A. Appendix

A.1. Unsupervised Contact Prediction

For unsupervised contact prediction, we adopt the methodology from Rao et al. (2021), which shows that sparse logistic regression trained on the attention maps of a single-sequence transformer is sufficient to predict protein contacts using a small number (between $1 - 20$) of training structures. To predict the probability of contact between amino acids at position i and j, the attention maps from each layer and head are independently symmetrized and corrected with APC (Dunn et al., 2008). The input features are then the values $\bar{a}_{lhij}$ for each layer l and head h. The models have 12 layers and 12 heads for a total of 144 attention heads.

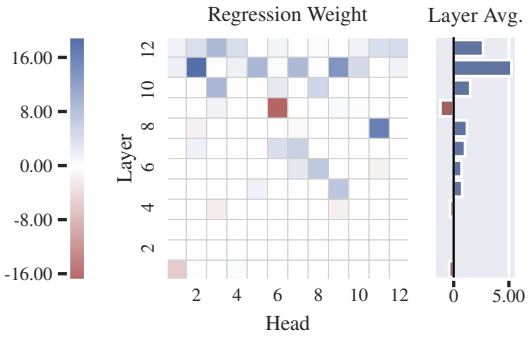

Figure A.1. Weight values of learned sparse logistic regression trained on 20 structures. A sparse subset (55 / 144) of contact heads, largely in the final layers, are predictive of protein contacts.

An L1-regularization coefficient of 0.15 is applied. The regression is trained on all contacts with sequence separation ≥ 6. 20 structures are used for training. Trained regression weights are shown in Fig. A.1.

A.2. Dataset Generation

For the unsupervised training set we retrieve the UniRef-50 (Suzek et al., 2007) database dated 2018-03. The UniRef50 clusters are partitioned randomly in 90% train and 10% test sets. For each sequence, we construct an MSA using HHblits, version 3.1.0. (Steinegger et al., 2019) against the UniClust30$_{2017-10}$ database (Mirdita et al., 2017). Default settings are used for HHblits except for the the number of search iterations (`-n`), which we set to 3.

A.3. Ablation Studies

Ablation studies are conducted over a set of seven hyperparameters listed in Table A.2. Since the cost of an exhaustive search over all combinations of hyperparameters is prohibitive, we instead train an exhaustive search over four of the hyperparameters (embedding size, block order, tied attention, and masking pattern) for 10k updates. The best run is then selected as the base hyperparameter setting for the full ablation study, in which only one parameter is changed at a time.

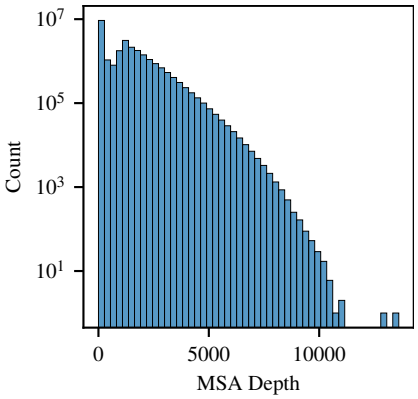

Figure A.2. Distribution of MSA depths in the MSA Transformer training set. Average MSA depth is 1192 and median MSA depth is 1101.

Table A.1. Validation perplexity and denoising accuracy on UniRef50 validation MSAs. PSSM probabilities and nearest-neighbor matching are used as baselines. To compute perplexity under the PSSM, we construct PSSMs using the input MSA, taking the cross-entropy between the PSSM and a one-hot encoding of the masked amino acid. When calculating PSSM probabilities, we search over pseudocounts in the range $[10^{-10}, 10)$, and select 10^{-2}, which minimizes perplexity. For denoising accuracy, the argmax for each column is used. For nearest-neighbor matching, masked tokens are predicted using the values from the sequence with minimum hamming distance to the masked sequence. This does not provide a probability distribution, so perplexity cannot be calculated. MSAs with depth 1 are ignored, since the baselines fail in this condition. Perplexity ranges from 1 for a perfect model to 21 for a uniform model selecting over the common amino acids and gap token.

