keys, and only sequence information is used to compute values. Second, the positional embeddings are both set to be one-hot encodings given by identity matrices, i.e. $E_{pos} = I_L \in \mathbb{R}^{L\times L}$ and $E_{seq} = I_A \in \mathbb{R}^{A\times A}$. This means that queries, keys, and values are simply rows of the underlying weight matrices $W_K, W_Q \in \mathbb{R}^{L\times d}$ and $W_V \in \mathbb{R}^{A\times A}$. Note that the shape of W_K and W_Q now explicitly depends on the length of the MSA we train on. Third, the dense layer D is removed, since outputs are already of the correct dimension A. The underlying attention mechanism remains unchanged, but now the full conditional takes a simpler form -

$$p(x_i|x_{\setminus i}) = \mathrm{softmax}\left(\sum_{h=1}^{H}\sum_{j\neq i}\mathrm{softmax}\left(\frac{W_Q^h W_K^{h,T}}{\sqrt{d}}\right)_{ij} V_j^h\right). \tag{3}$$

By decoupling the queries and keys from sequence information, notice that the attention maps $\mathcal{A}$ are now the same for *all* inputs x. This is crucial for showing factored attention is a Pairwise MRF.

Predicting contacts from factored attention: Following the contact extraction procedure for a single attention layer, we could define $\widehat{C} = \frac{1}{H}\sum_{h=1}^{H}\mathcal{A}^h$. As we will show in Section 3.3, factored attention in fact comes from a valid Pairwise MRF with a pairwise interaction tensor given by $\tilde{W} = \sum_{h=1}^{H}\mathcal{A}^h \otimes W_V^h$. We choose to instead adopt the Frobenius norm approach from Potts models and predicted contacts with $\widehat{C} = \frac{1}{H}\sum_{h=1}^{H}\mathcal{A}^h\|W_V^h\|_F^2$.

3.3 Connecting Attention to Potts

Factored attention is a Pairwise MRF Consider the interaction matrix $\tilde{W}$ of shape $A \times A$ defined as $\tilde{W}^{ij} = \sum_h \mathcal{A}_{ij}^h W_V^h$. The full conditional from Equation 3 can be rewritten in terms of $\tilde{W}$ as $p(x_i|x_{\setminus i}) = \mathrm{softmax}\left(\sum_{j\neq i}\tilde{W}^{ij}(x_i, x_j)\right)$. Because queries and keys are not a function of the input sequence, the function $\tilde{W}^{ij}(x_i, x_j)$ is *only* a function of x_i and x_j. Thus the following

expression defines a valid joint distribution, showing that factored attention is a Pairwise MRF -

$$p(x) = \frac{1}{Z} \exp\left\{ \sum_{i<j} \tilde{W}^{ij}(x_i, x_j) \right\} = \frac{1}{Z} \exp\left\{ \sum_{i<j} \sum_{h=1}^{H} \mathrm{softmax}\left(\frac{W_Q^h W_K^{h,T}}{\sqrt{d}} \right)_{ij} x_j^T W_V^h x_i \right\}.$$

We now address why this tensor $\tilde{W}$ does not specify a valid Pairwise MRF for a layer of standard attention. If the queries Q and keys K are functions of sequence as well as position, then the softmax over rows implies that $\mathcal{A}_{ij}^h$ is a function of all positions x_k due to the denominator of the softmax. Understanding if an attention layer comes from a valid joint distribution is an open problem, as this softmax over sequences complicates matters considerably (Wang & Cho, 2019).

Comparing the Pairwise MRFs Comparing the two parametrizations of the pairwise tensors W and $\tilde{W}$ reveals that attention-based models benefit from multiple levels of parameter sharing. For a pair of positions x_i and x_j, a Potts model computes interactions with a single $A \times A$ interaction matrix W^{ij}. A natural intermediate between Potts and factored attention is to fully parametrize the $L \times L$ matrices $\mathcal{A}^h$. For this model, the interaction is $\sum_h \mathcal{A}_{ij}^h x_j^T W_V^h x_i$. This model uses H $A \times A$ interaction matrices W_V^h which are shared for all pairs of positions. Because amino acids interactions fall into common classes (Kawashima et al., 2008), sharing of the W_V^h matrices is natural. The disadvantage of this model is that it still has $\mathcal{O}(L^2)$ parameters and that the interactions between, e.g., (x_i, x_1) and (x_i, x_3) are given by independent parameters. Factored attention is a further factorization of this model, where the $L \times L$ matrices are assumed to be Gramian, $\mathcal{A} = W_Q W_K^T$. This fixes both issues, since x_i uses the query vector Q_i when computing its interaction with all positions and we now have $\mathcal{O}(L)$ parameters to train.

Since factored attention is a special case of the single attention layer, the latter enjoys all the parameter sharing benefits of the former. The single attention layer has the further advantage of sequence-dependent attention maps, a significant deviation from factored attention.

Inclusion of Single-Site Potentials The Potts model includes the single-site potentials $b^i(x_i)$ in addition to the W tensor of interactions. Typically, attention does not have such a single-site term. If included, this acts as a bias term added to the logits, where the vector $b^i \in \mathbb{R}^A$ is position-dependent.

Self-Supervised Losses Using the conditionals $p_\theta(x_i|x_{\setminus i})$ with parameters θ, Potts models are typically trained with Pseudolikelihood Maximization while attention-based models are trained with Masked Language Modeling. The corresponding losses $\mathcal{L}_{PL}$ and $\mathcal{L}_{MLM}$ are

$$\mathcal{L}_{PL}(\theta; x) = \sum_{i=1}^{L} \log p_\theta(x_i|x_{\setminus i}) \qquad \mathcal{L}_{MLM}(\theta; x, M) = \sum_{i \in M} \log p_\theta(x_i|x_{\setminus M}),$$

where $M \subset \{1, \cdots, L\}$ is a fixed mask. We train Potts and factored attention trained with both losses, since they can be implemented through masking the W or $\tilde{W}$ matrix directly. We train standard attention only with MLM, we need L separate forward passes to compute $\mathcal{L}_{PL}$.

4 Results

We use a set of 748 protein families from Yang et al. (2019) to evaluate all models. For Potts models and single attention layers, we train separate models on each individual MSA. Once trained, we extract contacts as described in Section 3. For the pretrained Transformer ProtBERT-BFD, we take the pretrained weights and extract contacts using the procedure described by Vig et al. (2020). Further details on ProtBERT-BFD are available in A.3. We train models using PyTorchLightning and Weights and Biases (Falcon, 2019; Biewald, 2020). We compare predicted contact maps $\widehat{C}$ to true contact maps C using standard metrics based on precision and recall. A particularly important metric is precision at L, where L is the length of the sequence. This is computed by sorting $\widehat{C}$, taking the top L entries to be predicted contacts, and computing the precision for these predictions. We provide more information on data and metrics in Appendix A.4.

