Direct Coupling Analysis and the Attention Mechanism

Francesco Caredda[1†] and Andrea Pagnani[1,2,3†]

[1]DISAT, Politecnico di Torino, Corso Duca degli Abruzzi, 24, I-10129, Torino, Italy
[2]Italian Institute for Genomic Medicine, IRCCS Candiolo, SP-142, I-10060, Candiolo, Italy
[3]INFN, Sezione di Torino, Torino, Via Pietro Giuria, 1 10125 Torino Italy

Abstract

Proteins serve as the foundation for nearly all biological functions within cells, encompassing roles in transport, signaling, enzymatic activity, and more. Their functionalities hinge significantly on their intricate three-dimensional structures, often posing challenges in terms of difficulty, time, and expense for accurate determination. The introduction of AlphaFold 2 marked a groundbreaking solution to the enduring challenge of predicting a protein's tertiary structure from its amino acid sequence. However, the inherent complexity of AlphaFold's architecture presents obstacles in deciphering its learning process and understanding the decision-making that ultimately shapes the protein's final structure.

In this study, we introduce a shallow, unsupervised model designed to understand the self-attention layer within the Evoformer block of AlphaFold. We establish a method based on Direct Coupling Analysis (DCA), wherein the interaction tensor undergoes decomposition, leveraging the same structure employed in Transformer architectures. The model's parameters, notably fewer than those in standard DCA, are interpretable through an examination of the resulting attention matrices. These matrices enable the extraction of contact information, subsequently utilized for constructing the contact map of a protein family. Additionally, the self-attention decomposition in the DCA Hamiltonian form adopted here facilitates the definition of multi-family learning architecture, enabling the inference of parameter sets shared across diverse protein families. Finally, an autoregressive generative version of the model is implemented, capable of efficiently generating new proteins in silico. This generative model reproduces the summary statistics of the original protein family while concurrently inferring direct contacts in the tertiary structure of the protein. The effectiveness of our Attention-Based DCA architecture is evaluated using Multiple Sequence Alignments (MSAs) of varying lengths and depths, with structural data sourced from the Pfam database.

1 Introduction

Proteins constitute a diverse category of biological compounds constructed from a set of 20 amino acids. Within an organism, they serve various functions, including structural support, mobility,

and enzymatic activities. The effectiveness of a protein is intricately linked to its three-dimensional arrangement, known as its tertiary structure. This structure dictates the protein's biological functionality when isolated and its interactions with other molecules within the cellular environment. Despite the multitude of tasks proteins can perform, they lack a standardized three-dimensional conformation. Nonetheless, under physiological conditions, a protein's three-dimensional configuration is influenced by its amino acid sequence [1]. Determining this dependence is challenging both theoretically and computationally due to the complexity of the system involved. Only recently, the problem of predicting the folding of an amino acid sequence has seen a historical computational breakthrough thanks to *AlphaFold-2* in 2020 [2].

AlphaFold-2 exploited decades of research in computational biology together with recent developments in machine learning. The most fundamental idea, which has been the center of this research area for years, is that evolutionary information can be extracted and used to determine patterns in phylogenetically related protein sequences (viz. homologs). Indeed, in the course of evolution, to preserve the functionality of a class of proteins, their structure must be conserved. Natural selection imposes constraints on single active sites or multi-amino acid motifs which are fundamental for the correct folding of the sequence. This leads to the idea of conservation and co-evolution [3], [4]. Given a Multiple Sequence Alignment (MSA) [5], single- and pair-wise frequencies of amino acids along different positions are enough to extract summary statistics that can be used to determine structural information by inferring the parameters of a Potts model, in what is commonly known as Direct Coupling Analysis (DCA) [6], [7]. The inference has been done in various ways and with various degrees of approximation during the last decade, leading to algorithms capable of determining the contact map of a protein family (PlmDCA, [8]), but also generating in-silico sequences representative of the full statistics of the original protein family (bmDCA [9], [10], ArDCA [11]).

A similar strategy was introduced in *AlphaFold-2* [2], where the self-attention mechanism [12], [13] allows for a direct representation of correlations over the MSA, albeit through multiple layers. The attention mechanism was originally introduced in the context of Natural Language Processing (NLP) to overcome the limitations of sequential encoder-decoder architectures [14]. The basic idea is that long-range correlations within a dataset can be captured by a so-called *attention map*, encoding a custom functional relation between features of the dataset. For instance, in NLP the correlations to capture emerge at the semantic level [14], whereas in the case of structural protein inference from homology data, at the level of the individual residues as in the *Evoformer* architectural block in *AlphaFold-2*, in which a contact representation of the residues in a sequence is updated by conservation and co-evolution information extracted from an MSA and processed through self-attention layers.

Following the initial work of Bhattacharya et al. [15], and the recent theoretical evaluation using the Replica Method by Rende et al. [16], we present a Direct Coupling Analysis architecture in which the Potts model is written in the form of *factored self-attention*, where the degrees of freedom of each position in the amino acid sequence is decoupled from the *color* degrees of freedom representing each possible amino acid. The inference of the trainable parameters in the model is performed by a maximum (pseudo-) likelihood principle, following the work discussed in [8].

2 Materials and Methods

2.1 Factored Attention

As in every Direct Coupling Analysis implementation, sequences in a protein family are represented by a Multiple Sequence Alignment and can be thought of as independent samples (cf. Supplementary Materials at 4 for details) from a probability distribution that we model as a Gibbs-Boltzmann measure over a Hamiltonian function defined as a Potts model [7]:

$$P(\boldsymbol{a}) = \frac{1}{Z} e^{-H(\boldsymbol{a})} \quad , \tag{1}$$

$$H(\boldsymbol{a}) = -\sum_{i,j} J_{ij}(a_i, a_j) - \sum_{i=1}^{L} h_i(a_i) \quad , \tag{2}$$

where, $\mathbf{a} = (a_1, \ldots, a_L)$ is a sequence of L amino acids (a_i take value in an alphabet of 21 letters), J, and h are respectively the direct interaction tensor and the local field terms, while Z is a normalization constant, also known as partition function in the Statistical Physics jargon, given by:

$$Z = \sum_{\boldsymbol{a} \in \{1,\ldots,q\}^L} e^{-H(\boldsymbol{a})} \quad , \tag{3}$$

where $q = 21$ is the length of the amino acid dictionary, corresponding to the 20 natural amino acids plus a gap sign used during the alignment procedure in building the MSA. The inverse temperature β, which usually multiplies the energy term, is set equal to one, which is equivalent to implying its dependence directly inside terms J and h.

At this generic stage, tensor $J \in \mathbb{R}^{L,L,q,q}$ encodes both positional and amino acid information, while $h \in \mathbb{R}^{L,q}$ represents the local biases of each position in the sequence. To implement a factored attention mechanism, we discard the local field term, which is possible due to a gauge invariance of the parameters, and we write the interaction tensor using the popular transformer [12] implementation of the attention mechanism for which:

$$J_{ij}(a_i, a_j) = \sum_{h=1}^{H} \text{softmax}(Q^h {K^h}^T)_{ij} V^h_{a_i,a_j} = \sum_{h=1}^{H} A^h_{ij} V^h_{a_i,a_j} \quad , \tag{4}$$

where, using the jargon from NLP, $Q^h \in \mathbb{R}^{L,d}$, $K^h \in \mathbb{R}^{L,d}$, $V^h \in \mathbb{R}^{q,q}$ are respectively the *query*, *key* and *value* matrices in one of H attention heads that build the interaction tensor J. The `softmax` is performed column-wise and the resulting matrix A_{ij}, $i, j \in 1, \ldots, L$ (such that for all $i \in 1, \ldots, L$, $\sum_{j=1}^{L} A_{ij} = 1$) represents the self-attention of each pair of residues. The attention matrix in this form is meant to highlight the co-evolution relationships between different positions in a sequence, meanwhile a $M \times M$ row-attention would concern phylogenetic relations between different sequences in the same MSA; for more on this, cf. Rao et al. [17], Sgarbossa at al. [18].

A major implication of the decomposition of the interaction tensor into matrices Q, K, and V is that we are factoring the positional information contained inside matrices Q and K, depending

on the specific protein family at hand, and the information contained inside matrix V which should capture the more universal traits characterizing the interactions between the twenty natural amino acids. This turns out to be an interesting point that can be further expanded in the future and which plays an already crucial point in the multi-family version of the model discussed in section 2.3.

To infer the parameters, i.e. train the model, we implement a pseudo-likelihood approximation: the full probability distribution is given by a factorization into single-site distributions conditioned to all the other sites in the sequence. In this way the full pseudo-likelihood of the model over an MSA $\mathcal{D} = \{a_{ij}\} \in \mathbb{R}^{L,M}$ with M sequences (depth of the MSA) each of L amino acids (length of the MSA) can be written as

$$\begin{aligned}
\mathcal{L}(\mathbf{J}|\mathcal{D}) &= \frac{1}{M}\sum_{m=1}^{M} \log P(\mathbf{a}^m|\mathbf{J}) \\
&= \frac{1}{M}\sum_{m=1}^{M} \log \frac{\exp\Big\{\sum_{j\neq i} J_{ij}(a_i^m, a_j^m)\Big\}}{\sum_{a=1}^{q}\exp\Big\{\sum_{j\neq i} J_{ij}(a, a_j^m)\}} \\
&= \frac{1}{M}\sum_{m=1}^{M}\left\{\sum_{j\neq i} J_{ij}(A_i^m, A_j^m) - \log\Big[\sum_{a=1}^{q}\exp\Big\{\sum_{j\neq i} J_{ij}(a, a_j^m)\Big\}\Big]\right\} \quad .
\end{aligned} \tag{5}$$

Finally, following what was done for PlmDCA [8], after an analysis of different regularization schemes we added an L2-regularization of the interaction tensor. This penalizes large values of the J_{ij} matrix to avoid overfitting the direct coupling score, defined in the following Equation 7. Thus, the total final likelihood is given by:

$$\tilde{\mathcal{L}}(\mathbf{J}, \mathcal{D}) = \mathcal{L}(\mathbf{J}, \mathcal{D}) + \lambda \sum_{i,j,a,b} J_{ij}^2(a, b) \quad . \tag{6}$$

The decomposition of the interaction tensor given by Equation 4 has the side effect of making the total log-likelihood a non-convex-up function of its parameters, mainly due to the matrix products and the softmax function. Compared to models such as PlmDCA or ArDCA, this makes the maximization procedure much more challenging. To smooth out the complex landscape of peaks and troughs of the total likelihood during the maximization procedure, we implement an ADAM stochastic gradient ascent over mini-batches of fixed size. [19].

Once the optimization is concluded and the parameters have been inferred, the actual contact prediction is performed by computing the average-product-corrected Frobenious norm of the interaction tensor which is interpreted as a score of the direct interaction between any position pair (i, j) in the chain [20]:

$$F_{ij} = \sqrt{\sum_{a_i, a_j} J_{ij}^2(a_i, a_j)} \quad . \tag{7}$$

The higher the score, the more likely two pairs are to be an actual contact in the structure. When testing the model on some protein family whose structure is known, we define the Positive Predicted

Value (PPV) as the percentage of true positive (TP) contacts among the predicted ones: $\mathrm{PPV}(n) = \mathrm{TP}(n)/n$. The curve given by this measure, a function of n, represents the global accuracy of the model.

Since we are dealing with a heavily non-convex optimization problem, the results for different runs are generally affected by intrinsic noise. To reduce this, for each MSA we run the model several times and we build a final interaction score that represents a merging between the scores of each run. This is done by picking for each position pair its maximum score among all the runs. In the section 3 we compare PPV curves from single runs and merged multiple runs of AttentionDCA.

