## Appendix A.

### A.1. *Recovering Factored Attention from Standard Attention*

Potts and Factored Attention estimate a single undirected graphical model from the training data. While a single graph can be a good approximation for the structure associated with a protein family, many families have *subfamilies* with different functional specializations and even different underlying contacts.[45,46] Since subfamily identity is rarely known, allowing edge weights to be a function of sequence could enable the estimation of a family of graphs.

In the language of the Transformer, factored attention estimates a single graph because it computes queries and keys using only the positional encoding. We show more precisely that factored attention can be recovered from standard attention by computing queries and keys from one-hot positional encodings and values from one-hot sequence embeddings.

**Single attention layer.** Given a sequence of dense vectors $x = (x_1, \dots, x_L)$ with $x_i \in \mathbb{R}^p$, the attention mechanism of the Transformer encoder (multihead scaled dot product self-attention) produces a continuous representation $y \in \mathbb{R}^{L\times p}$. If head size is $d$, this representation is computed using $H$ heads $(W_Q, W_K, W_V)$, where $W_Q, W_K, W_V \in \mathbb{R}^{p\times d}$. Queries, keys, and values are defined as $Q = xW_Q, K = xW_K, V = xW_V$. For a single head $(W_Q, W_K, W_V)$, the output is given by

$$y = \text{softmax}\Big(\frac{QK^T}{\sqrt{d}}\Big)V.$$

The full output in $\mathbb{R}^{dH}$ is produced by concatenating all head outputs. A single Transformer encoder layer passes the output through a dense layer, applying layer-norms and residual connection to aid optimization.

For the first layer, the input $x$ is a sequence of discrete tokens. To produce a dense vector combining sequence and position information, positional encodings and sequence embeddings are combined. The positional encoding $E_{pos} \in \mathbb{R}^{L\times e}$ produces a dense vector of dimension $e$ for each position $i$. The sequence embedding $E_{seq} \in \mathbb{R}^{A\times e}$ maps each element of the vocabulary to a dense vector of dimension $e$. Typically these are combined through summation to produce a dense vector $\tilde{x}_i = E_{seq}(x_i) + E_{pos}(i)$, which is input to the Transformer as described above.

For this paper, we use only multi-head self-attention without the dense layer, layer norm, or residual connections, as these drastically hurt performance when employed for one layer.

**Factored attention from standard attention.** Written explicitly, the input Transformer layer computes queries for a single head with $Q = (E_{pos} + E_{seq}(x))\, W_Q$. Keys and values are computed similarly. To recover factored attention, we instead compute queries and keys via $Q = E_{pos}W_Q$ and $K = E_{pos}W_K$, while values are given by $V = E_{seq}(x)W_V$. For simplicity, we one-hot encode both position and sequence, which corresponds using identity matrices $E_{pos} = I_L \in \mathbb{R}^{L\times L}$ and $E_{seq} = I_A \in \mathbb{R}^{A\times A}$.

**Implicit single-site term in single-layer attention.** For a single layer of attention, the product $E_{pos}W_V$ is a matrix in $\mathbb{R}^{L\times A}$. This matrix does not depend on sequence inputs, thus allowing it to act as a single-site term. This suggests why inclusion of an explicit single-site term in Figure A10 had no effect for single-layer attention.