We see in Figure 2 a representative sample of good quality contact maps extracted from the Potts model, factored attention, single-layer attention, and ProtBERT-BFD. All four models perform relatively well, with ProtBERT-BFD achieving the highest precision at L. Qualitatively, each model

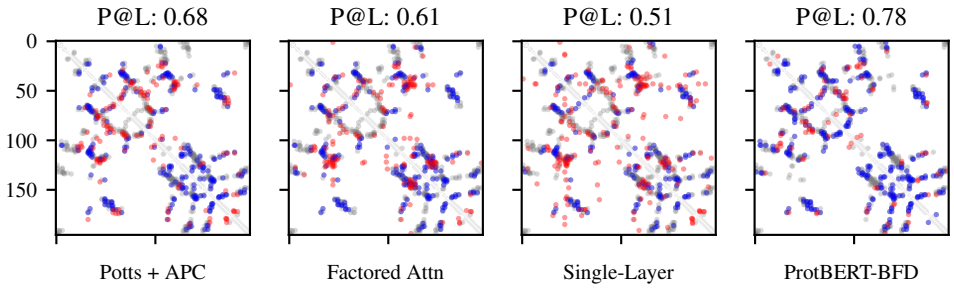


Figure 2: Predicted contact maps and precision at L ("P@L") for each model on 2BFW. Top $2L$ contacts shown, with blue indicating a true positive, red indicating a false positive, and grey indicating a true contact that was not predicted.

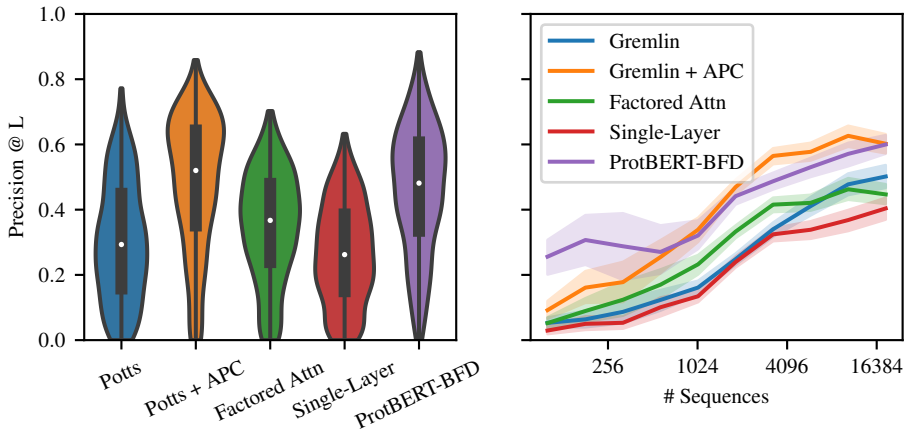


Figure 3: Left: Precision at L over multiple 748 families for each model. Right: Effect of sample size on precision at L.

achieves good coverage of clustered contacts across the structure, and false positives tend to be co-located with true contacts. In particular, the contact maps predicted by factored attention and Potts (with APC) are qualitatively similar.

In Figure 3, we summarize the performance of all models over the set of 748 protein families. We see from the violin plots that factored attention attains better performance than Potts and the single layer of standard attention. ProtBERT-BFD beats Potts and the two attention models. Once Average Product Correction (APC) is applied to the Potts model, the Potts model performance increases to above that of ProtBERT-BFD. In Appendix A.6, we show that APC does not considerably change the performance of any attention-based model. Since APC acts as a low-rank correction to contact maps, this suggests that the contact maps extracted from attention models do not have low-rank artefacts found in Potts models. One possible explanation is that the use of softmax in all attention models promotes sparsity of the attention maps.

Breaking down contacts by distance Prediction of long-range contacts is particularly valuable for structure prediction. We break down contacts into three groups: short (separated by 6-12 positions), medium (separated by 13-24 positions) and long (separated by more than 24 positions). Model ranking is maintained across the different contact classes, and all models do appreciably better at predicting longer range contacts than shorter range. See Appendix A.7 for details.

Effect of MSA Sample Size We next examine the effect of sample size on each model's performance. For MSAs with fewer than 256 sequences, ProtBERT-BFD has appreciably higher precision. Potts with APC closes this gap for sequences with more than 1024 sequences. We see in Figure 3 that across families of all sizes, factored attention uniformly outperforms single-layer at-

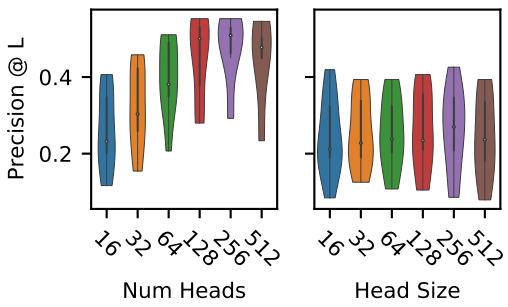


Figure 4: Examining effect of heads and head size on factored attention.

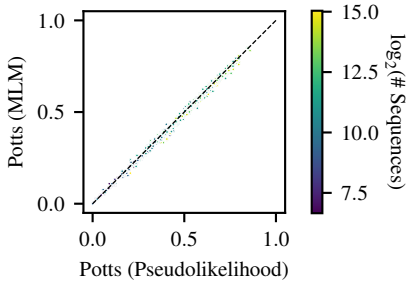


Figure 5: Masked Language Modeling trains Potts Models just as well as the standard Pseudolikelihood.

tention. Both attention layers are comparable to Potts without APC throughout the range of MSA sizes.

Replacing Pseudolikelihood with Masked Language Modeling We next compare Masked Language Modeling and pseudolikelood maximization for the Pairwise MRF-based models, namely Potts and factored attention. Figure 5 shows the comparison for Potts. Each point corresponds to a protein family (colored by sample size), with coordinates giving AUC for Potts with either loss. The two models track very closely across all 748 families. A similar trend is also seen for factored attention (see Appendix A.8).

Impact of number of heads To understand the impact of head size and number of heads on performance, we analyzed a set of 10 families over a range of hyperparameters. In Figure 4 we show the distribution of precision at L for each configuration. Increasing the number of heads has a large impact on performance up to 256 heads. Plotting the element-wise correlation between the W matrix of Potts and the $\tilde{W}$ matrix of factored attention, we see that factored attention takes advantage of many heads to have more Potts-like performance. As shown in Section 3.3, the number of heads in factored attention is the rank of the tensor $\tilde{W}$; this suggests why the partial recovery of W is impacted heavily by number of heads. Plots for intermediate values of H are available in Appendix A.9.

On the other hand, we find that the effect of head size for a fixed number of heads is marginal when compared to the trend with number of heads. Both performance and elementwise correlation (Appendix A.9) to the W matrix of Potts are not increasing as head size increases.

Ablations Given that the large Transformer is trained on unaligned sequences, we retrained our simple attention models on the test families without aligning the sequences. To do this, we remove all gaps and use standard padding in each batch during training. We found that performance decreased significantly (Appendix A.10). We also tested whether adding the single-site potential b^i had a significant impact on performance. We found that for both factored attention and standard attention, the addition of a single-site potential had a small but positive overall effect on performance (see also A.10).

