PNAS PERSPECTIVE OPEN ACCESS

Machine learning in biological physics: From biomolecular prediction to design

Jonathan Martin[a,1], Marcos Lequerica Mateos[b,1], José N. Onuchic[c,d,e,f,2], Ivan Coluzza[b,g,2], and Faruck Morcos[a,h,2]

Edited by Yuhai Tu, International Business Machines Corp., Yorktown Heights, NY; received September 4, 2023; accepted December 8, 2023, by Editorial Board Member Herbert Levine

Downloaded from https://www.pnas.org by "UNIVERSITY OF TEXAS AT DALLAS, LIB SERIALS ONLINE CAPITAL" on November 12, 2024 from IP address 129.110.241.53.

Machine learning has been proposed as an alternative to theoretical modeling when dealing with complex problems in biological physics. However, in this perspective, we argue that a more successful approach is a proper combination of these two methodologies. We discuss how ideas coming from physical modeling neuronal processing led to early formulations of computational neural networks, e.g., Hopfield networks. We then show how modern learning approaches like Potts models, Boltzmann machines, and the transformer architecture are related to each other, specifically, through a shared energy representation. We summarize recent efforts to establish these connections and provide examples on how each of these formulations integrating physical modeling and machine learning have been successful in tackling recent problems in biomolecular structure, dynamics, function, evolution, and design. Instances include protein structure prediction; improvement in computational complexity and accuracy of molecular dynamics simulations; better inference of the effects of mutations in proteins leading to improved evolutionary modeling and finally how machine learning is revolutionizing protein engineering and design. Going beyond naturally existing protein sequences, a connection to protein design is discussed where synthetic sequences are able to fold to naturally occurring motifs driven by a model rooted in physical principles. We show that this model is "learnable" and propose its future use in the generation of unique sequences that can fold into a target structure.

Potts model | protein design | protein structure and dynamics | protein evolution | transformer model

In recent years, a large number of fields of science have been impacted by theoretical and technical breakthroughs in the field of machine learning. These developments and prediction ability are fueled by the emergence of hardware that is optimized for learning architectures and the availability and storage of large amounts of high-quality data resulting from experimental efforts. Physics, and more specifically, biological physics is not an exception. On the contrary, biological physics is one of the sub-fields of science that has benefited the most by the convergence of large amounts of biological data and the development of modeling and learning approaches to unravel the mechanisms of biological phenomena. Clear examples include advancing our understanding of the sequence–structure–function relationships in biomolecules, the dynamics of protein folding, and biomedical applications. In this perspective, we aim to provide a glimpse at the state-of-the-art algorithms in machine learning and how they are utilized for several applications in biological physics. We provide a non-exhaustive, but focused, account on how important modern learning algorithms, like the transformer architecture, are inherently connected to early developments in biological physics such as the Hopfield Network. We show how a mathematical representation of several learning algorithms in terms of "energy" functions unifies these formulations and has been used for different applications and problems concerning biological phenomena. We focus on the study of biomolecules, their structures, functions, and dynamics. We also look into the problem of protein design and how a combination of physical models and machine learning can be used to engineer possible proteins that fold to specified structures. We demonstrate that the energy Hamiltonian used to design proteins is "learnable" in a similar way that evolutionary data can be used to infer relevant amino acid interactions in protein families. This observation opens the door to improve protein design with the synergy of de novo physical approaches and sequence features encoded throughout evolutionary time scales.

Connecting Hopfield Networks to Transformers

The field of machine learning has seen tremendous gains in modeling performance through implementations of the transformer neural network architecture (1), specifically using deep neural networks with repeated blocks of this

Author affiliations: [a]Department of Biological Sciences, University of Texas at Dallas, Richardson, TX 75080; [b]BCMaterials, Basque Center for Materials, Applications and Nanostructures, Universidad del País Vasco/Euskal Herriko Unibertsitatea Science Park, Leioa 48940, Spain; [c]Center for Theoretical Biological Physics, Rice University, Houston, TX 77005; [d]Department of Physics and Astronomy, Rice University, Houston, TX 77005; [e]Department of Chemistry, Rice University, Houston, TX 77005; [f]Department of BioSciences, Rice University, Houston, TX 77005; [g]Basque Foundation for Science, Ikerbasque, Bilbao 48940, Spain; and [h]Department of Bioengineering, Center for Systems Biology, University of Texas at Dallas, Richardson, TX 75080

Author contributions: J.N.O., I.C., and F.M. designed research; J.M. and M.L.M. performed research; J.M. and M.L.M. contributed new reagents/analytic tools; J.M., M.L.M., J.N.O., I.C., and F.M. analyzed data; and J.M., M.L.M., J.N.O., I.C., and F.M. wrote the paper.