2.2 Parameter reduction and hyper-parameters

Concerning the hyper-parameters of the learning process, we use a learning rate $\eta = 0.005$ and a mini-batch size $n_b = 1000$, while the number of epochs varies depending on the depth of each MSA, cf. table 2. A more insightful analysis is that on the hyper-parameters of the model itself, i.e. the number of heads H used to define the interaction tensor and the lower dimension d of the rectangular matrices Q and K. In standard DCA models, the size of the interaction tensor scales quadratically with the length of the protein family. However, empirically the number of contacts in a protein is proportional to its length L (and not to L^2) [21], therefore we can expect the interaction tensor to be somehow sparsified. In AttentionDCA, due to the low-rank decomposition of the interaction tensor, we obtain a linear scaling of the parameters with the size of the family L: $N_{\mathrm{AttentionDCA}} = 2HLd + Hq^2$. The parameter compression ratio concerning PlmDCA can be computed by fixing the value of d and H:

$$c_r := \frac{\text{number of parameter AttentionDCA}}{\text{number of parameters PlmDCA}} = \frac{2HdL + Hq^2}{L(L-1)q^2/2 + Lq} \quad . \tag{8}$$

During the process of tuning the hyper-parameters of the model, we observe that, at a fixed compression ratio, the precision of the contact prediction is constant regardless of the choice of H and d. Therefore the number of parameters determines the quality of the model. However, increasing the compression ratio, i.e., lowering the distance between PlmDCA and AttentionDCA in terms of the number of parameters, does not improve the contact prediction's accuracy indefinitely. Indeed, having the same number of parameters of PlmDCA results in sensibly worse results. This may be because the likelihood in this architecture does not have a clear absolute maximum since the function is non-convex-up. Increasing the parameters enhances this condition making the inference harder and harder. After some trials, we conclude that the optimal parameter compression lies between 5% and 20%. Focusing on families PF00014, PF00076, and PF00763 respectively, Figure 1 shows the PPV evaluated at L, $3L/2$, and $2L$ for different values of the head number H and the lower dimension d so that the parameter compression is fixed at values $c_r = \{0.05, 0.15, 0.25, 0.35\}$. It can be seen how a plateau is reached for $H > 10$, effectively highlighting the fact that performances remain constant at fixed compression, regardless of the hyper-parameters H and d. Another interesting feature is that for $H \lesssim 10$, the value of the PPV is significantly lower than otherwise. This is due to a non-reliable inference of the parameters, as it can be shown that soon after the beginning

of the gradient ascent, it hits a barrier and stops.

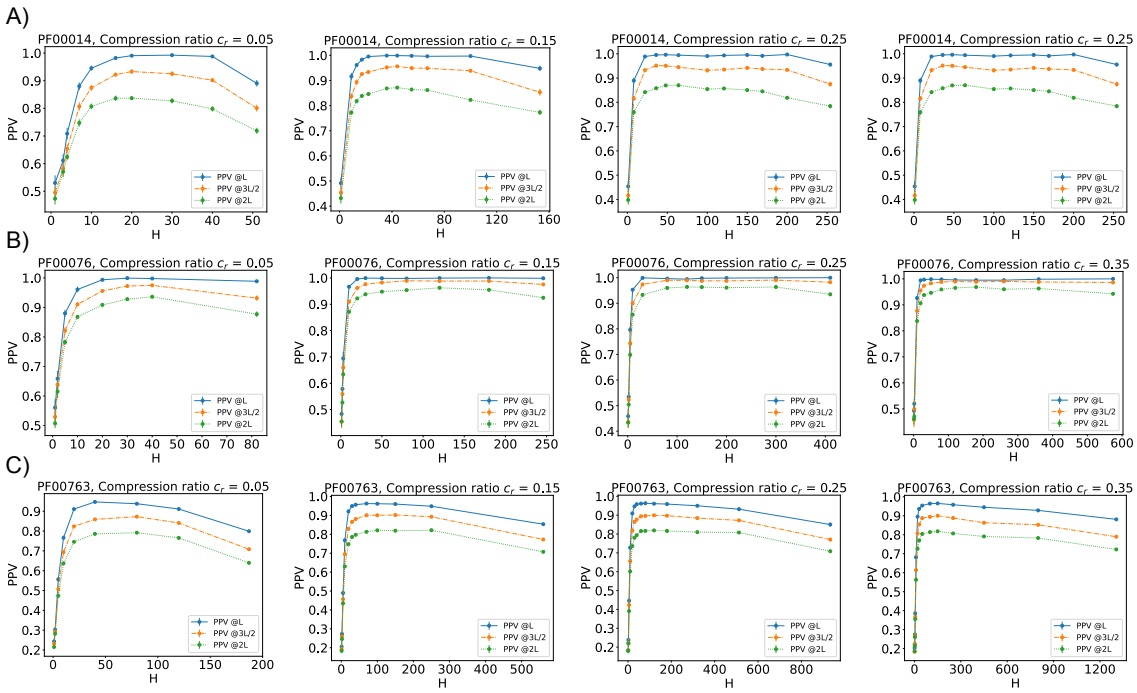


Figure 1: PPV @L, 3L/2 and 2L (blue, orange and green curve respectively) for different values of the compression ratio r for families PF00014, PF00076 and PF00763 respectively in panels A), B) and C). For each value of H there is a corresponding value of d given by Equation 8 at fixed c_r.

2.3 Multi-Family Learning

A long-standing goal of protein design is that of being able to selectively pick features from different protein families and generate sequences at first in-silico and then in-vivo so that these new artificial proteins reproduce those specifically chosen traits and functions [22], [23]. The first step toward this feat is that of designing a DCA model that can learn simultaneously from different MSAs. Standard DCA is not prone to this possibility since there is no obvious way to determine a set of parameters that can be shared among different families since both the interaction tensor and the local fields are family-dependent. However, in the Factored Attention implementation of DCA, the Value matrix $V \in \mathbb{R}^{q,q}$ does not depend on the specific family and therefore can be shared and learned across different MSAs. Given N_F protein families, we obtain a simple implementation of this just by defining a multi-family likelihood $\mathcal{L}_{MF}$ given by the sum of the N_F single-family likelihoods sharing the same set of Value matrices $\{V^h\}_{h=1,\dots,H}$:

$$\mathcal{L}_{MF} = \sum_{a=1}^{N_F} \mathcal{L}_a(\{Q_a^h, K_a^h, V^h\}, \mathcal{D}_a) \quad , \tag{9}$$

where each single-family likelihood is given by Equations 4 and 5. The set of parameters $\{Q_a^h, K_a^h\}$, one for each family, and $\{V^h\}$, shared across all families, can be used to extract the contact score of each MSA used during the learning. The $\{Q_a^h, K_a^h\}$ matrices must each have the same number of heads so that it is possible to share a common set of $\{V^h\}$ matrices among them. In section 3 we show the results of this PPV compared to those obtained from standard single-family learning.

2.4 Generative model

The maximum pseudo-likelihood criterion used for the inference in PlmDCA and AttentionDCA does neither provide a fast way to sample the probability distribution nor an accurate sample due to the approximation used to factorize the probability distribution. Because of this, to develop the generative version of the model we started from ArDCA [11], which is so far the state of the art for generative DCA architectures. In particular, it exploits a simple autoregressive model enforced by the exact decomposition

$$P(\mathbf{a}) = P(a_1)P(a_2|a_1)\cdots P(a_i|a_1,\ldots,a_{i-1})\cdots P(a_L|a_1,\ldots,a_{L-1}) \quad , \tag{10}$$

with

$$P(a_i|a_1,\ldots,a_{i-1}) = \frac{1}{Z_i(a_{<i})}\exp\Big\{\sum_{j<i} J_{ij}(a_i,a_j)\Big\} \quad , \tag{11}$$

$$Z_i(a_{<i}) = \sum_{a=1}^{q}\exp\Big\{\sum_{j<i} J_{ij}(a,a_j)\Big\} \quad . \tag{12}$$

This form allows for a fast computation of the single-site partition function and a quick autoregressive sampling from L single-valued distributions.

To exploit the already existing libraries of ArDCA and the architecture for AttentionDCA, we apply a multiplicative mask to the interaction tensor to implement an autoregressive structure of the distribution, i.e. so that J_{ij} is a lower-triangular matrix with zeros above and on the diagonal:

$$J_{ij}(a_i,a_j) = \begin{cases} \sum_{h=1}^{H} \mathrm{softmax}(Q^h {K^h}^T)_{ij} V^h_{a_i,a_j} & i > j \\ 0 & i \leq j \end{cases} \quad . \tag{13}$$

As discussed in Trinquier et al. [11], the resulting interaction tensor cannot be interpreted as a matrix of direct couplings in the same way as standard DCA, mainly because in the autoregressive implementation, the interactions of each position in the chain are conditioned only to partial sequences instead of being conditioned to all other amino acid positions. Therefore, the common technique to predict the residue-residue contacts is to extract coupling information directly from the epistatic score evaluated from the model. For amino acids b_i, b_j at positions (i,j), their epistatic score is defined as the difference between the effects of simultaneous mutations on both sites and

the sum of the single site mutations when introduced in the wild-type $\mathbf{a} = (a_i, \ldots, a_j)$ as in:

$$ES(b_i, b_j) = \Delta E(a_i \to b_i, a_j \to b_j) - \Delta E(a_i \to b_i) - \Delta E(a_j \to b_j) \quad , \tag{14}$$

where the energy difference for a given mutation $a_i \to b_i$ is given by:

$$\begin{aligned} \Delta E(a_i \to b_i) &= E(a_1, \ldots, b_i, \ldots, a_L) - E(a_1, \ldots, a_i, \ldots, a_L) \\ &= -\log \frac{P(a_1, \ldots, b_i, \ldots, a_L)}{P(a_1, \ldots, a_i, \ldots, a_L)} \quad . \end{aligned} \tag{15}$$

The Epistatic Score $ES_{i,j}$ replaces the Frobenious Score F_{ij} used in the non-autoregressive implementation of the model so that a contact map can be produced and compared with the true structure of the protein. An alternative to using the Epistatic Score to produce a contact score from the generative model is that of sampling an artificial MSA to be fed directly to a non-generative version, either PlmDCA or AttentionDCA, to define a meaningful Frobenious Score. Both alternatives will be discussed in the next section.

3 Results

The model, in its standard and autoregressive implementations, has been tested against nine protein families taken from the Protein Data Bank. Each family is represented by a MSA whose length L (number of amino acids per sequence) and depth M (number of sequences per MSA) vary, table 1 summarises some of the main information regarding each family. A crucial parameter is the effective depth M_{eff} that quantifies the effective number of independent sequences in the MSA. As we show in the following, the effective depth turns out to be crucial in determining the accuracy of the Epistatic Score contact prediction for the autoregressive generative version of the model, cf. 3.3.

3.1 Standard version

3.1.1 Contact Prediction

The standard way to perform the contact prediction within the DCA methods is to compute the Frobenious norm of the inferred interaction tensor after a transformation which brings the parameters in zero-sum gauge, the gauge that minimizes the Frobenious norm, and a consecutive *average product correction*. Sorting the Frobenious Score (FS) results in a list of contacts that can be compared to the actual known structure of the protein family to produce a PPV curve. Alternatively, the factored attention form of the interaction tensor suggests another way of determining the contacts of a protein family. Averaging the positional-dependent attention matrix through each head results in a matrix of the form:

$$A_{ij} = \sum_{h=1}^{H} A_{ij}^{h} = \sum_{h=1}^{H} \mathrm{softmax}(Q^h {K^h}^T)_{ij} \quad . \tag{16}$$

After the *average product correction*, the contact prediction is given by the list of contacts (i, j) sorted by highest A_{ij}. As already mentioned, the non-convexity of the problem and the roughness of

the energy landscape produce a noise in the final inference which can be smoothed out by averaging through different realizations. In particular in the case of the FS contact prediction we compute the score for each realization of the inference and, for each position pair (i, j), we define its score as the maximum through all realizations: $F_{ij} = max\{F_{ij}^1, F_{ij}^2, \ldots, F_{ij}^m\}$ among the resulting ones, so that to produce a final Merged Scored which results as the most likely. In the case of the Attention contact prediction, it is sufficient to simply average the resulting attention matrices of each realization.