5 Discussion

Shedding more light on what Transformers are learning from large protein databases is a pressing need to continue advancing state of the art for this field. Our work shows that attention layers can learn coevolutionary information when trained on Multiple Sequence Alignments. This raises an obvious question – are Transformers implicitly learning MSAs during pretraining? If so, how do we surface this information?

Another natural question is how to best represent the signal in MSAs. The evolutionary processes governing variation in protein families give rise to complex patterns of insertions and deletions. One potential approach for improving models is to draw on well-studied generative models from evolutionary biology to better model these patterns (Felsenstein & Felenstein, 2004; Yang et al.,

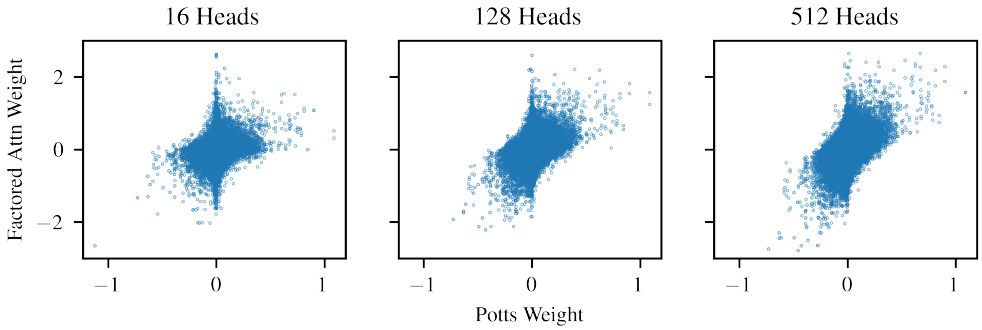


Figure 6: Factored attention weights become more correlated with Gremlin weights as the number of heads increases with Attention head size is fixed to 32. Pearson correlation increases from 0.06 with 16 attention heads to 0.24 with 512 attention heads.

2006). A complementary line of work has emerged which simultaneously learns from MSA-based features and self-supervised features (Luo et al., 2020; Gligorijevic et al., 2020). Whether neural and evolutionary features should be distinguished or integrated into one holistic architecture remains to be seen.

Training on a single family is the ideal setting for Potts and factored attention. We know structural fold is generally conserved within a protein family, so estimating a single graphical model for the family is a reasonable approximation. In the multifamily or heterogeneous family case, this assumption is now out the window and sequence-dependent attention maps become a key advantage. Is there a natural hierarchical extension of factored attention for dozens of related protein families (so-called superfamilies)? Similarly, for diverse families with structurally distinct subfamilies, can attention-based models be trained to identify these important clusters? Recently Shimagaki & Weigt (2019) and Tubiana et al. (2019) demonstrated success using Restricted Boltzmann Machines to model protein families. A combination of attention-based approaches with latent variable models could lead to wider model classes suited to modeling superfamilies.

The ability of factored attention to capture similar contacts to Potts without use of APC suggests that it may be more suitable for protein design. It has been previously shown that sequences sampled from Potts models fit with pseudolikelihood maximization do not match pairwise statistics of the MSA (Tubiana et al., 2019; Russ et al., 2020; Vorberg et al., 2018). Recent work has fit such models with sampling-based approaches such as Persistent Contrastive Divergence (Tieleman, 2008) and reported improved sample quality. Comparing sequences generated by factored attention with those from Potts models is a natural extension of this work.

Beyond immediate applications to protein science, our work contributes to the growing literature linking attention to other major topics in Machine Learning, such as Battaglia et al. (2018); Cordonnier et al. (2019); Ramsauer et al. (2020). Further theoretical analysis extending and explaining our study on the impact of number of heads is particularly interesting. We found that optimization of the models, especially single layer of attention, could be quite challenging. Difficulty of optimization is a major barrier to the applicability of Transformers, and is a major strength of Potts and Factored Attention. Future work on optimizing single-layer models could be fruitful for understanding how to train larger, multi-layer models.

REFERENCES

Sivaraman Balakrishnan, Hetunandan Kamisetty, Jaime G. Carbonell, Su-In Lee, and Christopher James Langmead. Learning generative models for protein fold families. *Proteins: Structure, Function, and Bioinformatics*, 79(4):1061–1078, apr 2011. ISSN 08873585. doi: 10.1002/prot.22934. URL `http://doi.wiley.com/10.1002/prot.22934`.

Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al.

Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

Lukas Biewald. Experiment tracking with weights and biases, 2020. URL https://www.wandb.com/. Software available from wandb.com.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are Few-Shot learners. May 2020.

Jean-Baptiste Cordonnier, Andreas Loukas, and Martin Jaggi. On the relationship between self-attention and convolutional layers. *arXiv preprint arXiv:1911.03584*, 2019.

Justas Dauparas, Haobo Wang, Avi Swartz, Peter Koo, Mor Nitzan, and Sergey Ovchinnikov. Unified framework for modeling multivariate distributions in biological sequences. June 2019.

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

Yilun Du, Joshua Meier, Jerry Ma, Rob Fergus, and Alexander Rives. Energy-based Models for Atomic-Resolution Protein Conformations. Technical report, sep 2019. URL https://github.com/facebookresearch/.

S D Dunn, L M Wahl, and G B Gloor. Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. *Bioinformatics*, 24(3):333–340, February 2008.

Richard Durbin, Sean R Eddy, Anders Krogh, and Graeme Mitchison. *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge university press, 1998.

Magnus Ekeberg, Cecilia Lövkvist, Yueheng Lan, Martin Weigt, and Erik Aurell. Improved contact prediction in proteins: using pseudolikelihoods to infer potts models. *Phys. Rev. E Stat. Nonlin. Soft Matter Phys.*, 87(1):012707, January 2013.

Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rihawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Debsindhu Bhowmik, and Burkhard Rost. ProtTrans: Towards cracking the language of life's code through Self-Supervised deep learning and high performance computing. July 2020.

WA Falcon. Pytorch lightning. *GitHub. Note: https://github.com/PyTorchLightning/pytorch-lightning*, 3, 2019.

Sarosh N. Fatakia, Stefano Costanzi, and Carson C. Chow. Computing Highly Correlated Positions Using Mutual Information and Graph Theory for G Protein-Coupled Receptors. *PLoS ONE*, 4 (3):e4681, mar 2009. ISSN 1932-6203. doi: 10.1371/journal.pone.0004681. URL https://dx.plos.org/10.1371/journal.pone.0004681.

Joseph Felsenstein and Joseph Felenstein. *Inferring phylogenies*, volume 2. Sinauer associates Sunderland, MA, 2004.

Anthony A. Fodor and Richard W. Aldrich. On Evolutionary Conservation of Thermodynamic Coupling in Proteins. *Journal of Biological Chemistry*, 279(18):19046–19050, apr 2004. ISSN 00219258. doi: 10.1074/jbc.M402560200. URL https://pubmed.ncbi.nlm.nih.gov/15023994/.