The authors declare no competing interest.

This article is a PNAS Direct Submission. Y.T. is a guest editor invited by the Editorial Board.

Copyright © 2024 the Author(s). Published by PNAS. This open access article is distributed under Creative Commons Attribution-NonCommercial-NoDerivatives License 4.0 (CC BY-NC-ND).

[1]J.M. and M.L.M. contributed equally to this work.

[2]To whom correspondence may be addressed. Email: jonuchic@rice.edu, ivan.coluzza@bcmaterials.net, or faruckm@utdallas.edu.

This article contains supporting information online at https://www.pnas.org/lookup/suppl/doi:10.1073/pnas.2311807121/-/DCSupplemental.

Published June 24, 2024.

architecture. Large language models like ChatGPT (2) make much use of them, using on the order of billions of learned parameters. In structural biology Alphafold (3) similarly uses this architecture to optimize all existent models and tools for structural prediction of proteins. This transformer network has been shown recently to be deeply connected with one of the early successes of machine learning, the Hopfield network (4, 5). The Hopfield network built on the results of Shun'ichi Amari (6) and alongside other works including backpropagation (7, 8) instigated a resurgence of interest in machine learning in science and engineering. We will briefly sketch a connection between these two models here.

The design of the Hopfield network was modeled directly on models of biological neurons, where a set of N neurons which receive inputs from other ones, described mathematically as a vector ($\mathbf{x} \in \{-1, 1\}^{N\times 1}$), corresponding to a neuron being active or inactive. Each neuron is connected to all other neurons and receives signal based on the summation of the inputs at each neuron scaled by their respective neuron's strength. These strengths can be represented as a weight matrix $\mathbf{W} \in \mathbb{R}^{N\times N}$, and the processing of an input can be written simply as

$$\mathbf{x}^* = \mathbf{W}\mathbf{x}, \qquad [1]$$

with the output of each neuron being the summation:

$$x_i^* = \sum_j x_i W_{ij} x_j. \qquad [2]$$

This output is typically passed through an activation function, commonly a sign or a sigmoid function to simulate the all-or-none activation of a neuron. One way to populate the values of the weight matrix is to update the values through an outer product of a configuration of neurons which is called a memory ξ:

$$\mathbf{W}^* = \xi\xi^T, \qquad [3]$$

and in this way, it can learn a particular configuration of neuron activations by storing the pairwise interactions of the states of the input. Once a memory has been stored, it is possible to recover this memory. It can be seen in Eq. **1** that the output $\mathbf{x}^*$ is again a valid input to $\mathbf{W}$ after passing through an activation function and, upon repeated processing through these neurons, it can recover our stored memory $\mathbf{x}^* = \xi$, given that the starting input is sufficiently similar to ξ.

This method was noted by both Amari and Hopfield to be connected to the Ising model (9, 10) (Fig. 1), where the process of storing a memory is mathematically similar to lowering the energy of a configuration of discretized atomic dipoles arranged in a lattice, where the strengths of interacting dipoles are described through a coupling matrix $\mathbf{W}$:

$$E = -\frac{1}{2}\sum_{i,j\neq i} W_{ij} x_i x_j, \qquad [4]$$

These methods later found their way to modeling biology in a different way, through a generalization of the Ising model called a Potts model (11). The Potts model generalizes the Ising model by allowing multiple discrete spins at each atom to be modeled. Successful implementations of Hamiltonian-based methods include the AWSEM method of structure prediction, which uses protein structure information as learned memories (12, 13). Later this form was applied to the study of protein sequences (14, 15). In this setting, the Hopfield neurons/modeled atoms can now be envisioned as the positions in a protein sequence of length N, where each position has 20 possible states (amino acids, vectorized as $\mathbf{x} \in \{0, 1\}^{N\times 20}$, with a single 1 per position), which can interact with all other positions in the sequence. The parameters for this model include $\mathbf{W}$, a coupling matrix where $\mathbf{W_{(ij)}}$ is a 20×20 block, and $\mathbf{h}$ which contains local field parameters that model single-site frequencies of amino acids at each position. When they are derived from empirical sequence data and combined with a maximum entropy modeling principle, this leads to the form