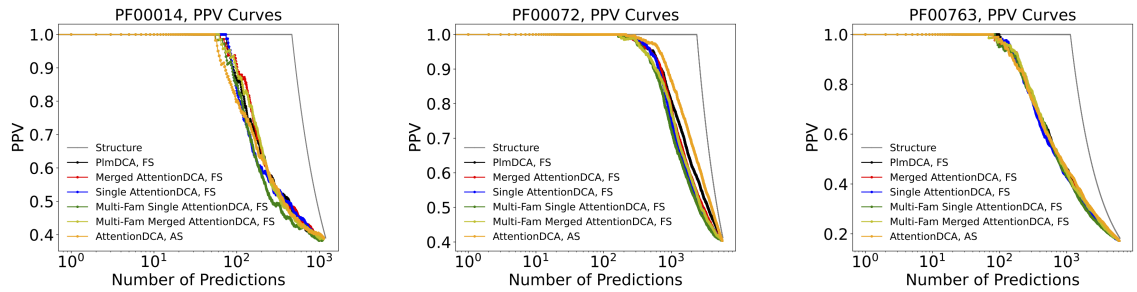


Figure 2: Positive Predicted Value Curves for families PF00014, PF00072, PF00763 respectively. Each plot compares PPVs computed through the Frobenious Score (FS) and the Attention Score (AS), in particular: PlmDCA with FS (black curve); merged and single AttentionDCA with FS (red and blue curves respectively); Multi-Family AttentionDCA with merged and single realization (green and yellow curves respectively); AttentionDCA with AS. The gray curve represents the PPV of an ideal model that predicts all and only positive contacts, hence it depicts the prediction of the full structure of the family.

Figures 2 shows the comparison between the PPV curves extracted from different implementations of AttentionDCA and those from PlmDCA (black curve), which serves as a benchmark model, applied to families PF00014, PF00072 and PF00763. Grey curves represent an ideal perfect model which predicts all and only positive contacts, while the red and blue curves are respectively the FS PPV from the merged- and single-realization of AttentionDCA. These results have been obtained by setting the compression ratio between PlmDCA and AttentionDCA to 20%, and for each family, apart from fluctuations in the single-realization curves, the results are in very strong agreement with PlmDCA, despite the significant parameter reduction between the two models. Dark and light green curves represent the PPVs evaluated from the multi-family Leaning discussed in section 2.3. In particular, the learning has been performed in parallel among all families using a fixed number of heads $H = 128$ and varying the lower dimension d to match the single-family learnings at 20% compression ratio. The shown curves are obtained through the Frobenious Score computed on the J_{ij} tensor which is given by Equation 4 and parameters $\{Q_F^h, K_F^h V_{\text{shared}}^h\}_{h=1,\ldots,H}$, for $F = \{\text{PF00014}, \text{PF00072}, \text{PF00763}\}$.

3.2 Attention Heads

In the field of Natural Language Processing [24], the self-attention mechanism is used to learn custom functional relationships between the elements of a dataset to produce an encoded description of the input itself [25]. In the context of protein structure prediction, the attention mechanism used

in AlphaFold2 untangles the physical and evolutionary constraints between phylogenetically related sequences [26], [27]. Since our model is effectively a single-layer self-attention architecture, we expect the attention heads inside the J_{ij} tensor to directly capture some level of structural information that could be compared to the actual contact map obtained by the average-product-corrected Frobenious norm of the interaction tensor. Ideally, one would expect each head to focus its attention on a specific spatial structure in the same way as a transformer head would specialize in different semantic functions when used in the context of NLP [28]. However, it is extremely difficult to determine if this is the case for protein attention due to the lack of an obvious interpretation key between the several possible structures arising in a protein and the variable number of heads. Also, given the form of Equation 4, a single head generally produces an asymmetric attention matrix, hence it is hard to understand which are the structures emerging from each one of them. A solution to both these problems is to simply average out all heads into a single attention matrix as in Equation 16.

The average attention matrix can be used to extract structural information by producing a list of contacts as discussed in subsection 3.1.1 and shown by the orange curves in Figure 2. However, the most direct way in which the average attention matrix can be interpreted is by comparing it to the actual contact map of the protein family. Figure 3 compares the contact map built from a significant fraction of predicted contacts and the heat map of the attention matrix for families PF00014, PF00072, and PF00763. In the contact maps, blue and red dots represent the positive and negative predictions, while gray dots show the actual structure of the family. Conversely, the heat maps are displayed on a grayscale, while the actual structure is given by the red dots underneath. It can be seen that the maxima of the attention matrix, i.e. the darker dots, lie within the structure of the protein, proving that the attention model is focused on the structural contacts.

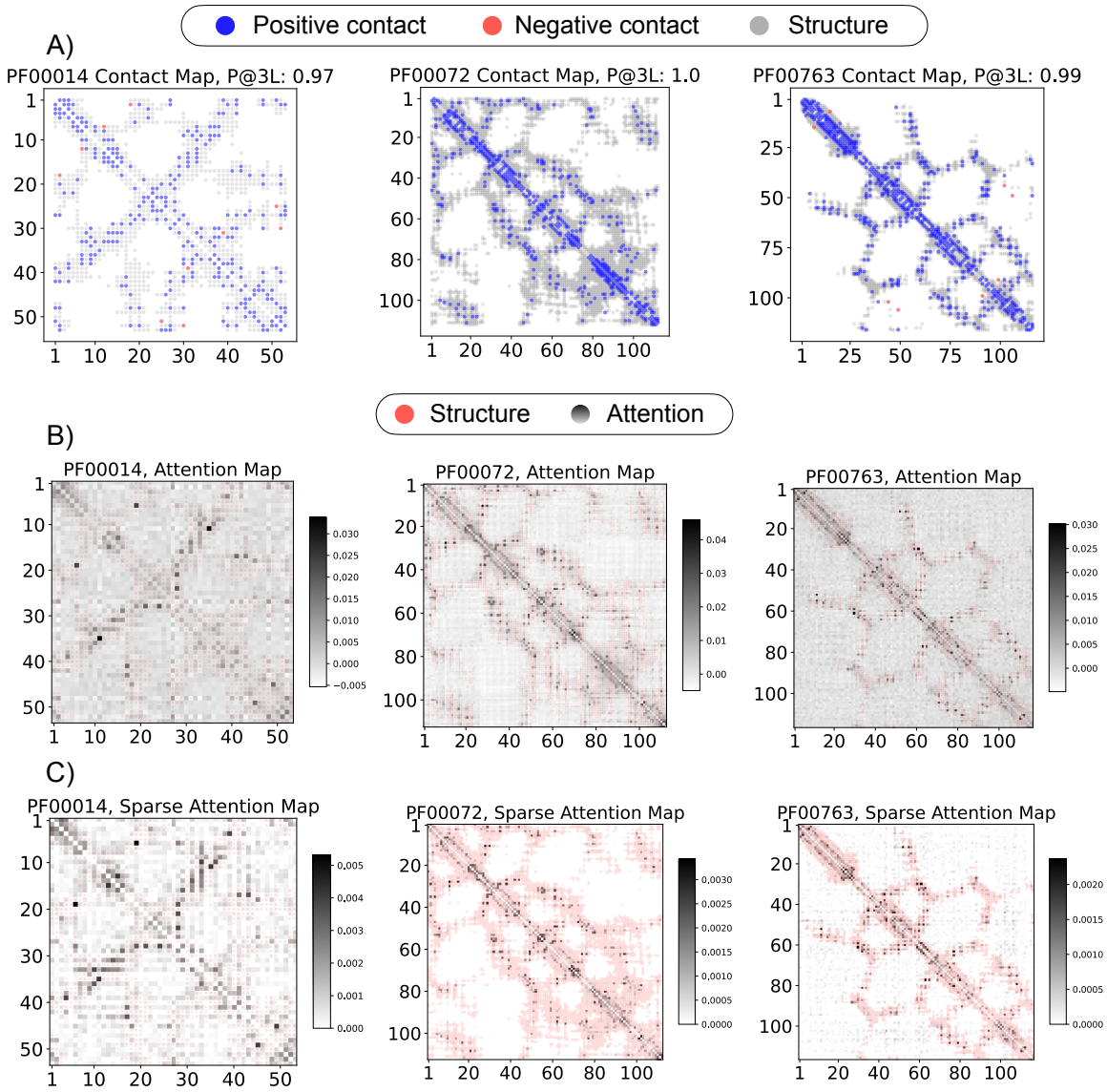


Figure 3: Various contact maps for families PF00014, PF00072, PF00763. A) Contact map built from the Frobenious Score applied to the interaction tensor by AttentionDCA. Blue and red dots represent positive and negative predictions respectively, while gray dots reproduce the structure of the family. B) Grayscale heat map of the averaged attention matrix built using Equation 16, from the $\{Q^h, K^h\}_{h=1,\dots,H}$ matrices by AttentionDCA. C) Grayscale heat map for the sparse attention matrix constructed by maximizing over the k highest elements from each head, using $k = L$, as in Equation 17. Both in B) and C) structural information is depicted in red.

Even though, as already mentioned, the specialization of single attention heads is not trivial to understand, an important feature that can be observed is the sparsification of the actual structural information contained in each one of them. A quantitative analysis of the sparsity of the attention heads can be as follows: given $\{\boldsymbol{A}^h\}_{h=1,\dots,H}$ (such that for each h, $A^h_{ij} \geq 0, i, j = 1, \dots, L$), for each head we select the k highest elements. This results in a list S of $k \times H$ elements that can be used to

build a new attention matrix $[\tilde{A}]_{i,j=1,\dots,L}$ either by maximizing or averaging for each position pair (i,j) over the subset of $S_{(i,j)} \subset S$ which contains all the elements relative to that particular (i,j):

$$\tilde{A}_{i,j} = \begin{cases} \max\left(A^h_{ij} | A^h_{ij} \in S_{(ij)}\right) \\ \frac{1}{|S_{ij}|} \sum_{A^h_{ij} \in S_{(ij)}} A^h_{ij} \end{cases} . \tag{17}$$

The resulting matrix is then *average-product*-corrected, symmetrized, and used as a contact score to produce a PPV curve and a contact map. By varying the value of k, i.e. the number of elements selected from each attention head, it is possible to evaluate the sparsity of the matrices by comparing the positive predicted values relative to each k. Figure 4 shows the PPV evaluated at L and $2L$ for increasing values of k for families PF00014, PF00072 and PF00763. Results are compared with the corresponding values from the standard Frobenious norm approach by AttentionDCA.

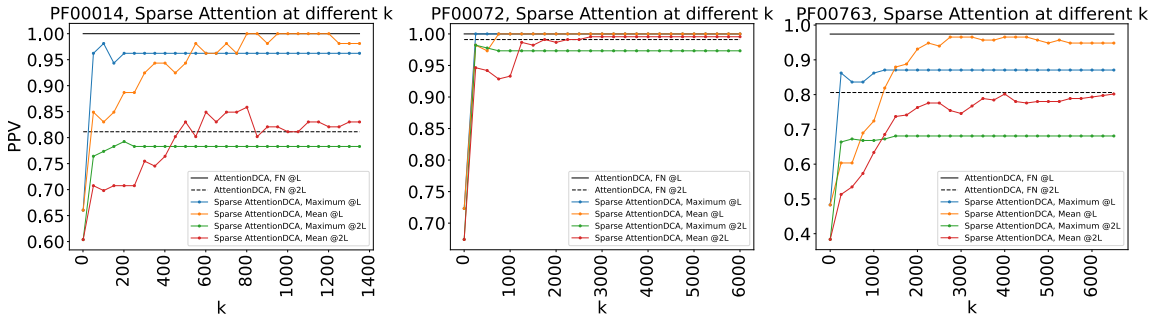


Figure 4: PPV curves @L and 2L for the average- and maximum-sparse Attention Score with as the value of the number of elements k from each varies. Blue and orange curves represent the PPV @L for the average- and maximum-sparse version respectively and are to be compared to the full black line representing standard AttentionDCA computed using the Frobenious Score (FS). Similarly, green and red curves represent the PPV @2L and are to be compared to the dotted black line. Families PF00014, PF00072 and PF00763 are shown.

It can be seen that when using the maximum to construct the sparse matrix, a very small fraction of the total elements is enough to return a PPV that is close to that obtained from the Frobenious norm. On the other hand, the PPV computed from the mean-version of the sparse matrix is slower to grow, but again for $k << k_{max} = \binom{L}{2}$ it reaches the accuracy of standard FS PPVs. This means that regardless of the version used for constructing $\tilde{A}$, a small number of elements is enough to capture structural information comparable to the full information obtained by the Frobenious norm. Figure 5 shows the full PPV curves from sparse attention matrices computed on a restricted number of significant values of k: $k = \{1, L/10, L, 4L, \binom{L}{2}\}$. Furthermore, Figure 3 c) shows the heatmap of the sparse attention matrix constructed using the maximum with $k = L$.