Model	Perplexity	Denoising Accuracy
PSSM	14.1	41.4
Nearest-Neighbor	-	46.7
MSA Transformer	**2.44**	**64.0**

For the full ablation study, each model is trained for 100k updates using a batch size of 512. The four best performing models are then further trained to 150k updates. Contact prediction on the trRosetta dataset (Yang et al., 2019) is used as a validation task. Precision after 100k updates (and 150k for the best models) is reported in Table A.2 and the full training curves are shown in Fig. A.3. The model with best hyperparameters is then further trained to 450k updates. The performance of this model is reported in Table A.3. Validation perplexity is also reported in Table A.2. In general we find limited correspondence between perplexity and contact prediction performance across models.

Table A.2. Hyperparameter search on MSA Transformer. P@L is long-range ($s \geq 24$) precision on unsupervised contact prediction following Rao et al. (2021). Perplexity is reported after 100k updates and precision is reported after 100k and 150k updates.

D	Block	Tied	Masking	Mask p	MSA Pos Emb	Subsample	P@L (100k)	P@L (150k)	Ppl (100k)
768	Row-Column	Sqrt	Uniform	0.15	No	Log-uniform	**56.3**	56.3	3.01
384							52.8	-	3.10
	Column-Row						55.7	-	3.01
		None					42.1	-	3.03
		Mean					50.1	-	3.00
			Column				38.8	-	3.54
				0.2			**56.6**	56.3	3.04
					Yes		**56.5**	**57.1**	3.00
						Full	**56.5**	56.1	**2.91**

Table A.3. Average precision on 14842 test structures for MSA and single-sequence models trained on 20 structures.

	$6 \leq$ sep < 12			$12 \leq$ sep < 24			$24 \leq$ sep		
Model	L	L/2	L/5	L	L/2	L/5	L	L/2	L/5
Potts	17.2	26.7	44.4	21.1	33.3	52.3	39.3	52.2	62.8
TAPE	9.9	12.3	16.4	10.0	12.6	16.6	11.2	14.0	17.9
ProtBERT-BFD	20.4	30.7	48.4	24.3	35.5	52.0	34.1	45.0	57.4
ProTrans-T5	20.1	30.6	48.5	24.6	36.1	52.4	35.6	46.1	57.8
ESM-1b	21.6	33.2	52.7	26.2	38.6	56.4	41.1	53.3	66.1
MSA Transformer	**25.6**	**41.0**	**64.6**	**31.9**	**48.9**	**71.1**	**57.4**	**71.7**	**82.1**

Table A.4. Supervised Contact Prediction performance on CASP13-FM and CAMEO-hard targets. Reported numbers are long-range ($s \geq 24$) contact precision. Three variants of the MSA Transformer are included for comparison: *unsupervised model, †supervised model using final hidden representations of the reference sequence as input, ‡supervised model using final hidden representations of reference sequence and all attention maps as input. Baseline and final trRosetta models are also included for comparison. L is defined as the number of valid residues.

	CASP13-FM			CAMEO		
Model	L	L/2	L/5	L	L/2	L/5
Co-evolutionary	40.1	52.5	65.2	47.3	60.9	72.1
Unirep	11.2	14.5	16.6	17.8	23.0	30.8
SeqVec	13.8	18.3	21.9	22.5	30.3	39.8
TAPE	12.3	14.4	17.8	15.9	20.6	26
ProtBERT-BFD	24.7	32.1	40.6	37.0	48.1	60.0
ProTrans-T5	25.0	32.9	41.4	40.8	52.5	63.3
ESM-1b	28.2	37.4	50.2	44.4	57.2	68.4
trRosetta$_{\text{base}}$	45.7	58.4	69.6	-	-	-
trRosetta$_{\text{full}}$	51.8	66.6	80.1	53.2	67.1	77.5
MSA Transformer*	43.4	58.2	71.1	43.4	56.0	66.2
MSA Transformer†	54.5	**70.0**	**80.2**	53.6	68.4	78.0
MSA Transformer‡	**54.6**	68.4	77.5	**55.8**	**69.8**	**79.1**