$$P(x_1, \ldots x_N) = \frac{1}{Z}\exp\left\{\sum_{i\leq j} \mathbf{x}_i \mathbf{W_{(ij)}} \mathbf{x}_j^T + \sum_i \mathbf{h}_{(i)} \mathbf{x}_i^T\right\}. \qquad [5]$$

This Boltzmann distribution form has had great success in predicting the critical structural residues of a protein, the selection temperature for folding (16, 17), and residues important for protein–protein interactions (18, 19).

Recently, another form of Eq. **4** was described (20):

$$E = -\sum_{m=1}^{M} F(\mathbf{x}^T \xi_m) \qquad [6]$$

We recast the matrix $\mathbf{W}$ as a set of M memories $\boldsymbol{\xi} \in \{-1, 1\}^{N\times M}$ and apply some smooth function $F(s)$ to the dot product of each pattern and our input state $\mathbf{x}$. When $F(s) = s^2$ the energy is equivalent to Eq. **4** and scaling the polynomial up has a number of interesting effects. One is allowing the energy to distinguish between XOR relationships in stored data for odd polynomials, another is that the representations of the stored memories become more interpretable at high polynomials, and most notably the memory capacity scales nonlinearly with the polynomial for a given memory size N (20).

What happens when the polynomial is scaled up to infinity? First analyzed in ref. 21, setting $F(s) = \exp(s)$ increases the capacity even further and was later shown to allow storing of continuous valued patterns (5). Their analysis led to a form using the softmax equation which exponentially averages all of the dot product comparisons (and is equivalent in form to a Boltzmann distribution). The equivalent for Eq. **1** is now an exponentially weighted average of stored patterns:

$$\mathbf{x}^* = \sum_{m=1}^{M} \xi_m \frac{\exp(\beta \mathbf{x}^T \xi_m)}{\sum_M \exp(\beta \mathbf{x}^T \xi_m)}. \qquad [7]$$

The authors of ref. 5 go on to show that this form of energy and pattern update are equivalent to the attention mechanism of the aforementioned transformer model.

To generalize Eq. **7** and achieve the self-attention portion of the transformer architecture, we must first convert our binary spin states into vector spins. We map the spin input vector through a set of learned linear encoding matrices to create a set of vectors termed queries, keys, and values.

Downloaded from https://www.pnas.org by "UNIVERSITY OF TEXAS AT DALLAS, LIB SERIALS ONLINE CAPITAL" on November 12, 2024 from IP address 129.110.241.53.