Vladimir Gligorijevic, P Douglas Renfrew, Tomasz Kosciolek, Julia Koehler Leman, Daniel Berenberg, Tommi Vatanen, Chris Chandler, Bryn C Taylor, Ian M Fisk, Hera Vlamakis, et al. Structure-based function prediction using graph convolutional networks. *bioRxiv*, pp. 786236, 2020.

Ulrike Göbel, Chris Sander, Reinhard Schneider, and Alfonso Valencia. Correlated mutations and residue contacts in proteins. *Proteins: Structure, Function, and Bioinformatics*, 18(4):309–317, 1994. ISSN 10970134. doi: 10.1002/prot.340180402. URL https://pubmed.ncbi.nlm.nih.gov/8208723/.

John Ingraham, Vikas K Garg, Regina Barzilay, and Tommi Jaakkola. Generative models for graph-based protein design. In *Neural Information Processing Systems*. MIT, 2019. URL https://papers.nips.cc/paper/9711-generative-models-for-graph-based-protein-design.

L. Steven Johnson, Sean R. Eddy, and Elon Portugaly. Hidden Markov model speed heuristic and iterative HMM search procedure. *BMC Bioinformatics*, 11(1):431, aug 2010. ISSN 14712105. doi: 10.1186/1471-2105-11-431. URL https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-11-431.

Hetunandan Kamisetty, Sergey Ovchinnikov, and David Baker. Assessing the utility of coevolution-based residue-residue contact predictions in a sequence- and structure-rich era. *Proc. Natl. Acad. Sci. U. S. A.*, 110(39):15674–15679, September 2013.

Shuichi Kawashima, Piotr Pokarowski, Maria Pokarowska, Andrzej Kolinski, Toshiaki Katayama, and Minoru Kanehisa. AAindex: amino acid index database, progress report 2008. *Nucleic Acids Res.*, 36(Database issue):D202–5, January 2008.

B T Korber, R M Farber, D H Wolpert, and A S Lapedes. Covariation of mutations in the V3 loop of human immunodeficiency virus type 1 envelope protein: an information theoretic analysis. *Proc. Natl. Acad. Sci. U. S. A.*, 90(15):7176–7180, August 1993.

Alan S Lapedes, Bertrand G. Giraud, LonChang Liu, and Gary D Stormo. Correlated mutations in models of protein sequences: Phylogenetic and structural effects. *Lect. Notes Monogr. Ser.*, 33: 236–256, 1999.

Steve W. Lockless and Rama Ranganathan. Evolutionarily conserved pathways of energetic connectivity in protein families. *Science*, 286(5438):295–299, oct 1999. ISSN 00368075. doi: 10.1126/science.286.5438.295. URL https://science.sciencemag.org/content/286/5438/295https://science.sciencemag.org/content/286/5438/295.abstract.

Yunan Luo, Lam Vo, Hantian Ding, Yufeng Su, Yang Liu, Wesley Wei Qian, Huimin Zhao, and Jian Peng. Evolutionary context-integrated deep sequence modeling for protein engineering. In *International Conference on Research in Computational Molecular Biology*, pp. 261–263. Springer, 2020.

A Madani, B McCann, N Naik, N S Keskar, and others. ProGen: Language modeling for protein generation. *arXiv preprint arXiv*, 2020.

Ananthan Nambiar, Simon Liu, Mark Hopkins, Maeve Heflin, Sergei Maslov, and Anna Ritz. Transforming the language of life: Transformer neural networks for protein prediction tasks. June 2020.

Sergey Ovchinnikov, Hetunandan Kamisetty, and David Baker. Robust and accurate prediction of residue-residue interactions across protein interfaces using evolutionary information. *Elife*, 3: e02030, May 2014.

Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Milena Pavlović, Geir Kjetil Sandve, Victor Greiff, et al. Hopfield networks is all you need. *arXiv preprint arXiv:2008.02217*, 2020.

Roshan Rao, Nicholas Bhattacharya, Neil Thomas, Yan Duan, Peter Chen, John Canny, Pieter Abbeel, and Yun Song. Evaluating protein transfer learning with TAPE. In H Wallach, H Larochelle, A Beygelzimer, F dAlché-Buc, E Fox, and R Garnett (eds.), *Advances in Neural Information Processing Systems 32*, pp. 9689–9701. Curran Associates, Inc., 2019.

Michael Remmert, Andreas Biegert, Andreas Hauser, and Johannes Söding. HHblits: Lightning-fast iterative protein sequence searching by HMM-HMM alignment. *Nature Methods*, 9(2):173–175, feb 2012. ISSN 15487091. doi: 10.1038/nmeth.1818. URL http://predictioncenter.org/casp9/groups{_}analysis.cgi?type=server{&}tbm=on/.

Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, 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. August 2020.

William P. Russ, Matteo Figliuzzi, Christian Stocker, Pierre Barrat-Charlaix, Michael Socolich, Peter Kast, Donald Hilvert, Remi Monasson, Simona Cocco, Martin Weigt, and Rama Ranganathan. An evolution-based model for designing chorismate mutase enzymes. *Science*, 369(6502):440–445, 2020. ISSN 0036-8075. doi: 10.1126/science.aba3304. URL https://science.sciencemag.org/content/369/6502/440.

Stefan Seemayer, Markus Gruber, and Johannes Söding. CCMpred - Fast and precise prediction of protein residue-residue contacts from correlated mutations. *Bioinformatics*, 30(21):3128–3130, may 2014. ISSN 14602059. doi: 10.1093/bioinformatics/btu500. URL https://pubmed.ncbi.nlm.nih.gov/25064567/.

Andrew W. Senior, Richard Evans, John Jumper, James Kirkpatrick, Laurent Sifre, Tim Green, Chongli Qin, Augustin Žídek, Alexander W.R. Nelson, Alex Bridgland, Hugo Penedones, Stig Petersen, Karen Simonyan, Steve Crossan, Pushmeet Kohli, David T. Jones, David Silver, Koray Kavukcuoglu, and Demis Hassabis. Improved protein structure prediction using potentials from deep learning. *Nature*, 577(7792):706–710, jan 2020. ISSN 14764687. doi: 10.1038/s41586-019-1923-7. URL https://doi.org/10.1038/s41586-019-1923-7.

Kai Shimagaki and Martin Weigt. Selection of sequence motifs and generative Hopfield-Potts models for protein families. *Phys Rev E*, 100(3-1):032128, September 2019.

Martin Steinegger, Markus Meier, Milot Mirdita, Harald Vöhringer, Stephan J. Haunsberger, and Johannes Söding. HH-suite3 for fast remote homology detection and deep protein annotation. *BMC Bioinformatics*, 20(1):473, sep 2019. ISSN 14712105. doi: 10.1186/s12859-019-3019-7. URL https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-019-3019-7.

Baris E Suzek, Hongzhan Huang, Peter McGarvey, Raja Mazumder, and Cathy H Wu. UniRef: comprehensive and non-redundant UniProt reference clusters. *Bioinformatics*, 23(10):1282–1288, May 2007.

Tijmen Tieleman. Training restricted boltzmann machines using approximations to the likelihood gradient. In *Proceedings of the 25th international conference on Machine learning*, pp. 1064–1071, 2008.