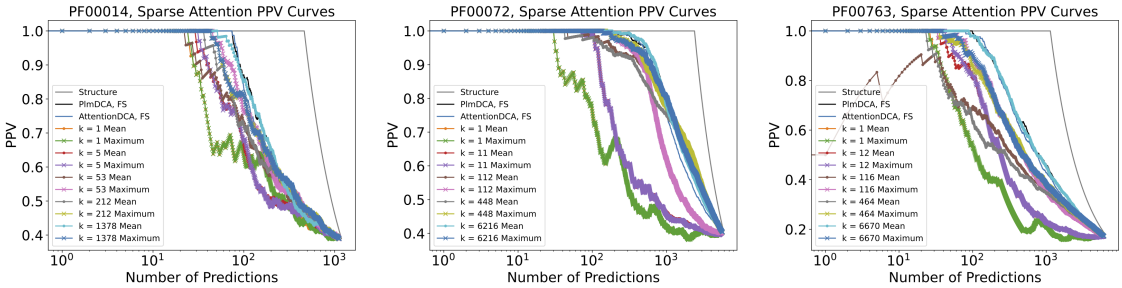


Figure 5: PPV curves of families PF00014, PF00072, PF00763 for the average- and maximum-sparse Attention Score computed for $k = \{1, L/10, L, 4L, \binom{L}{2}\}$ and compared to the standard Frobenious Score (FS) from PlmDCA and AttentionDCA (black and blue curves respectively).

3.3 Autoregressive generative version

Even though the autoregressive version of our architecture can be used for contact prediction, its main target is to efficiently sample the distribution to generate large MSAs matching the summary statistics of the natural MSAs used for inference.

As described in section 2.4, the form of the interaction term $J_{ij}(a, b)$ of the autoregressive version does not allow for a meaningful mapping to direct interactions among amino acids. Therefore, the epistatic score measure defined in Equation 14 must be used to define a list of predicted contacts and the corresponding PPV curve. Figure 8 shows different PPV curves computed on families PF00014, PF00072 and PF00763. In particular, it compares the performance of PlmDCA (black curve) and ArDCA (red curve), both acting as benchmarks and those of the autoregressive AttentionDCA in the single- and merged-realization implementations (blue and green curves respectively). Along with the epistatic measure, another way to perform a contact prediction is to produce a generated MSA, sampled using AttentionDCA, and feed it to PlmDCA or AttentionDCA itself (in its non-generative version). This last possibility is represented by the yellow curve and it is among the best options for contact prediction within the autoregressive versions, either AttentionDCA or ArDCA. This can also be thought of as a way to test the accuracy of the generative capabilities of the autoregressive model, which exploits the exact decomposition in Equation 10 to sample from L single-valued probability distributions. Moreover, the generativity can be assessed by comparing the summary statistics of natural and generated MSAs. The most informative summary statistics are given by the connected two-site correlations of amino acids in different positions of an MSA, cf. Supplementary Materials 4. Their comparison is shown in Figure 8 B), along with their Pearson's correlation coefficient, proving the accuracy of the model. Finally, a PCA analysis can be performed to show that the principal components of the generated data are comparable to those of the natural data. Figure 6 shows the comparison between the two principal components of the natural data and the artificial data generated by ArDCA, used as a benchmark model, and by AttentionDCA on families PF00014, PF00763 and PF13354 (cf. Supplementary Materials for the other families).

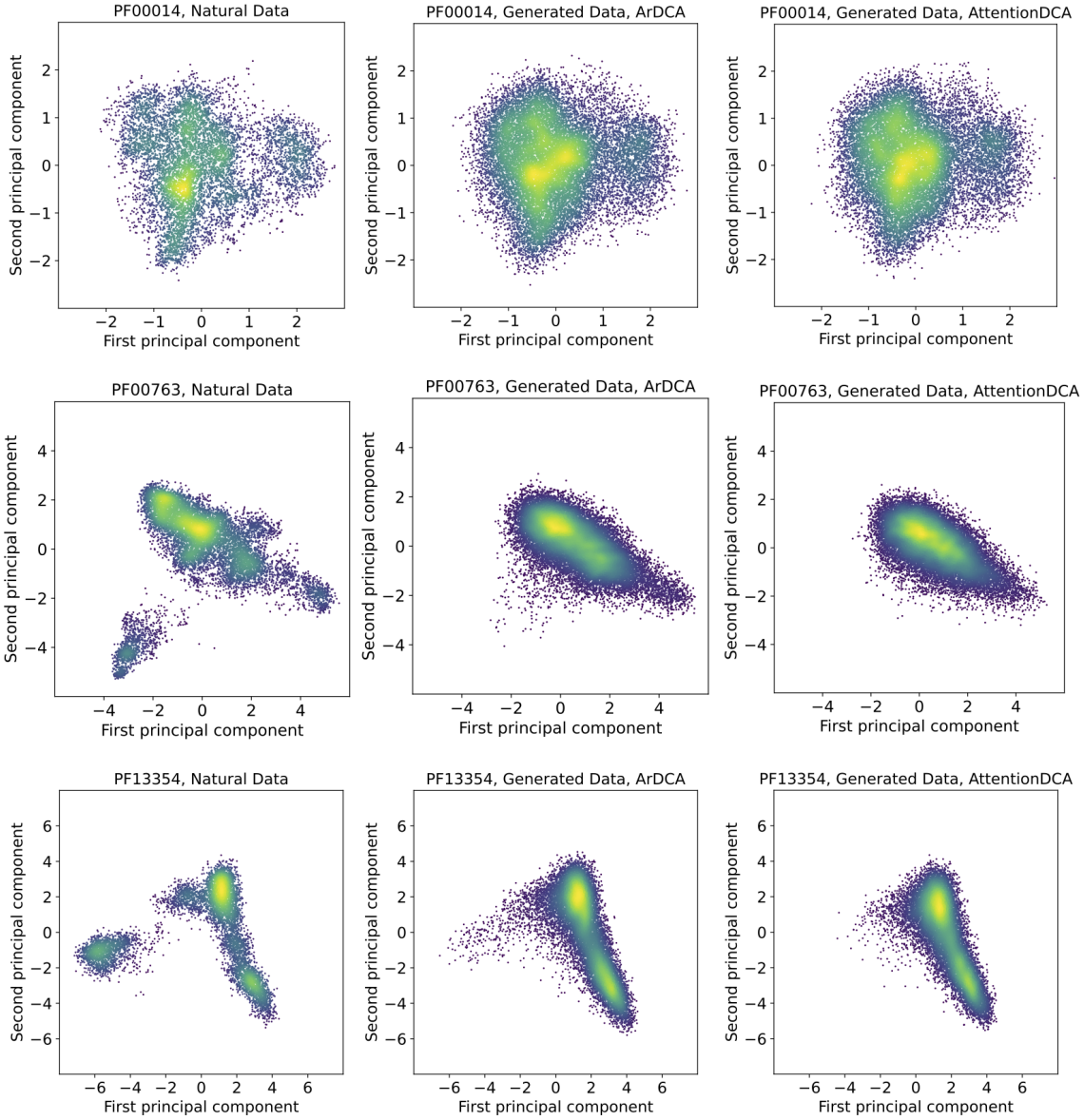


Figure 6: Principal Component Analysis on protein families PF00014, PF00763 and PF13354. Comparison between natural data and artificial data generated by using ArDCA and AttentionDCA.

For many families among the ones described in table 1, the results of the contact prediction from the epistatic score lie significantly below those from the Frobenious score of the standard version of AttentionDCA or PlmDCA. The reasons for this can be found in the dependency of the autoregressive version on the M_{eff} of the family. If one were to reduce the effective depth of an MSA, after a certain threshold, the DCA inference would suffer a lack of observations and the contact prediction would get less and less reliable. Different DCA models and different inference techniques have variable M_{eff} thresholds.

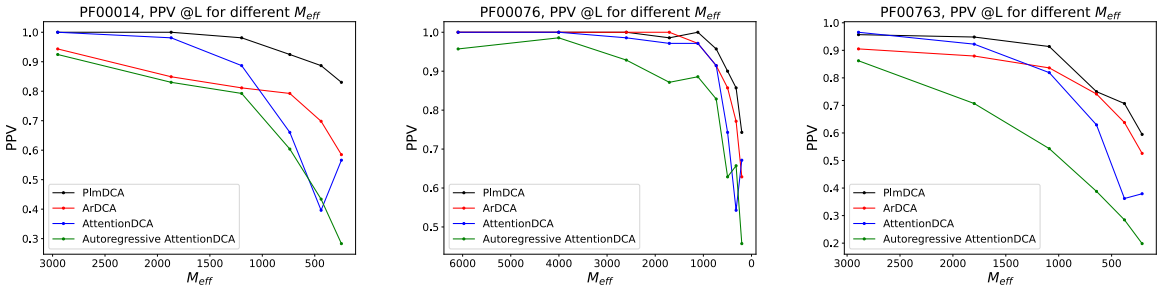


Figure 7: PPV curves @L as the M_{eff} of the family varies, computed using PlmDCA (FS), ArDCA (ES), AttentionDCA (FS) and Autoregressive AttentionDCA (ES). Families PF00014, PF00076, and PF00763 are shown.

Figure 7 shows how the accuracy of different DCA methods decays as a function of the effective depth for families of different lengths. While PlmDCA exhibits the best resilience when the effective depth is lowered, having an adequate accuracy even below $M_{eff} < 1000$, the autoregressive version of AttentionDCA degrades for $M_{eff} \leq 2000$. Because of this, performances on shallow families such as PF00595, PF13354 or PF00035 are intrinsically worse than those from PlmDCA and the standard version of AttentionDCA.