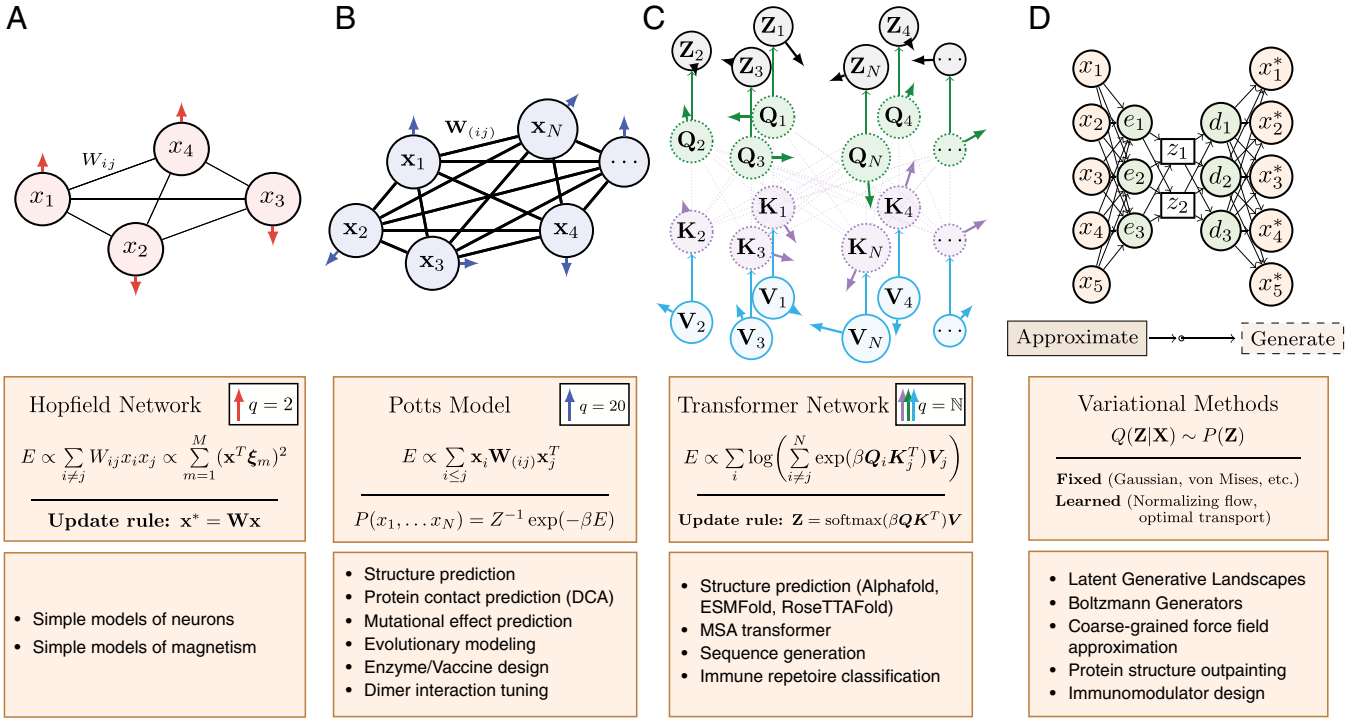

Fig. 1. Machine learning architectures and applications in biological physics. (*A*) Hopfield Networks, (*B*) Potts models, and (*C*) transformer networks. Each diagram represents a configuration of the spin systems, where the number of possible vector spins are listed in the white boxes. Learning is generally done through minimizing the energy of an encoded dataset, either through updating spin coupling parameters **W** (in *A* and *B*) or aligning encoded spins (*C*). Dashed lines in c represent the dot product comparison between the two spins, as opposed to explicit coupling parameters in *A* and *B*. (*D*) Schematic of a variational approximation scheme, with inputs as x and sampled outputs as x^*. These methods typically use a probabilistic latent space which has reduced dimensionality compared to the input data (a data bottleneck) which can enforce learning of only critical features, can permit more readily interpretable latent spaces (yields a "map"), and allows a computationally more efficient sampling than sampling from higher dimensional distributions. There is flexibility in how to model approximation or generation (including *A*–*C*), and while they can offer greater speed/interpretability over methods *A*–*C*, the introduced bottleneck can limit their overall performance. Note that approximation and generation could be handled by a single reversible network.

We can see the vector **x** as a "tokenized" matrix where each unique state is given a unique vector of dimension t ($\mathbf{X} \in \mathbb{R}^{N \times t}$), and the encoding matrices each with dimension $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{d \times t}$. Then, with encodings $\boldsymbol{Q} = \mathbf{W}_Q\mathbf{X}$, $\boldsymbol{K} = \mathbf{W}_K\mathbf{X}$, and $\boldsymbol{V} = \mathbf{W}_V\mathbf{X}$ (leaving output matrices with dimension $\mathbb{R}^{N \times d}$), we arrive at the equation

$$\boldsymbol{Z} = \mathrm{softmax}_i(\beta \boldsymbol{Q}\boldsymbol{K}^T)\boldsymbol{V}, \quad [8]$$

where the interior of the softmax function is a square matrix with elements $\mathbf{Q}_i^T\mathbf{K}_j$, and the softmax function is applied to each row independently. This is seen by examining a component of the output matrix:

$$\boldsymbol{Z}_i = \sum_j \mathrm{softmax}(\beta \boldsymbol{Q}_i^T \boldsymbol{K}_j)\boldsymbol{V}_j. \quad [9]$$

In this form, the model resembles an n-vector model (22), which is a generalization of the Potts model, where, instead of limiting our measurement of interaction to a fixed set of spin positions, we allow total freedom of our spin vectors and measure the dot product of interacting positions to determine the energy. Unlike an n-vector model, the coupling matrix is contained within the spin vectors themselves, as opposed to a distinct coupling matrix which would require an exchange parameter for each possible vector pairing.