Jérôme Tubiana, Simona Cocco, and Rémi Monasson. Learning protein constitutive motifs from sequence data. *Elife*, 8, March 2019.

J Vig, A Madani, L R Varshney, C Xiong, and others. Bertology meets biology: Interpreting attention in protein language models. *arXiv preprint arXiv*, 2020.

Susann Vorberg, Stefan Seemayer, and Johannes Söding. Synthetic protein alignments by ccmgen quantify noise in residue-residue contact prediction. *PLOS Computational Biology*, 14(11): 1–25, 11 2018. doi: 10.1371/journal.pcbi.1006526. URL https://doi.org/10.1371/journal.pcbi.1006526.

Alex Wang and Kyunghyun Cho. Bert has a mouth, and it must speak: Bert as a markov random field language model. *arXiv preprint arXiv:1902.04094*, 2019.

Sheng Wang, Siqi Sun, Zhen Li, Renyu Zhang, and Jinbo Xu. Accurate de novo prediction of protein contact map by ultra-deep learning model. *PLOS Computational Biology*, 13(1):1–34, 01 2017. doi: 10.1371/journal.pcbi.1005324. URL https://doi.org/10.1371/journal.pcbi.1005324.

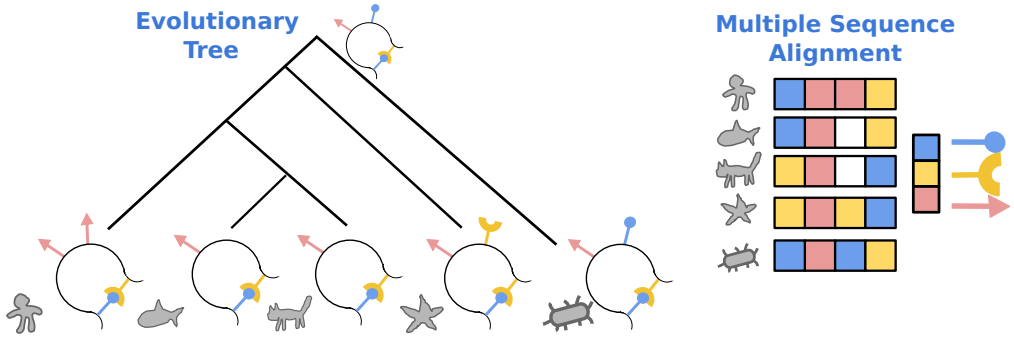


Figure 7: The tree on the right depicts evolution of a protein family. The protein at the root is the ancestral protein, and the five proteins at the leaves are its present-day descendants. The alignment on the left is the corresponding Multiple Sequence Alignment of observed sequences.

M Weigt, R A White, H Szurmant, and others. Identification of direct residue contacts in protein–protein interaction by message passing. *Proceedings of the*, 2009.

Jianyi Yang, Ivan Anishchenko, Hahnbeom Park, Zhenling Peng, Sergey Ovchinnikov, David Baker, and John Harvard. Improved protein structure prediction using predicted inter-residue orientations. *bioRxiv*, pp. 846279, 2019. doi: 10.1101/846279. URL https://www.biorxiv.org/content/10.1101/846279v1.

Ziheng Yang et al. *Computational molecular evolution*, volume 284. Oxford University Press Oxford, 2006.

A Appendix

A.1 Multiple Sequence Alignments

Protein Evolution We demonstrate key concepts of protein evolution in Figure 7. On the left is a *phylogenetic tree*. The leaves represent five observed proteins in a family, while the root represents their most recent common ancestor. The ancestral protein was a loop and evolution preserved this loop structure. Thus every observed sequence has one yellow and one blue amino acid on its ends, which lock together to pinch off the loop. The amino acids in the middle of the loop exhibit considerable differences within the family, presumably leading to variations in function. This variation within the protein family is captured by the MSA on the right through its placement of gap characters. Compared to standard padding, shown in 8, the placement of gap characters (white squares) in the MSA ensures all blue and yellow amino acids lie in a single column. This signal is obfuscated by standard padding.

The problem of multiplying aligning a set of sequences has been extensively studied in Bioinformatics and Computational Biology (Durbin et al., 1998). Given a set of N sequences of average length $\tilde{L}$, a full multiple alignment can be generated by performing dynamic programming on an array of size $\mathcal{O}(\tilde{L}^N)$. This is computationally intractable; the common workaround is to perform iterative pairwise alignment to a fixed reference sequence. We use the HHSuite (Steinegger et al., 2019) for generating MSAs.

A.2 Models

A.2.1 Potts Models

Loss For loss $\mathcal{L}$ set either to $\mathcal{L}_{MLM}$ or $\mathcal{L}_{PL}$, we use the ℓ_2 regularization with factor λ and MSA-specific scaling from Ovchinnikov et al. (2014)

$$\mathcal{L}_{reg}(W, b; x) = \mathcal{L}(W, b; x) + \frac{\lambda LA}{2N}\|W\|_F + \frac{\lambda}{N}\|b\|_F.$$

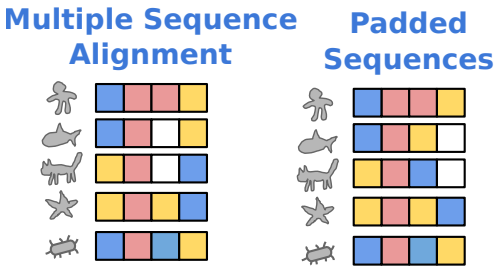


Figure 8: MSA for sequences from Figure 7 compared to a padded batch of the same sequences.

Optimizer Originally, the Potts model implementation we follow used L-BFGS. to perform pseudolikelihood maximization, since it is a convex problem. To accommodate first-order optimization, Dauparas et al. (2019) introduced a modification of Adam which adds up all running variances for the parameters of W and b.

Special Tokens Pads and gaps at positions in x are encoded as a vector of all zeros. This ensures that the model does not condition on those positions when computing full conditionals. Similarly, when a position is masked for MLM it is set to all zeros in order to prevent information leakage.

Hyperparameters We set $\lambda = 0.5$ and use a batch size of 4096.

A.2.2 Scaled Dot-Product Attention Layer

Special Tokens We augment the amino acid set $\mathcal{A}$ to include a special pad/gap token and a special mask token. These tokens are placed at appropriate positions during training and have separate learned embeddings in E_{seq}.

Hyperparameters We use embedding size $e = 256$, attention head size $d = 64$, and number of heads $H = 16$. For training, we use AdamW with learning rate $5e - 3$ and weight decay $2e - 3$, attention dropout rate of 0.05, and batch size of 64. For masked language modeling, we set mask probability to 0.15.

A.2.3 Factored Attention Layer

Special Tokens As with Potts models, all pad, gap, and mask tokens are encoded as a vector of zeros. Since one-hots are fed directly to the value matrices W_V, this results in those positions being ignored when computing full conditionals.