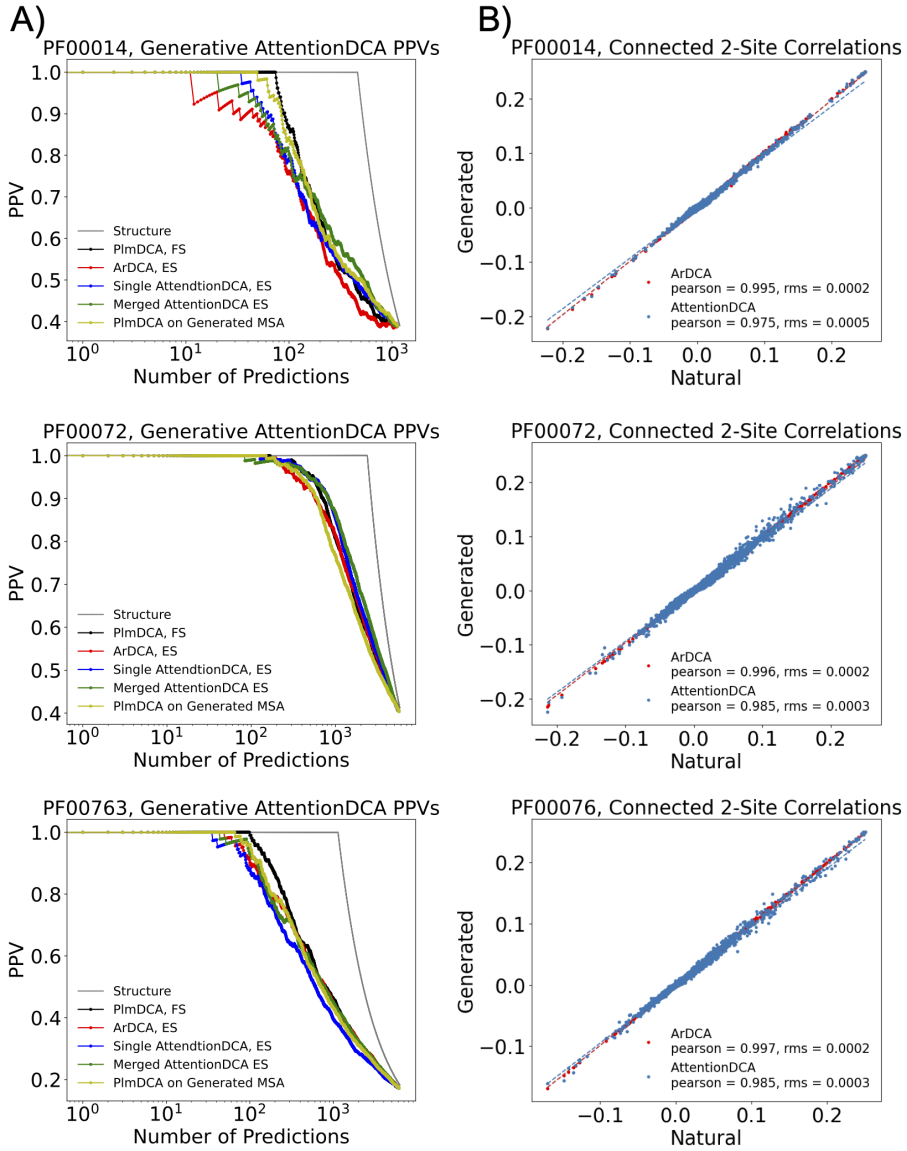


Figure 8: Results from Autoregressive AttentionDCA applied to families PF00014, PF00072 and PF00763. For each family, the panel shows the PPV curves and the connected 2-site correlations. A) The left plot compares the PPV curves from PlmDCA (FS, black curve), ArDCA (ES, red curve) with the single- and merged-realizations of AttentionDCA with ES (blue and green curves respectively) and the PPV from the FS of PlmDCA applied to a Generated MSA sampled from Autoregressive AttentionDCA. The gray curve represents the PPV of an ideal model that predicts all and only positive contacts, hence it depicts the prediction of the full structure of the family. B) The right plot shows a comparison between the 2-site connected correlations of the natural and artificial MSAs generated by Autoregressive AttentionDCA and ArDCA (blue and red dots respectively).

Table 1: Summary of the protein families used throughout the analysis and results (MSA's length, depth, effective depth, number of parameters in PlmDCA, number of parameters of AttentionDCA at $c_r = 0.2$, 2-site connected correlation Pearson coefficient for ArDCA and AttentionDCA, Frobenious score positive predicted values at L for PlmDCA and AttentionDCA).

	Description	L	M	M_{eff}	# params PlmDCA	# params AttentionDCA	ρ ArDCA	ρ AttentionDCA	PPV@L PlmDCA	PPV@L AttentionDCA
PF00014	Kunitz Domain	53	8871	2950	608,811	124,288	0.995	0.975	1.0	1.0
PF00035	DSRM	67	18462	2176	976,458	193,664	0.986	0.963	0.880	0.895
PF00072	RR	112	574565	59290	2,743,608	543,872	0.996	0.985	1.0	1.0
PF00076	RRM	70	79366	6093	1,066,485	217,728	0.997	0.986	1.0	0.985
PF00169	PH Domain	105	55196	4115	2,410,065	486,528	0.985	0.958	0.990	0.990
PF00595	PDZ Domain	82	15299	1489	1,466,283	287,360	0.973	0.846	0.939	0.878
PF00677	Lumazine BD	87	16192	2853	1,651,608	323,712	0.994	0.982	0.908	0.896
PF00763	Catalytic Domain	116	10886	2892	2,943,906	590,976	0.983	0.941	0.956	0.956
PF13354	Beta-Lactamase	202	7515	1559	8,956,983	1,814,656	0.942	0.910	0.965	0.935

4 Discussion

This comprehensive analysis of a single-layer self-attention mechanism involved a thorough comparison with the standard pseudo-likelihood approximation found in PlmDCA. Our exploration spanned a diverse dataset, encompassing nine protein families meticulously curated from the Protein Data Bank.

The primary feature introduced in our self-attention model lies in its low-rank decomposition, strategically designed to offer a more streamlined representation of the model's parameters. The pivotal metric for assessing the efficacy of this low-rank representation is the compression ratio c_r, a ratio that gauges the number of parameters in AttentionDCA against those in traditional DCA models (cf. Eq. 8). By fixing the MSA length and alphabet size, we identified pairs of internal dimension (d) and number of heads (H) for which the algorithm consistently demonstrated robust performance in contact prediction, albeit a significant parameter reduction compared to traditional DCA. This approach provides a nuanced understanding of how varying the number of model parameters impacts predictive capabilities.

Unlike the conventional PlmDCA method, our self-attention model adeptly disentangles spatial information exclusively through the sequence composition in terms of amino acids, as elucidated in Eq. 16. This unique encoding of spatial information in attention matrices $\{A^h_{ij}\}$ highlights an efficient representation wherein only a limited number of elements, as depicted in Fig. 4, suffice to capture structurally relevant information. The specialization of different attention heads in capturing distinct sets of contacts underscores the model's adaptability to varied structural patterns without resorting to explicit sparsity priors like L_1 or L_0 regularization. One noteworthy facet of our self-attention representation is its ability to exploit the factorization between sequence and amino acid spaces, as detailed in Eq. 4. This enables the concurrent learning of multiple families, as exemplified in Eq. 9, where the log-likelihood elegantly factorizes across different parameter sets, interconnected by a universal multi-head space of amino acid interactions $\{V^h\}$.

Beyond its primary application in contact prediction, the self-attention layer seamlessly transitions into a fully generative architecture by implementing an auto-regressive decomposition. Although the initially observed epistatic score contact prediction marginally lagged behind that of ArDCA, our subsequent experimentation involving sequence generation revealed comparable results, as demonstrated in Fig. 8. Ultimately, the observed heterogeneities in the principal component analysis of the generated dataset mirrored the clustering captured by ArDCA, affirming the versatility of our self-attention model across various tasks, as shown in Figure 6.

References

1. Dill KA and MacCallum JL. The protein-folding problem, 50 years on. Science (New York, N.Y.) 2012;338:1042–6.

2. Jumper J, Evans R, Pritzel A, et al. Highly accurate protein structure prediction with AlphaFold. Nature 2021;596. Number: 7873 Publisher: Nature Publishing Group:583–9.

3. Juan D de, Pazos F, and Valencia A. Emerging methods in protein co-evolution. Nature Reviews Genetics 2013;14. Number: 4 Publisher: Nature Publishing Group:249–61.

4. Cocco S, Feinauer C, Figliuzzi M, Monasson R, and Weigt M. Inverse statistical physics of protein sequences: a key issues review. Reports on Progress in Physics 2018;81. Publisher: IOP Publishing:032601.

5. Durbin R, Eddy SR, Krogh A, and Mitchison G. Biological sequence analysis: Probabilistic models of proteins and nucleic acids.

6. Weigt M, White RA, Szurmant H, Hoch JA, and Hwa T. Identification of direct residue contacts in protein-protein interaction by message passing. Proceedings of the National Academy of Sciences of the United States of America 2009;106:67–72.

7. Morcos F, Pagnani A, Lunt B, et al. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. Proceedings of the National Academy of Sciences 2011;108. Publisher: Proceedings of the National Academy of Sciences:E1293–E1301.

8. Ekeberg M, Lövkvist C, Lan Y, Weigt M, and Aurell E. Improved contact prediction in proteins: Using pseudolikelihoods to infer Potts models. Physical Review E 2013;87. Publisher: American Physical Society:012707.

9. Ackley DH, Hinton GE, and Sejnowski TJ. A learning algorithm for boltzmann machines. Cognitive Science 1985;9:147–69.

10. Figliuzzi M, Barrat-Charlaix P, and Weigt M. How Pairwise Coevolutionary Models Capture the Collective Residue Variability in Proteins? Molecular Biology and Evolution 2018;35:1018–27.

11. Trinquier J, Uguzzoni G, Pagnani A, Zamponi F, and Weigt M. Efficient generative modeling of protein sequences using simple autoregressive models. Nature Communications 2021;12. Number: 1 Publisher: Nature Publishing Group:5800.

12. Vaswani A, Shazeer N, Parmar N, et al. Attention Is All You Need. 2017.

13. Nambiar A, Liu S, Hopkins M, Heflin M, Maslov S, and Ritz A. Transforming the Language of Life: Transformer Neural Networks for Protein Prediction Tasks. Pages: 2020.06.15.153643 Section: New Results. 2020.

14. Wolf T, Debut L, Sanh V, et al. Transformers: State-of-the-Art Natural Language Processing. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online: Association for Computational Linguistics, 2020:38–45.

15. Bhattacharya N, Thomas N, Rao R, et al. Single Layers of Attention Suffice to Predict Protein Contacts. preprint. Bioinformatics, 2020.

16. Rende R, Gerace F, Laio A, and Goldt S. Optimal inference of a generalised Potts model by single-layer transformers with factored attention. 2023.

17. Rao R, Liu J, Verkuil R, et al. MSA Transformer. Pages: 2021.02.12.430858 Section: New Results. 2021.

18. Sgarbossa D, Lupo U, and Bitbol AF. Generative power of a protein language model trained on multiple sequence alignments. eLife 2023;12. Ed. by Colwell LJ and Walczak AM. Publisher: eLife Sciences Publications, Ltd:e79854.

19. Kingma DP and Ba J. Adam: A Method for Stochastic Optimization. 2017.

20. Dunn S, Wahl L, and Gloor G. Mutual information without the influence of phylogeny or entropy dramatically improves residue contact prediction. Bioinformatics 2008;24:333–40.

21. Taylor WR and Sadowski MI. Structural Constraints on the Covariance Matrix Derived from Multiple Aligned Protein Sequences. PLOS ONE 2011;6. Publisher: Public Library of Science:e28265.

22. Shakhnovich EI and Gutin AM. Engineering of stable and fast-folding sequences of model proteins. Proceedings of the National Academy of Sciences 1993;90. Publisher: Proceedings of the National Academy of Sciences:7195–9.

23. Kuhlman B, Dantas G, Ireton GC, Varani G, Stoddard BL, and Baker D. Design of a novel globular protein fold with atomic-level accuracy. Science (New York, N.Y.) 2003;302:1364–8.

24. Otter DW, Medina JR, and Kalita JK. A Survey of the Usages of Deep Learning in Natural Language Processing. 2019.

25. Bahdanau D, Cho K, and Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translate. 2016.

26. Vig J, Madani A, Varshney LR, Xiong C, Socher R, and Rajani NF. BERTology Meets Biology: Interpreting Attention in Protein Language Models. 2021.

27. Niu Z, Zhong G, and Yu H. A review on the attention mechanism of deep learning. Neurocomputing 2021;452:48–62.

28. Ofer D, Brandes N, and Linial M. The language of proteins: NLP, machine learning & protein sequences. Computational and Structural Biotechnology Journal 2021;19:1750–8.

Acknowledgments

The authors acknowledge illuminating discussions with Sergey Ovchinnikov, Martin Weigt, and Francesco Zamponi.

Author Contributions

FC and AP contributed equally.

Funding

The Authors acknowledge financial support from the project "Explainable Models for Protein Design", funded by the MIUR Progetti di Ricerca di Rilevante Interesse Nazionale (PRIN) Bando 2022

- grant 2022TE5B7X. We also acknowledge "Centro Nazionale di Ricerca in High-Performance Computing, Big Data, and Quantum Computing" (ICSC), and "FAIR - Future Artificial Intelligence Research", received funding from the European Union Next-GenerationEU (Piano Nazionale di Ripresa e Resilienza—Missione 4 Componente 2, Investimento 1.3—D.D. 1555 11/10/2022, PE00000013). This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

Conflicts of Interest

None.

Supplementary Materials

Summary Statistics

A Multiple Sequence Alignment can be used to extract summary statistics representing the protein family. In particular, single- and pair-wise frequency counts of the amino acids in the alignment are defined as:

$$f_i(a) = \frac{1}{M}\sum_{m=1}^{M}\delta(a, a_i^m) \quad ,$$
$$f_{i,j}(a,b) = \frac{1}{M}\sum_{m=1}^{M}\delta(a, a_i^m)\delta(b, a_j^m) \quad . \tag{18}$$

The application of a maximum entropy principle to the frequency counts returns the Potts model at the base of every DCA method. Higher statistics would produce other terms in the Hamiltonian which could not be inferred due to the relatively small size of the datasets at hand.