The connection to statistical physics models was furthered with the development of fully energy-based transformers (23). The new energy function for the softmax form is

$$E = -\frac{1}{\beta} \sum_i \log\left(\sum_{i \neq j}^{N} \exp(\beta \boldsymbol{Q}_i \boldsymbol{K}_j^T) + \sum_M \exp(\beta \boldsymbol{\xi}_m^T \boldsymbol{X}_i)\right). \quad [10]$$

Here, we can combine the attention mechanism in Eq. **8** with the exponential form of Eq. **6** and define a set of memories $\boldsymbol{\xi} \in \mathbb{R}^{N \times M}$ which are learned such that overall the energy function is only minimized when the unencoded spins are correctly modeled by the learned Hopfield memories. This closely resembles the local field parameter **h** in Eq. **5**, which ensures that the energy reflects the overall frequencies of the spin states at each site. The design of this energy emphasizes the original design of the Hopfield network: the storage of information in a minimal energy well which can be reached through a dynamic update process and a sufficiently close initial configuration. An energy function derived naively from Eq. **8** does not necessarily have this property (see also refs. 5 and 24 for alternative formulations).

Importantly, this described encoding of queries, keys, and values does not allow positional information to be passed (a + 1 at v_1 is indistinguishable from a + 1 at v_3 when encoded with our matrices, which would lead our inference to something akin to a statistical potential), so typically this information is added back in through a positional encoding which offsets the values of the vectors by their position in the

Downloaded from https://www.pnas.org by "UNIVERSITY OF TEXAS AT DALLAS, LIB SERIALS ONLINE CAPITAL" on November 12, 2024 from IP address 129.110.241.53.

array. One conceptually important version of this encoding is the relative positional encoding (25, 26), where instead of encoding absolute position, the vectors are offset only by their relative distance. Alphafold implements this and considers only protein subsequences in Eq. **8**, yet over many subsequence blocks and many different protein sequences a more "generic" relationship of amino acids may be learned for structure prediction purposes. It is still not clear in what way this is generic; some relationships are difficult to model accurately with transformers and positional encoding, such as position-dependent rule-based patterning (27). In their results, any positional encoding scheme gives out-of-distribution results when the encodings are pushed to sequence lengths not seen during training. Untangling how queries and keys encode coupling strength, position encoding, and token identity may help clarify what is being learned.

The values matrix in Eq. **8** was a design choice motivated by machine translation of languages, where the couplings between words in a sentence give information on how the same sentence in another language should be composed. This setup allows a lot of flexibility for engineering complex architectures where information from different regimes can be combined to make predictions. Understanding the transformer's weights and predictions is also not typically possible, while weight matrices in Potts models can be more readily interpreted. Perhaps more physics-based analysis can bring some intuitions to help better understand this emerging class of models (see ref. 28 for an example of these models).

Variational Methods and Generative AI

Another fast-growing field is the approximation of complex probability distributions through variational methods. In these schemes, it is assumed that the data being modeled ($\mathbf{x} \in \mathbf{X}$) are jointly distributed with some unknown variable(s) $P(\mathbf{X}, \mathbf{Z})$, and it is possible to approximate these latent variables by modeling some new distribution Q such as

$$Q(\mathbf{Z}|\mathbf{X}) \sim P(\mathbf{Z}). \quad \textbf{[11]}$$

Introduced conceptually for probabilistic graphical models in ref. 29, much of the modern interest was ignited by combining this concept with the framework of autoencoder networks (30) and the development of probability reparameterization techniques which yield computationally efficient algorithms (31). Kingma et al.'s method set $P(\mathbf{Z})$ to the standard Gaussian distribution, and the approximation fit was measured as the Kullback–Leibler divergence

$$\mathrm{KL}(Q(\mathbf{Z}|\mathbf{X})||P(\mathbf{Z})). \quad \textbf{[12]}$$

The choices for how to model $P(\mathbf{Z})$ are myriad and growing, but they could currently be classified as being unlearned [$P(\mathbf{Z})$ is assumed and inflexible, such as Gaussian in ref. 31, wrapped normal distributions (32, 33), Dirichlet (34)], normalizing flow based (35–37), or optimal transport based (38, 39). Normalizing flow (and recently an extension to optimal transport) integrates learnable parameters into the variational scheme which transform samples from an initial known distribution, generally in an invertible way, such that the change in probabilities due to the transformation can be calculated and constrained. Another related method is the Generative Adversarial Network, which in the field of images can produce qualitatively better results (40), though in practice they can be difficult to train. These have been applied to protein sequence and structure generation (41, 42), and generally promise greater flexibility as in Eq. **12**, as there are many classes of f-divergences and probability distances that can be substituted there (43).