Hyperparameters We found that factored attention was much easier to train than standard attention, so we chose to match hyperparameters for simplicity. Thus, attention head size $d = 64$, and number of heads $H = 16$. For training, we use AdamW with learning rate $5e - 3$ and weight decay $2e - 3$, attention dropout rate of 0.05, and batch size of 64. For masked language modeling, we set mask probability to 0.15.

A.3 ProtBERT Details

Hyperparameters ProtBERT-BFD has 16 heads with a hidden size of 1024 and 30 layers. The training dataset BFD has $2,122$ million protein sequences. See Elnaggar et al. (2020) more information.

Since we do not fine-tune any pretrained Transformers for this study, we treat a transformer with K layers and H heads as a fixed mapping from a sequence x of length L to a collection of KH attention maps $\mathcal{A}^{h,k}(x) \in \mathbb{R}^{L \times L}$.

Contact Extraction Let S index a subset of heads across layers, so $S \subset \{1, \ldots, K\} \times \{1, \ldots, H\}$. For example, $S = \{(1, 8), (10, 6)\}$ indexes the eight head of layer one and the sixth head of layer ten. Following Vig et al. (2020), we average the attention maps of these heads to obtain a contact

map,

$$\widehat{C}^S_{ij}(x) = \frac{1}{|S|} \sum_{s \in S} \mathcal{A}^s_{ij}(x).$$

Identification of Heads for Contact Prediction Crucial to these contact extraction procedures is selection of the subset S. We offer two possibilities. We used a set of five hundred protein structures separate from our test set to identify heads with high individual precision for contact prediction. Six heads were identified all from the last layer: heads 0, 2, 4, 7, 8, and 11.

A.4 Evaluation Procedure

A.4.1 Selection of Protein Families

The experiments in this paper are performed on a subset of families used in the TRRosetta paper Yang et al. (2019). This dataset consists of 15,015 MSAs generated from the UniRefSuzek et al. (2007) database as well as metagenomic datasets.

Figure 9: The length and MSA size distribution for our 748 family subset compared to the full 17000 families in the trRosetta dataset selected for training

Selection of Families We built three sets of families for model development.

1. A set of six families was chosen for initial debugging and hyperparameter sweeps, such as exploring number of heads.
2. A set of ten families was chosen for further optimization of training details, notably learning rate, weight decay and various forms of dropout.
3. A further set of 732 families was chosen as the final test run. Only one choice of hyperparameters for Potts models, factored attention, and standard attention was used for testing on this family. These families were chosen randomly from trRosetta, and is a representative subsample of the full set, see Figure 9

The set of six families was chosen based on an initial run of Potts with default hyperparameters. We chose a family with lots of sequences $(N \geq 10000)$ on which Potts did well and poorly. We chose similar families for $1000 \leq N \leq 10000$ and $N \leq 1000$.

The set of ten families was chosen to represent a large span of lengths in the TRRosetta dataset. We created grouped length into fifty buckets and sampled one family per bucket.

The set of 732 families was sampled at random from the remaining families in TRRosetta. Plots shown below indicate that this sample is a reasonable representation of both length and sample size variation in the overall TRRosetta dataset, thus we decided to avoid more complicated family identification procedures. The 748 families on which we report results in violin plots are the combination of all three sets of families mentioned above.

A.4.2 Producing Contact Maps

A PDB structure gives 3D coordinates for every atom in a structure. We use Euclidean distance between the beta carbons to define distance between any pair of positions. A pair of positions where this distance is less than $8\mathring{A}$ is declared to be a contact.

A.4.3 Scoring Contact Predictions

Given a predicted contact map $\widehat{C} \in \mathbb{R}^{L \times L}$ and a true contact map $C \in \{0,1\}^{L \times L}$, we describe metrics for scoring $\widehat{C}$.

A sequence $x = (x_1, \ldots, x_L)$ of length L has $\binom{L}{2}$ potential contacts. Empirical evidence indicates that we expect to see $\mathcal{O}(L)$ contacts, making contact prediction a sparse prediction task. Accordingly, we focus on precision-recall based quantitative analyses of $\hat{C}$. Common practice in the field is to sort all $\binom{L}{2}$ entries of $\hat{C}$ in decreasing order and evaluate precision at various length thresholds, such as the top L or $L/10$ predictions. Note that this analysis is analogous to choosing recall cutoffs along a precision-recall curve, where sorted length index plays the role of recall on the x axis. In addition to the precision at various length (recall) cutoffs, we also computed Area Under the Precision-Recall Curve (AUC). AUC is a widely used metric for comparing classifiers when the positive class is rare.

A.5 Hyperparameter Optimization

Potts Model The Potts model implementation using psuedolikelihood has been optimized by others, so we did not tune performance. Since performance with MLM was comparable to pseudolikelihood, we did not sweep for MLM either.

Standard Attention Standard attention is by far the most sensitive model to hyperparameters. To find a reasonable set of hyperparameters, we first swept over the six families in the debug set, performing a grid search over

- $H \in \{32, 64, 128, 256, 512\}$
- $d \in \{32, 64, 128, 256, 512\}$
- $e \in \{128, 256, 512\}$
- attention dropout in $\{0, 0.1\}$
- learning rate in $\{1e-2, 5e-3\}$
- weight decay in $\{1e-3, 0\}$

We found that the choice $H = 256$, $d = 32$, $e = 128$, attention dropout of 0.1, learning rate of $5e-3$ and weight decay of $1e-3$ performed well across all six families. Due to GPU memory constraints, we had to set $H = 128$ for further runs. To ensure training dynamics were reasonable, we then ran a further 1992 runs on 50 families with random uniform distributions over

- attention dropout in $[0, 0.15]$
- learning rate in $[2e-2, 1e-3]$
- weight decay in $[0, 1e-2]$.

This sweep led to the configuration of learning rate $5e-3$, weight decay of $2e-3$, and attention dropout of 0.05.

Factored Attention A sweep similar to the one for standard attention revealed that more heads helped performance, as detailed in Section 4. Due to lightened memory requirements, we $H = 256$ and otherwise matched hyperparameters with standard attention. We found factored attention to be more robust to selection of weight decay and learning rate than standard attention.

A.6 Effect of APC for Attention Models

Figure 10 shows the distribution of precision at L for attention-based models on the left and the same distribution after applying APC on the right. APC has a small positive effect on the performance of factored attention. It makes no overall difference on the performance of the other two models.

A.7 Detailed Contact Prediction Results

Figure 11 shows the breakdown of contacts into different regimes. We see that for short-range contacts, ProtBERT-BFD has a slight edge. On the other hand, for long-range contacts the Potts model with APC does best.

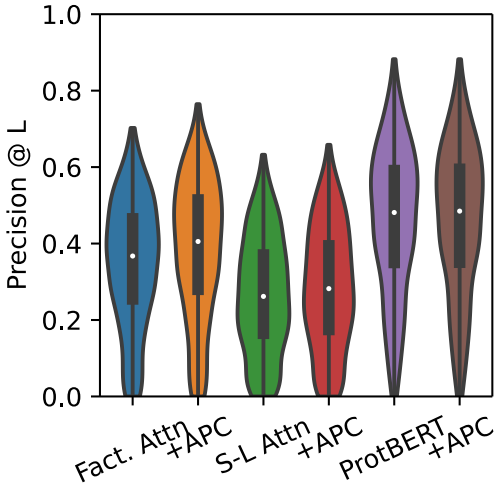


Figure 10: APC has a small positive effect on the performance of factored attention. It makes no overall difference on the performance of the other two models. In Figure 3 we show that APC has a dramatic effect on the Potts model.