Potts (Balakrishnan et al., 2011), TAPE transformer (Rao et al., 2019), ESM-1b (Rives et al., 2020), ProtBERT-BFD, and ProTrans-T5 (Elnaggar et al., 2020) are used as unsupervised contact prediction comparisons. The best MSA Transformer outperforms all other methods by a wide margin, increasing long-range precision at L by a full 16 points. See below for a discussion of all seven hyperparameters.

A.3.1. Embedding Size (D)

Since the MSA Transformer is provided with more information than single sequence protein language models, it is possible that many fewer parameters are needed to learn the data distribution. To test this hypothesis we train a model with half the embedding size (384 instead of 768) resulting in 30M total parameters. The resulting model achieves a Top-L long-range precision of 52.8 after 100k updates, 3.5 points lower than the baseline model. This suggests that model size is still an important factor in contact precision, although also shows that a model with fewer than 30M parameters can still outperform 650M and 3B parameter single-sequence models.

A.3.2. Masking Pattern

We consider two strategies for applying BERT masking to the MSA: uniform and column. Uniform masking applies masking uniformly at random across the MSA. Column masking always masks full columns of the MSA. This makes the training objective substantially more difficult since the model cannot look within a column of an MSA for information about masked tokens. We find that column masking is significantly worse (by almost 20 points) than uniform masking.

A.3.3. Block Ordering

Row attention followed by column attention slightly outperforms column attention followed by row attention.

A.3.4. Tied Attention

We consider three strategies for row attention: untied, mean normalization, and square root normalization (see Section 3). We find that tied attention substantially outperforms untied attention and that square root normalization outperforms mean normalization.

A.3.5. Masking Percentage

As the MSA Transformer has more context than single sequence models, its training objective is substantially easier than that of single sequence models. Therefore, we explore whether increasing the masking percentage (and thereby increasing task difficulty) would improve the model. However, we do not find a statistically significant difference between masking 15% or 20% of the positions. Therefore, we use a masking percentage of 15% in all other studies for consistency with ESM-1b and previous masked language models.

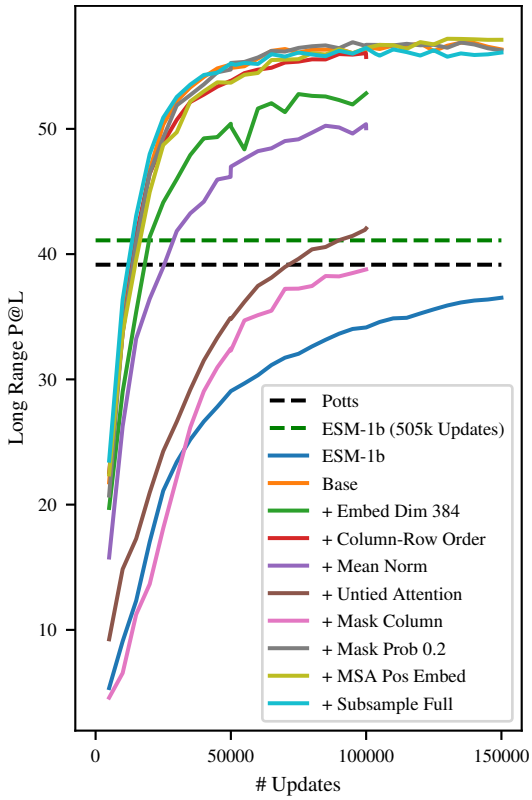

Figure A.3. Training curves for MSA Transformer with different hyperparameters. See Section 4.4 for a description of each hyperparameter searched over. ESM-1b training curve, ESM-1b final performance (after 505k updates), and average Potts performance are included as dashed lines for comparison.