A good generative model should be able to reproduce the summary statistics and higher statistics derived from it, as in the case of the connected 2-site correlations shown in Figure 8 B) and given by:

$$c_{ij}(a,b) = f_{i,j}(a,b) - f_i(a)f_j(b) \quad . \tag{19}$$

Sequence re-weighting and M_{eff}

Sequences in a natural MSA are not truly independently distributed. Indeed, a certain degree of homology arises due to the phylogenetic relationships between different sequences and it brings about probabilistic biases towards specific sets of sequences. To avoid the over-sampling of similar sequences and the relative under-representation of other sequences, it is necessary to introduce a reweighting scheme. For each sequence $\mathbf{a}^i$, a weight is defined as the inverse of the number of sequences similar to it:

$$w^i = \frac{1}{m^i} = \left|\left\{j \middle| 1 \leq j \leq M, \mathrm{seqid}(\mathbf{a}^i, \mathbf{a}^j) > xL\right\}\right|^{-1} \quad , \tag{20}$$

where x is a similarity threshold that we set equal to 0.9. These weights enter the computation of the pseudo-likelihood as in:

$$\mathcal{L}(\mathbf{J}, \mathcal{D}) = -\frac{1}{M_{eff}} \sum_{m=1}^{M} w^m \left\{ \sum_{j \neq i} J_{ij}(A_i^m, A_j^m) - \log \left[\sum_{a=1}^{q} \exp\left\{ \sum_{j \neq i} J_{ij}(a, a_j^m) \right\} \right] \right\} \quad . \tag{21}$$

where the effective depth M_{eff} is then defined as the sum of each weight:

$$M_{eff} = \sum_{i=1}^{L} \frac{1}{m^i} \quad . \tag{22}$$

Using this re-weighting scheme, the frequency counts can be written as:

$$\begin{aligned} f_i(a) &= \frac{1}{M_{eff}} \sum_{m=1}^{M} w^m \delta(a, a_i^m) \quad , \\ f_{i,j}(a, b) &= \frac{1}{M_{eff}} \sum_{m=1}^{M} w^m \delta(a, a_i^m) \delta(b, a_j^m) \quad . \end{aligned} \tag{23}$$

Depending on the effective depth of an MSA, the DCA method can be more or less efficient, depending on its complexity.

Average Product Correction

The standard procedure to extract contact information from the interaction tensor of a DCA method is to compute its Frobenious norm to map each $J_{ij}(a, b) \in \mathbb{R}^{q \times q}$ matrix into a scalar value, useful to define a contact score for positions (i, j) as in Equation 7. Empirically, it can be seen that some residues are more prone to establish interactions than others, resulting in a confounding factor for contact prediction. The average product correction mitigates this effect by subtracting from the Frobenious norm a null-model contribution for each position pair, so that:

$$F_{ij}^{APC} = F_{ij} - \frac{F_i F_j}{F} \quad , \tag{24}$$

with

$$\begin{aligned} F_i &= \frac{1}{L} \sum_{k \neq i} F_{ik} \quad , \\ F &= \frac{1}{L(L-1)} \sum_{k,l \neq i,j} F_{kl} \quad . \end{aligned}$$

Learning Parameters

Table 2 shows the parameters used for the inference of the standard and autoregressive version of AttentionDCA. If not specified otherwise, the same parameters have been used in both versions. The size of the mini-batches used has been fixed to $b_s = 1000$ for all families, along with the compression

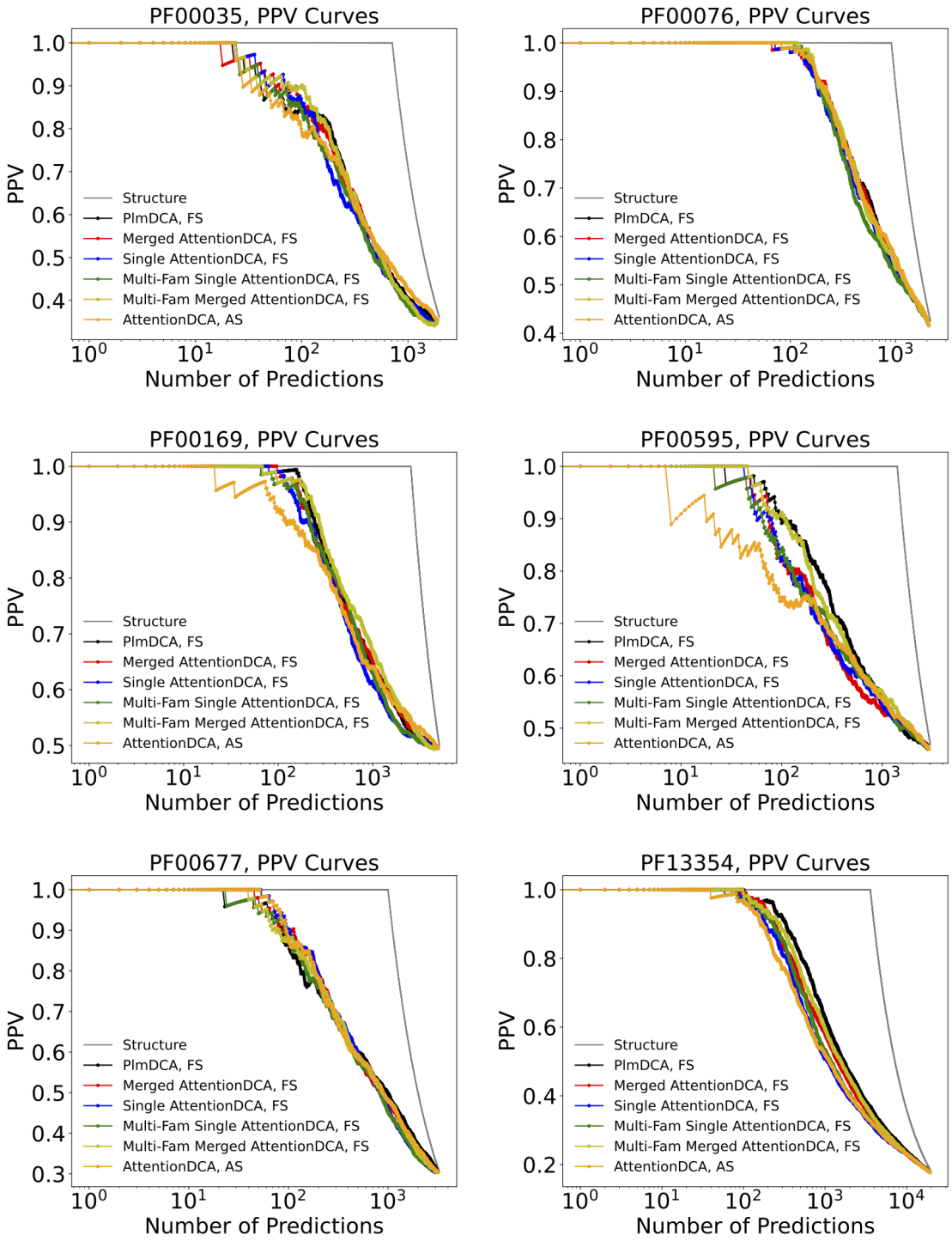
ratio $c_r = 0.20$ and the number of heads $H = 128$. For each family, the number of epochs and the regularisation have been selected in order to get an optimal contact prediction.

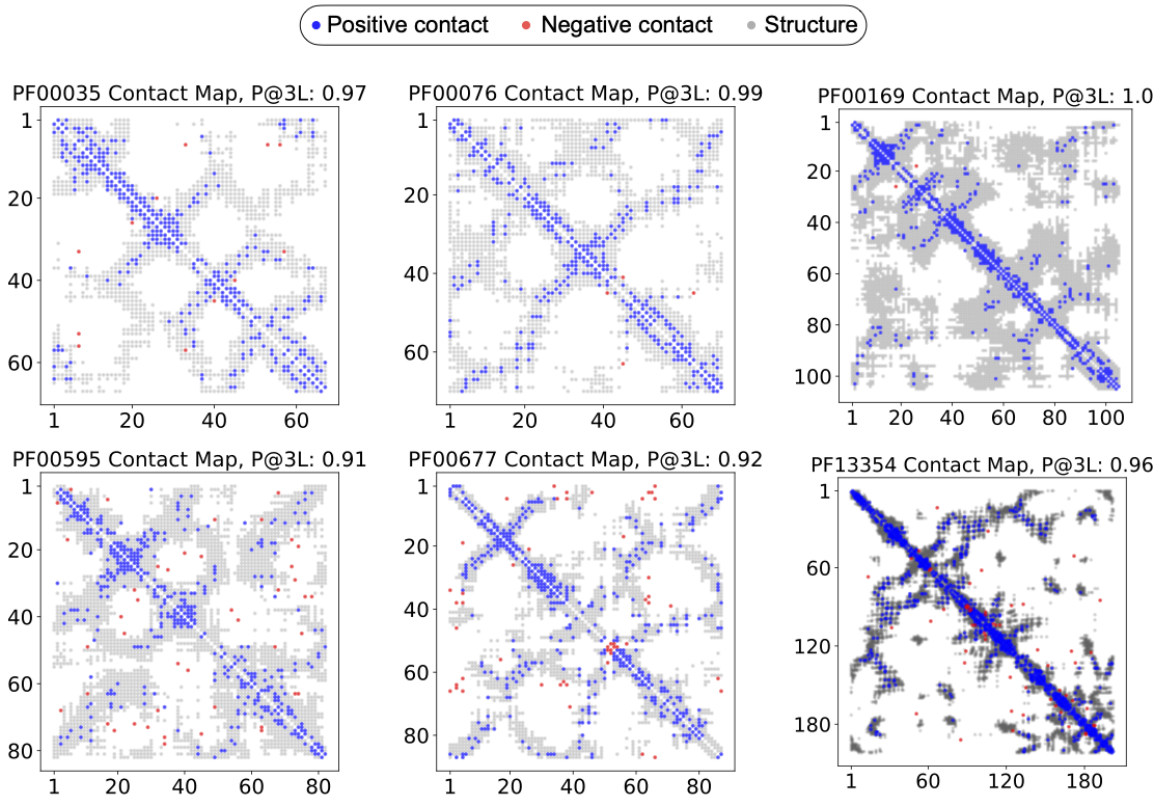
Table 2: Inference parameters and hyper-parameters of the model for each protein family.

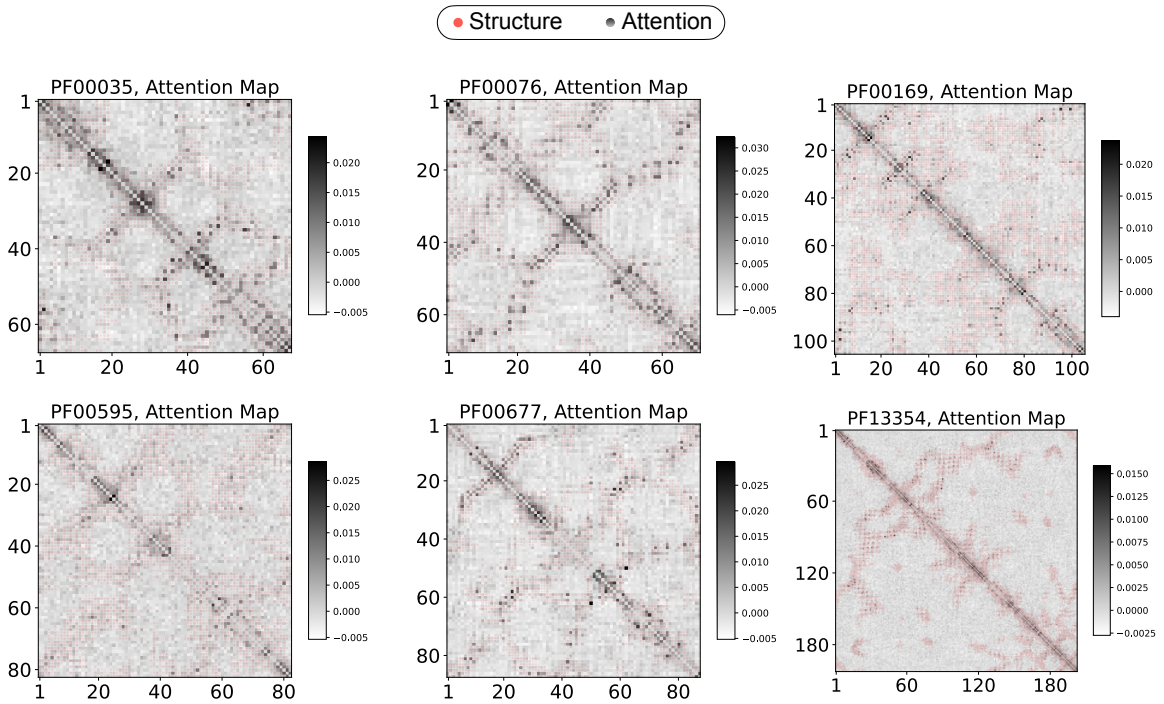
	H	d	λ Standard	λ Autoregressive	# epochs
PF00014	128	5	0.001	1.0×10^{-5}	100
PF00035	128	8	0.001	1.0×10^{-5}	100
PF00072	128	17	0.001	1.0×10^{-5}	20
PF00076	128	9	0.001	1.0×10^{-5}	100
PF00169	128	16	0.001	1.0×10^{-5}	100
PF00595	128	11	0.001	0.01	150
PF00677	128	12	0.001	1.0×10^{-5}	100
PF00763	128	18	0.001	1.0×10^{-5}	50
PF13354	128	34	0.001	0.01	200