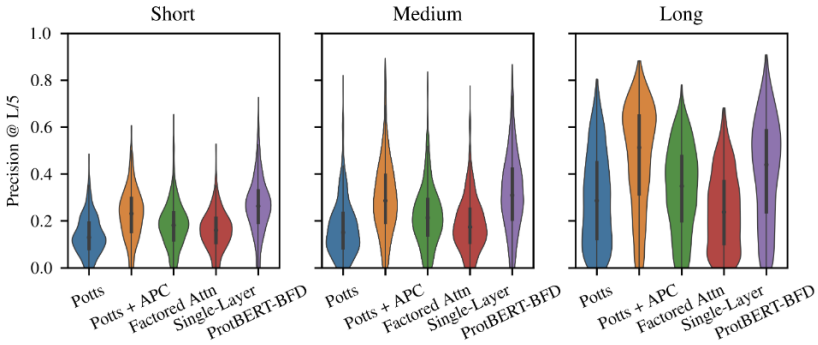


Figure 11: Short range (separated by 6-12 positions), Medium (13-24) and Long ($\geq$ 24) range contacts. Model ranking is maintained across the different contact classes, and all models do appreciably better at predicting longer range contacts.

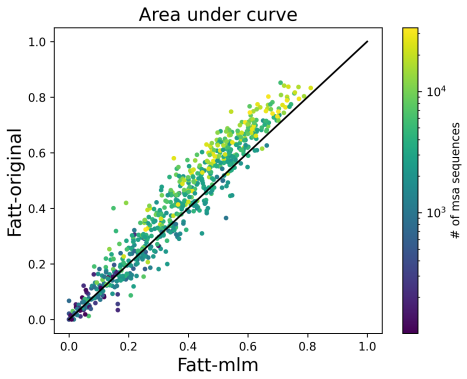


Figure 12: Masked Language Modeling doesn't significantly change the performance of the factored attention model.

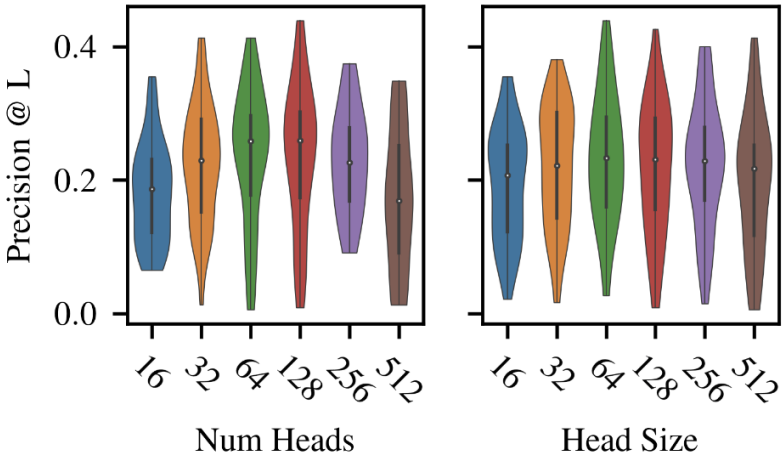


Figure 13: Effect of head size and number of heads on single-layer attention's performance. The left plot is produced for fixed head size of 32. The right plot is produced for fix number of heads 16.

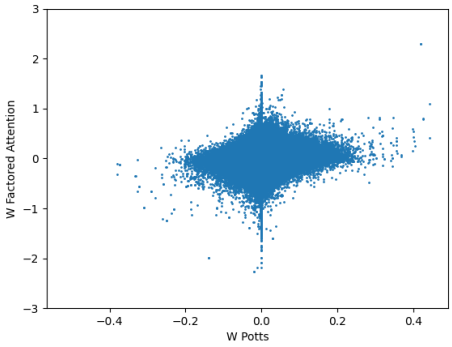
A.8 Comparing Losses for Factored Attention

In Figure 12, we plot the performance of factored attention trained with pseudolikelihood versus its performance with MLM. We see that pseudolikelihood confers a larger advantage for factored attention than for the Potts model, but that there are still no outliers of particular families that benefit greatly. Coloring by MSA size indicates that there is some added increase in performance for medium to large-size families.

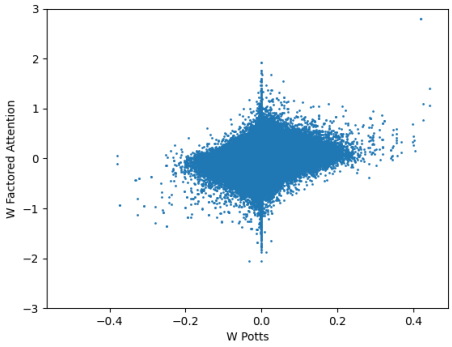
A.9 Impact of Heads and Head Size

Figure 13 shows the impact of head size and number of heads separately on the performance of a single layer of attention. There is some increase in performance up until 128 heads, consistent with the result seen for factored attention. There is also a less pronounced trend suggesting head size has some positive benefit but it isn't conclusive.

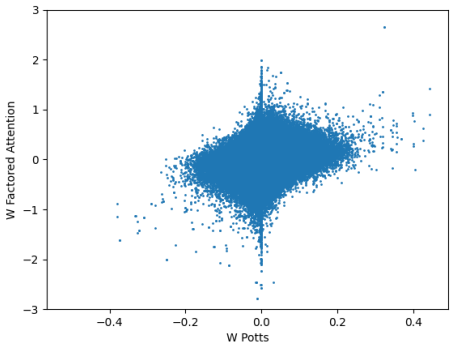
In Figure 14 we see that increasing the number of heads while holding the head size fixed leads to increased correlation of the factorized attention $\tilde{W}$ with the Potts W matrix. This trend does not hold for the reverse in Figure 15: holding number of heads fixed while scaling head size does not lead to appreciable increases in correlation