An Overview of Modern Machine Learning Methods and Their Connection to Protein Folding, Structure, Dynamics, Function, and Evolution

Fig. 1 summarizes the learning architectures described in the previous section and presents an overview of different applications in biological physics that utilize machine learning as a tool to enhance discovery, inference, and computational complexity. In the following sections, we provide a brief recount of such applications.

Protein Structure Prediction. The problem of protein folding or that of inferring a three-dimensional molecular structure of a protein using a sequence of amino acids has been relevant in the past five decades. Throughout the years, this problem has been tackled with multiple approaches including both experimental and theoretical methods (17, 44, 45). Here, we will focus on recent approaches using machine learning or statistical inference in conjunction with physical-based approaches to determine the fold of globular or membrane-bound proteins. A key idea for these types of approaches is the fact that collections of related protein sequences encode for similar structures. Therefore, looking at the coevolutionary patterns of these sequences can provide crucial information on long-range residue–residue interactions that drive the folding of a protein. Estimating these long-range contacts through the learning of the parameters of the joint probability of sequences in a protein family was a key initial step toward improving structure prediction. Methods like Direct Coupling Analysis (DCA) (15, 46), PSICOV (47), and GREMLIN (48), among several others, assumed the distribution of sequences to be Boltzmann distributed with a Potts Hamiltonian describing amino acid couplings and local fields (Fig. 1*B* and Eq. **5**). These couplings relate to amino acid pairing propensities and the local fields are connected to single site conservation (49). The ability to predict contact maps reliably from sequence alone was a stepping stone to combine these long-range amino acid pairings with physical interaction models to predict folded structures. One early example was the combination of structure-based models (50, 51) with DCA to predict structures for several folds and families (52, 53). Other related examples include the use of geometric constraints (54), combining these coevolutionary couplings with Rosetta (55) or with structure-based Potts methods like AWSEM (56), as well as combining contact map prediction with techniques like deep learning to improve contact prediction (57). These initial studies opened the door for other machine learning techniques to be incorporated into the problem, culminating with the

Downloaded from https://www.pnas.org by "UNIVERSITY OF TEXAS AT DALLAS, LIB SERIALS ONLINE CAPITAL" on November 12, 2024 from IP address 129.110.241.53.

introduction of the first version of Alphafold that used coevolutionary information with a combination of structural learning to substantially increase the accuracy of prediction (3, 58). The problem has recently reached a milestone in predictive accuracy with the introduction of AlphaFold2 (3) and RoseTTAFold (59, 60) which owe much of their improvement to transformer-based architectures (Fig. 1*D* and Eq. **9**). In particular, Alphafold2 saw large improvements through invariant point attention and iteratively recycling transformer inputs, which is reminiscent of the energy-based transformer's energy minimization procedure. Predictions have been made for hundreds of thousands of known sequences without experimentally determined structures and have been deposited in a database in collaboration with the European Molecular Biology Laboratory - European Bioinformatics Institute (EMBL-EBI) (61). Similar efforts integrating transformer-based language models have been made to predict millions of proteins in a framework called ESMfold (62) which reaches similar accuracy as AlphaFold with considerable improvements in computational complexity. Efforts to allow scientists to experiment and fine-tune the Alphafold architecture gave rise to OpenFold (63) which allows scientists to retrain and analyze an open-source version of the Alphafold pipeline as well as ColabFold (64) for easy web-based access for those without access to powerful hardware. Finally, accurate predictions of 3D protein structures that do not depend directly on multiple sequence alignments with methods like long short-term memory neural networks (LSTM) show promise (65, 66). This method has particularly fast inference and is a great example that lean and well-crafted tools can be built with modern machine learning methodology to tackle the folding problem.