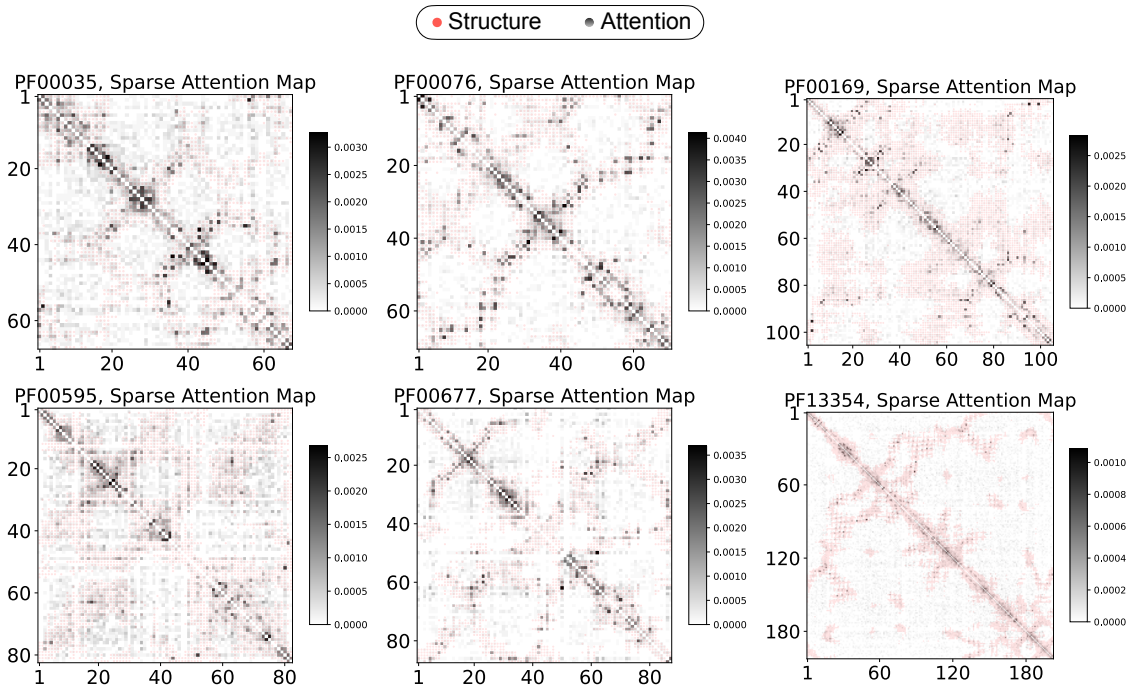
Figures

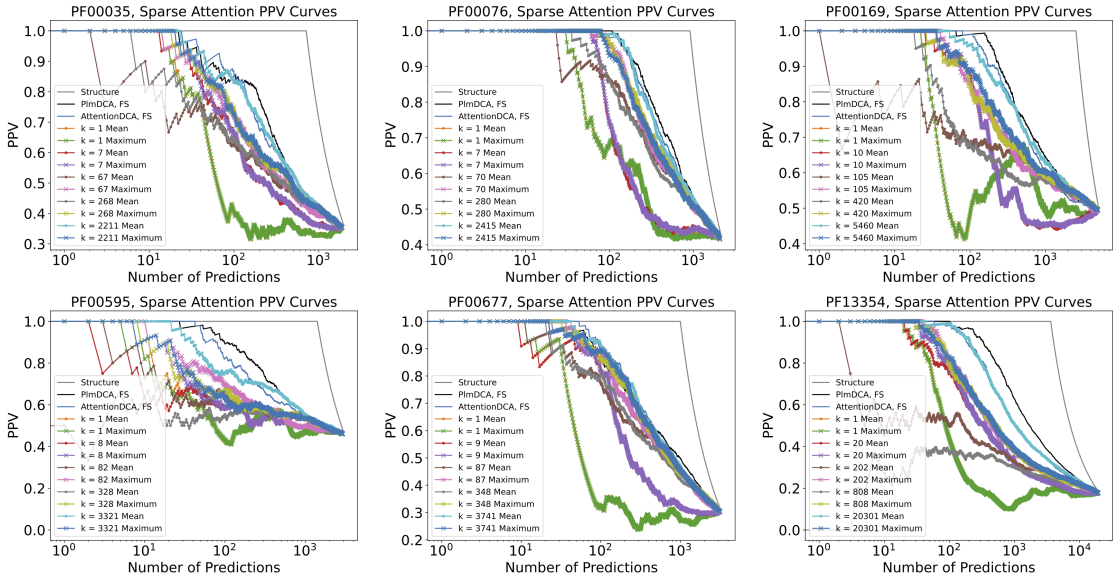
In the following, we include additional figures showing the results on the complete dataset of protein families available to us. Respectively we present PPV curves for the non-generative version (cf. Fig. 2 in the main text), contact map from the Frobenious score of the non-generative version (cf. Fig. 3 A)), attention maps (cf. Fig. 3 B)), sparse attention maps (cf. Fig. 3 C)), sparse attention PPV curves (cf. Fig. 5), PPV @L for sparse attention at different k (cf. Fig. 4), PPV curves for the generative version (cf. Fig. 8 A), 2-site connected correlations (cf. Fig. 8 B)), principal component analysis on ArDCA and PlmDCA (cf. Fig. 6).

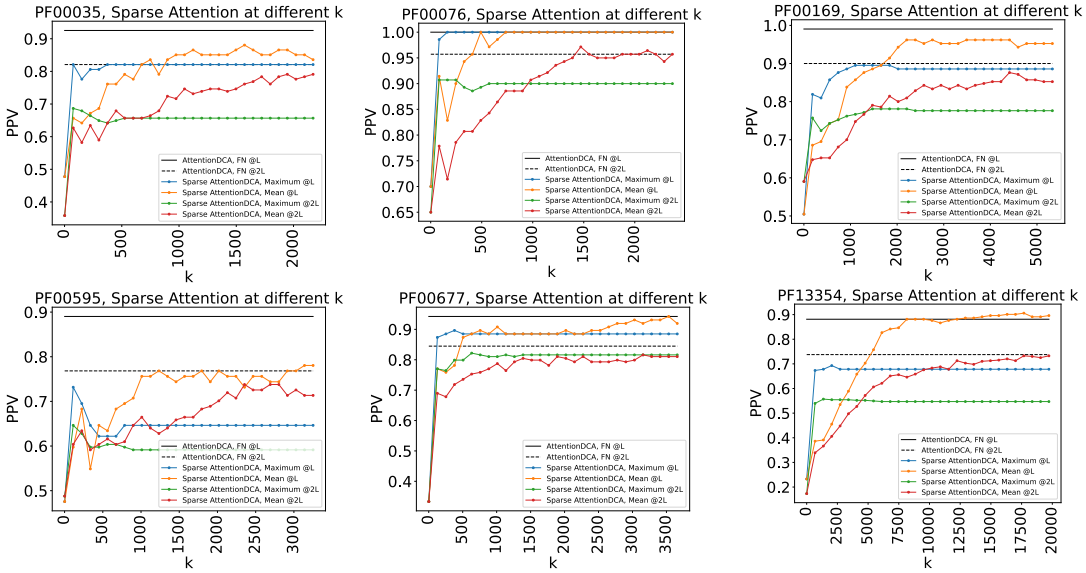

PF00035, PPV Curves
PF00076, PPV Curves
PF00169, PPV Curves
PF00595, PPV Curves
PF00677, PPV Curves
PF13354, PPV Curves
PPV
Number of Predictions
Structure
PlmDCA, FS
Merged AttentionDCA, FS
Single AttentionDCA, FS
Multi-Fam Single AttentionDCA, FS
Multi-Fam Merged AttentionDCA, FS
AttentionDCA, AS

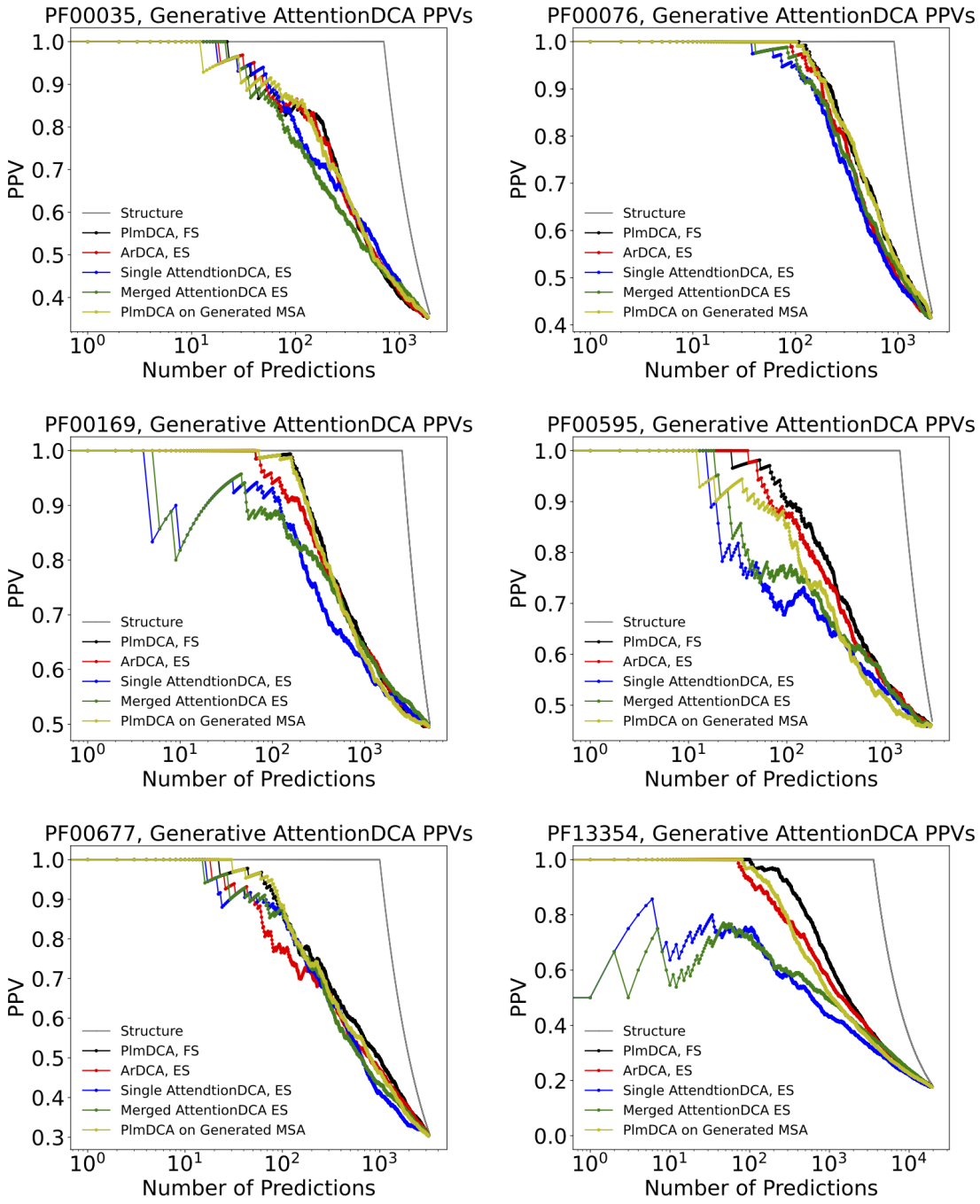

Positive contact
Negative contact
Structure
PF00035 Contact Map, P@3L: 0.97
PF00076 Contact Map, P@3L: 0.99
PF00169 Contact Map, P@3L: 1.0
PF00595 Contact Map, P@3L: 0.91
PF00677 Contact Map, P@3L: 0.92
PF13354 Contact Map, P@3L: 0.96