A.3.6. MSA Positional Embedding

An MSA is an unordered set of sequences. However, due to the tools used to construct MSAs, there may be some pattern to the ordering of sequences in the MSA. We therefore examine the use of a learned MSA positional embedding in addition to the existing learned sequence positional embedding. The positional embedding for a sequence is then a learned function of its position in the input MSA (not in the full MSA). Subsampled sequences in the input MSA are sorted according to their relative ordering in the full MSA. We find that the inclusion of an MSA positional embedding does modestly increase model performance, and therefore include it in our final model.

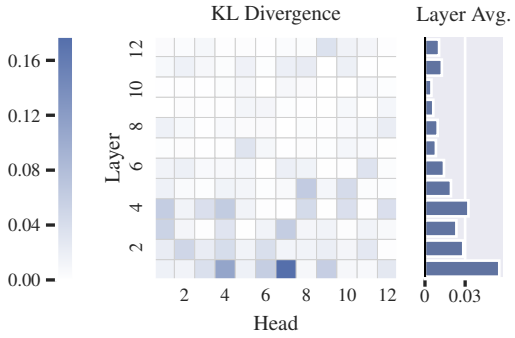

Figure A.4. KL Divergence between distribution of row attention across amino acids and background distribution of amino acids. The fraction of attention on an amino acid k is defined as the average over the dataset of $a_i^{lh}\mathbb{1}\{x_i == k\}$, where x_i is a particular token in the input MSA and a^{lh} is the attention in a particular layer and head. KL Divergence is large for early layers but decreases in later layers.

A.3.7. Subsample Strategy

At training time we explore two subsampling strategies. The first strategy is adapted from Yang et al. (2019): we sample the number of output sequences from a log-uniform distribution, with a maximum of N/L sequences to avoid exceeding the maximum tokens we are able to fit in GPU memory. Then, we sample that number of sequences uniformly from the MSA, ensuring that the reference sequence is always chosen. In the second strategy, we always sample the full N/L sequences from the MSA. In our hyperparameter search, most models use the first strategy, while our final model uses the second. We find no statistically significant difference in performance between the two strategies. However, it is possible that the log-uniform strategy would help prevent overfitting and ultimately perform better after more training.

The CCMpred implementation of Potts (Balakrishnan et al., 2011; Ekeberg et al., 2013), UniRep (Alley et al., 2019), SeqVec (Heinzinger et al., 2019), TAPE transformer (Rao et al., 2019), ESM-1b (Rives et al., 2020), ProtBERT-BFD, and ProTrans-T5 (Elnaggar et al., 2020) are used as supervised contact prediction comparisons. In Table A.4 we show the complete results for long-range precision over the CASP-13 FM targets and CAMEO-hard domains referenced in (Yang et al., 2019). All baseline models are trained for 200 epochs with a batch size of 4.

A.4. Attention to Amino Acids

Vig et al. (2020) examine the distribution of amino acids attended to by single-sequence models. The attention in single-sequence models is roughly equivalent to the row-attention in our model, but there is no column-attention analogue. We therefore examine the distribution of amino acids attended to by the column attention heads. In Fig. A.4 we show the KL-divergence between the distribution of attention across amino acids and the background distribution of amino acids. The divergence is large for earlier layers in the model but decreases in later layers, suggesting the model stops focusing on the amino acid identities in favor of focusing on other properties.

A.5. Sequence Weights

Sequence reweighting is a common technique used for fitting Potts models which helps to compensate for data bias in MSAs (Morcos et al., 2011). Informally, sequence reweighting downweights groups of highly similar sequences to prevent them from having as large of an effect on the model. The sequence weight w_i is defined as,

$$w_i = \left(1 + \sum_{j \neq i} \mathbb{1}\left\{d_{\text{hamming}}(x_i, x_j) < 0.2\right\}\right)^{-1} \tag{3}$$

where x_i, x_j are the i-th and j-th sequences in the MSA, d_{hamming} is the hamming distance between two sequences normalized by sequence length, and w_i is the sequence weight of the i-th sequence.