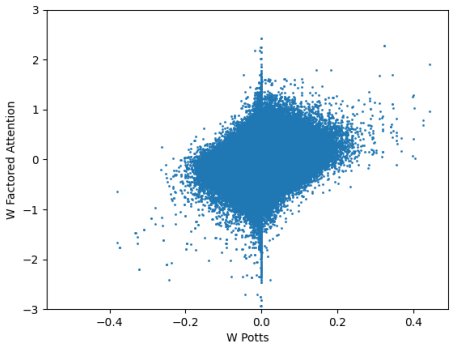
(a) 16 heads, size 32



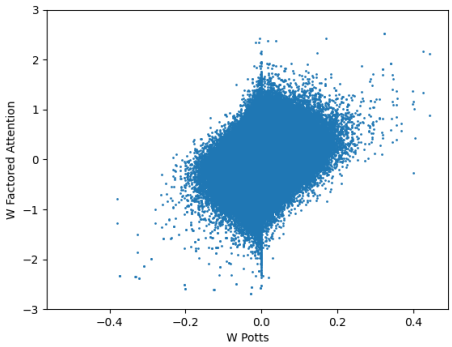
(b) 32 heads, size 32



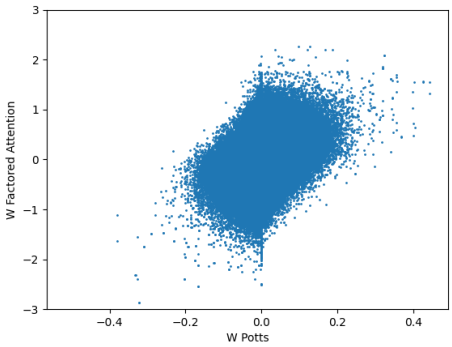
(c) 64 heads, size 32



(d) 128 heads, size 32

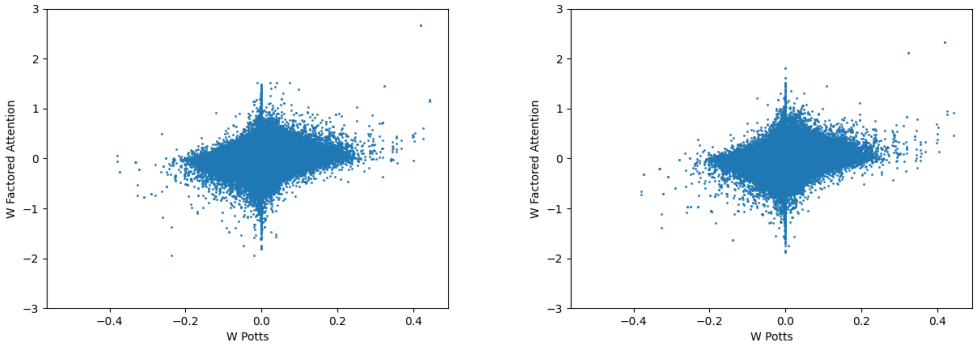


(e) 256 heads, size 32



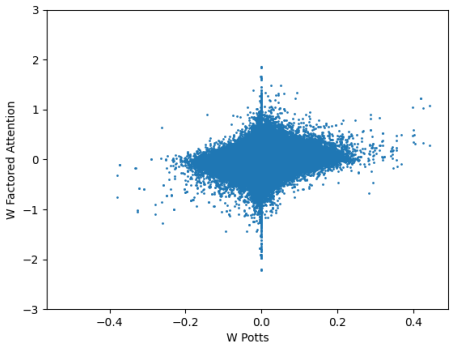
(f) 512 heads, size 32

Figure 14: Increasing the number of heads leads to increased correlation with the Potts W matrix

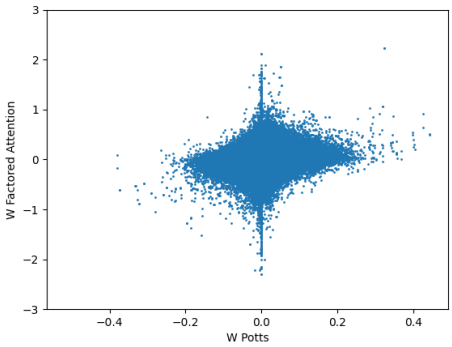


(a) 16 heads, size 16

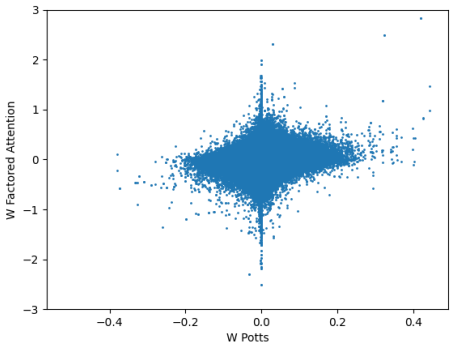
(b) 16 heads, size 32



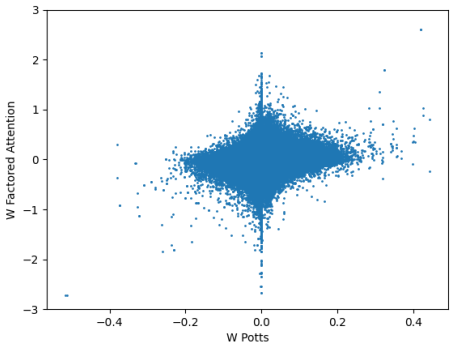
(c) 16 heads, size 64



(d) 16 heads, size 128

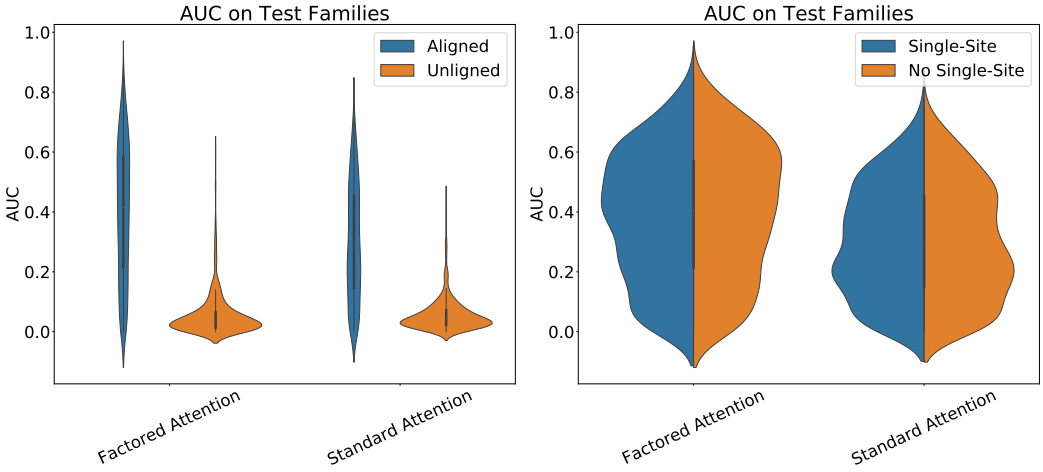


(e) 16 heads, size 256



(f) 16 heads, size 512

Figure 15: Increasing head size does not lead to increased correlation with the Potts W matrix



(a) Impact of training on unaligned sequences.

(b) Effect of single-site potential on attention layers.

A.10 Ablations

In Figure 16a, we see that there's a drastic reduction in performance when training on unaligned sequences. It appears one particular family showed nontrivial AUC for both attention models even when unaligned, but this is likely because the family has few gaps. This experiment strongly suggests that small models trained directly on unaligned sequences should not be expected to succeed.

In Figure 16b, we see that addition of a single-site term to either factored or standard attention produces little additional benefit. The hyperparameters for both models are fully fixed except for the addition or removal of the single-site potential. This suggests that attention models are able to learn the single-site statistics without an explicit term to absorb that signal.