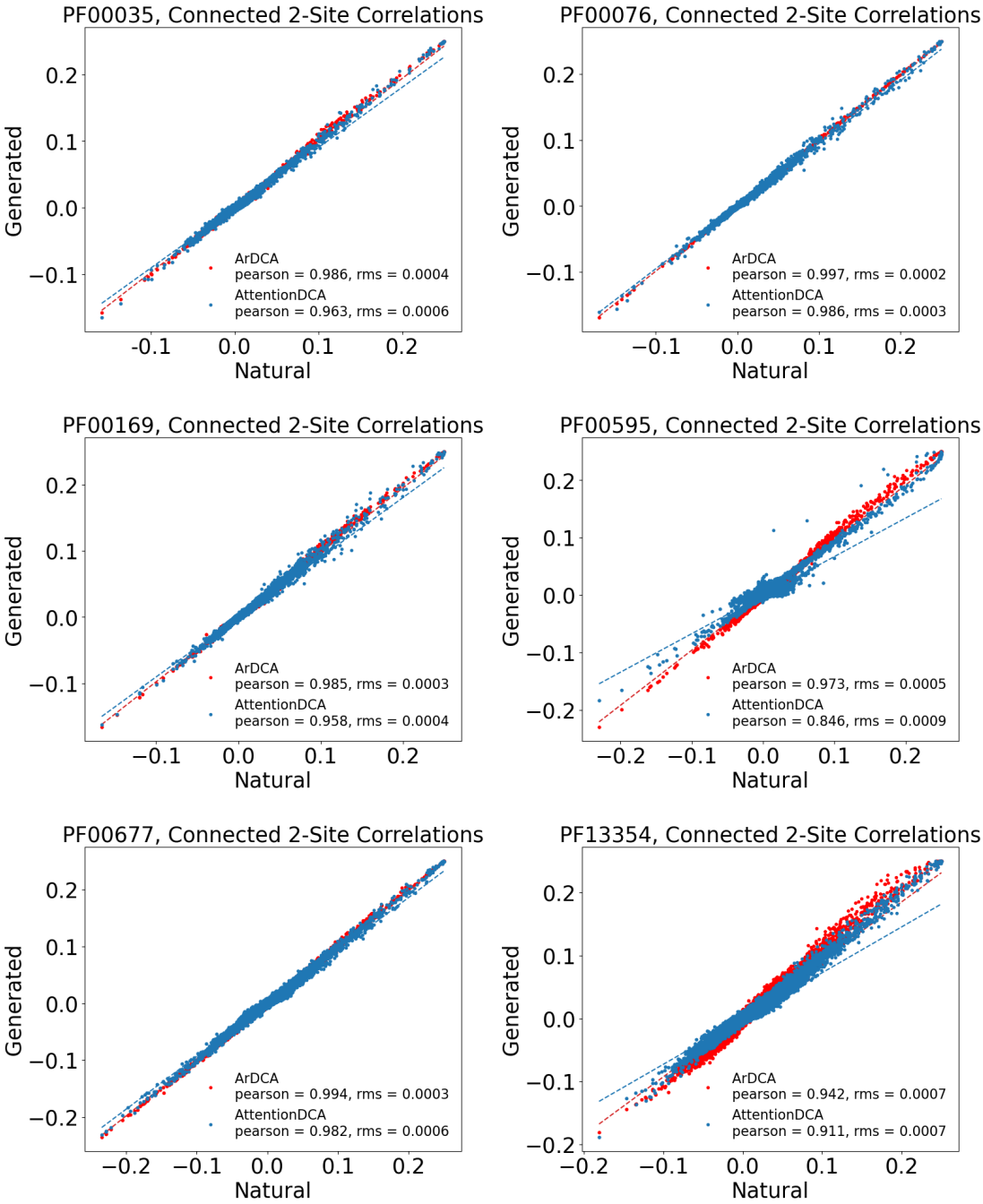

Structure
Attention
PF00035, Attention Map
PF00076, Attention Map
PF00169, Attention Map
PF00595, Attention Map
PF00677, Attention Map
PF13354, Attention Map

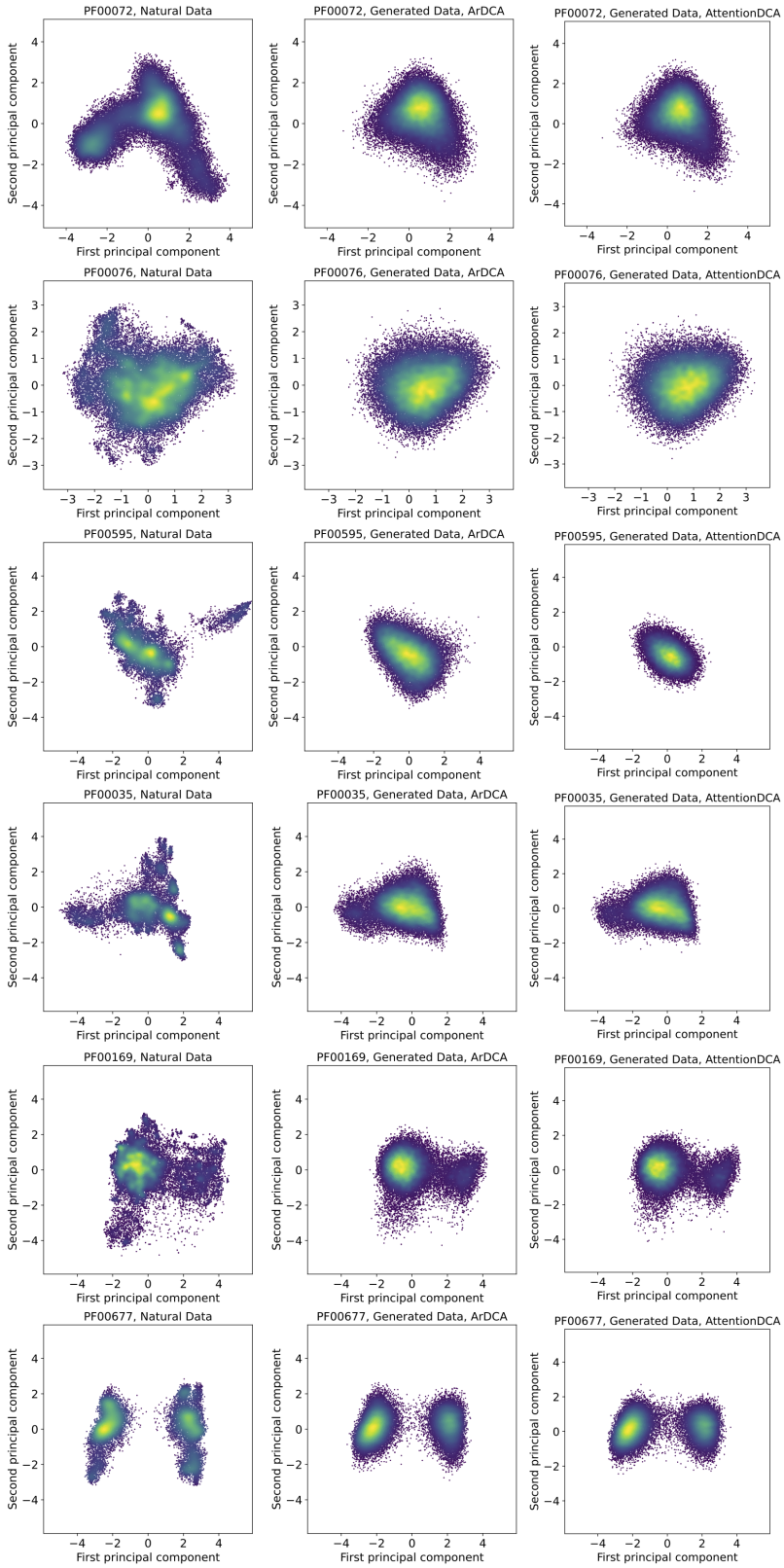

Structure
Attention
PF00035, Sparse Attention Map
PF00076, Sparse Attention Map
PF00169, Sparse Attention Map
PF00595, Sparse Attention Map
PF00677, Sparse Attention Map
PF13354, Sparse Attention Map


PF00035, Sparse Attention PPV Curves
PF00076, Sparse Attention PPV Curves
PF00169, Sparse Attention PPV Curves
PF00595, Sparse Attention PPV Curves
PF00677, Sparse Attention PPV Curves
PF13354, Sparse Attention PPV Curves
PPV
Number of Predictions
Structure
PlmDCA, FS
AttentionDCA, FS


PF00035, Sparse Attention at different k
PF00076, Sparse Attention at different k
PF00169, Sparse Attention at different k
PF00595, Sparse Attention at different k
PF00677, Sparse Attention at different k
PF13354, Sparse Attention at different k
PPV
k
AttentionDCA, FN @L
AttentionDCA, FN @2L
Sparse AttentionDCA, Maximum @L
Sparse AttentionDCA, Mean @L
Sparse AttentionDCA, Maximum @2L
Sparse AttentionDCA, Mean @2L


PF00035, Generative AttentionDCA PPVs
PF00076, Generative AttentionDCA PPVs
PF00169, Generative AttentionDCA PPVs
PF00595, Generative AttentionDCA PPVs
PF00677, Generative AttentionDCA PPVs
PF13354, Generative AttentionDCA PPVs
PPV
Number of Predictions
Structure
PlmDCA, FS
ArDCA, ES
Single AttendtionDCA, ES
Merged AttentionDCA ES
PlmDCA on Generated MSA


PF00035, Connected 2-Site Correlations
ArDCA
pearson = 0.986, rms = 0.0004
AttentionDCA
pearson = 0.963, rms = 0.0006
PF00076, Connected 2-Site Correlations
ArDCA
pearson = 0.997, rms = 0.0002
AttentionDCA
pearson = 0.986, rms = 0.0003
PF00169, Connected 2-Site Correlations
ArDCA
pearson = 0.985, rms = 0.0003
AttentionDCA
pearson = 0.958, rms = 0.0004
PF00595, Connected 2-Site Correlations
ArDCA
pearson = 0.973, rms = 0.0005
AttentionDCA
pearson = 0.846, rms = 0.0009
PF00677, Connected 2-Site Correlations
ArDCA
pearson = 0.994, rms = 0.0003
AttentionDCA
pearson = 0.982, rms = 0.0006
PF13354, Connected 2-Site Correlations
ArDCA
pearson = 0.942, rms = 0.0007
AttentionDCA
pearson = 0.911, rms = 0.0007
Generated
Natural


PF00072, Natural Data
PF00072, Generated Data, ArDCA
PF00072, Generated Data, AttentionDCA
PF00076, Natural Data
PF00076, Generated Data, ArDCA
PF00076, Generated Data, AttentionDCA
PF00595, Natural Data
PF00595, Generated Data, ArDCA
PF00595, Generated Data, AttentionDCA
PF00035, Natural Data
PF00035, Generated Data, ArDCA
PF00035, Generated Data, AttentionDCA
PF00169, Natural Data
PF00169, Generated Data, ArDCA
PF00169, Generated Data, AttentionDCA
PF00677, Natural Data
PF00677, Generated Data, ArDCA
PF00677, Generated Data, AttentionDCA
First principal component
Second principal component