Protein Dynamics Enhanced by Machine Learning. Coarse-grained modeling of a macromolecule aims to reduce the complexity of simulating molecular dynamics while maintaining an accurate model of atomic or quantum mechanical interactions described by all-atom force fields. Modern molecular dynamics approaches are looking into advances in machine learning to help bridge the gap between all-atomic models and coarse-grained ones. The choice of how to represent a force field thus is quite important; a recent advance uses a variational scheme (Fig. 1*D*) to find the parameters of coarse-grained force fields which reproduce the results of an all-atom simulation (67). They developed an extension of contrastive noise sampling, where noise is generated through an approximating distribution ($\mathbf{Q}(X)$) derived from all-atom input data in order to optimize the parameters of simpler coarse-grained force fields to fit said data. Using the input data to approximate the noise is one solution to the difficult problem of generating useful noise in the high-dimensional data space where all-atom simulations exist, and their method can also be used to generalize force fields from the simulations of a wide array of proteins to produce a generic and accurate set of force fields.

A significant problem in simulating molecular dynamics, either coarse-grained or all-atom, is the modeling of transitions between large conformational changes in proteins which happen on long timescales and calculating the free energy differences between them. One recent approach, the Boltzmann generator (68), combines variational approximation (Fig. 1*D*) with Boltzmann distributions (Fig. 1*B*), where short simulations of distant conformations are modeled as Boltzmann distributed samples produced through a normalizing flow from a shared Gaussian latent distribution (Eq. **11**), after which the two states can be connected in a consistent way and more meaningful free energy calculations between conformations can be performed. Normalizing flows are also being extended to flow-matching methods in coarse-grained simulations (69), where they are used to fit the densities of force fields in all-atom simulations to coarse-grained approximations and show good modeling performance even with small amounts of data. There are other methods being used to improve learning of simulation parameters or the sampling of molecular conformations. In ref. 70, swarm-based learning was combined with small-angle X-ray scattering to tackle the problem of weighting experimental data with a physical model in an MD simulation. In ref. 71, an encoder (Fig. 1*D*, "Approximate" step) takes all atom simulation coordinates and approximates the eigenfunctions of the simulation's Markovian state transitions, allowing for inclusion or exclusion of the slow or fast spectra. This embedded space is further processed before being decoded (Fig. 1*D*, "Generate") through a Wasserstein Generative Adversarial Network (GAN) (43) to produce samples from the low dimensional distribution of molecule configurations. They are able to generate all-atom configurations from this low dimensional distribution with a fraction of the computer time compared to a more complete description of the molecular transitions.

Functional Landscapes and Evolution. In addition to be able to predict structural features of proteins like residue–residue interaction contacts, sequence Potts formulations can use its energy representation (Fig. 1*B*) to assess the potential effects of mutations after learning parameters in a family of proteins. For instance, one can assume that a change in sequence that leads to an unfavorable energy can be a proxy of a disruptive mutation and conversely, a mutation that maintains or favors its energy would lead to beneficial mutations that improve functionality, enzymatic activity, or stability of the protein assessed. Recent examples include the use of DCA to predict the effect of mutations in cancer-related proteins (72) and discerning disruptive and enhancing mutations in signaling proteins (73). DCA formulations have also been used to unravel the mutational landscape of TEM-1 β-lactamase (74) to accurately predict the effects of mutations in its antibiotic resistance activity. Epistatic models using the Potts formulation like EVmutation (75) have been also utilized to infer, accurately, the mutational effects from high-throughput mutagenesis experiments as well as measurements of human disease–related mutations.

The potential to characterize fitness landscapes from the learning formulation in Fig. 1*B* can be exploited to investigate protein evolution using epistatic contributions. It has been shown that epistatic interactions obtained from the learned parameters in the Boltzmann distribution of sequence energies can be used to model the dynamics of sequence evolution in such a way that it recapitulates many statistical assumptions of sequence variation in past models of sequence evolution. One example is

Downloaded from https://www.pnas.org by "UNIVERSITY OF TEXAS AT DALLAS, LIB SERIALS ONLINE CAPITAL" on November 12, 2024 from IP address 129.110.241.53.

Sequence Evolution with Epistatic Contributions (SEEC) (76) and related models that have been useful to explain experimental outcomes (77–79). Therefore, being able to model the evolutionary constrains that a protein family has been exposed through evolutionary time-scales can then have significant applications. For example, the ability to predict mutable sites in SARS-CoV-2 proteins and epitopes (80); inferring polymorphisms in Escherichia coli in a context-aware manner (81) or identifying potential weak spots in viral proteins like HIV to guide rational vaccine design (82–84).

Along with the formulation of Boltzmann machines as discussed in the first section (Fig. 1*B*), another relevant method is restricted Boltzmann machines (RBM). In the RBM formulation for sequence data, the coupling parameters are replaced by a set of hidden units which, conditionally independently of each other, model the statistical features of the distribution being modeled. One particular formulation by Tubiana et al. (85) has been tailored to model independently covarying motifs within protein sequence data. It has been applied to predicting human leukocyte antigen 1–binding motifs in major histocompatibility complexes (86), and this emphasis on relatively small, functionally connected motifs could have interesting applications in directed evolution/high-throughput mutagenesis as well. Using data from an individual's immune repertoire, the updated Hopfield network and convolutional neural networks (CNNs) were combined to predict if a particular pathogen will be detected by the immune system (87). CNNs allow variable length sequences to be considered, and this work showed that these classifiers can be assessed to learn which sequence features predict a disease state.

Functional/evolutionary clustering has also been performed with variational autoencoders (VAE's), where scoring protein sequences generated from a low dimensional landscape with a Boltzmann energy function (Fig. 1*B*) elucidated an underlying fitness landscape learned by standard Gaussian VAEs (88). Novel generated sequences can be assessed by their statistical energy to better assess in vitro viability, and visualizing the underlying landscape can provide clearer delineation between functional clusters and can potentially pinpoint modifications which yield novel functions.

Transformers (Fig. 1*C*) trained on sequence data offer an interesting development on the Potts model–based methods in that, due to their positional encoding method, they can be trained on aligned sequence sets of different sizes which allows training a single model on vastly more data than previously allowed (89). Once trained in an unsupervised way [conceptually similar to Potts model training (90)], more models can be fit to the output for purposes like contact prediction or structure prediction as in ref. 62. One recent assessment (91) demonstrates an interesting property, potentially derived from the specific method for computing attention in Eq. **8**, where the transformer learns phylogenetic information (Hamming distance) and structural contact prediction in a more disentangled way than a Potts model. Understanding how this works could be beneficial, as separating residue correlations arising due to evolutionary history (autocorrelation) from correlations due to functional constraints is a long-standing problem.

Large language models have also been applied for function prediction purposes (92), where protein sequences are used to predict a functional label. ProtNLM is currently used in Uniprot (93) as a way to provide names where the established methods fail to provide any label, and generally language models are a promising strategy due to how well they can learn when given very large datasets.

Protein Design and Generation.

Sequence focused generative methods. Methods that take into account the evolutionary history of protein families are a clear choice for the goal of designing or engineering proteins that do not exist in nature. This is due to the enormous sequence space accessible to amino acid chains. The extant number of amino acid sequences, although large, is still small compared to the combinatorial possibilities that could preserve molecular interactions relevant for folding and function. Therefore, methods that can infer and explore such sequence space surrounding proper fitness or sequence energy wells are amenable for sequence design. A significant and relevant representative of this approach is the work done by Russ et al. on chorismate mutase enzymes. They utilized a generative model based on a Potts model called Boltzmann machine DCA to model the family of chorismate mutases using alignments from this protein family. Then, sampling from the distribution estimated using this approach, they were able to create a large collection of sequences that folded and had wild-type-like enzymatic activity (94). Other examples include the creation of functional chimeras in protein repressors whose domains were initially incompatible (95) and the use of SEEC to generate functional variants after being evolved in silico (96).

Outside of models of evolution, there are other sequence-based methods for designing functional amino acid chains which are being developed. Sequence-based transformer models, when trained in an unsupervised way, can generate sequences through iteratively masking residues in a template sequence and predicting what that residue would be, given the rest of the sequence, similar to Gibbs sampling used in their Potts model counterparts (97), though the sequences produced show unique statistical properties when compared to Potts based methods (statistical co-occurrence of sets of amino acids are sometimes better replicated than pairs of amino acids). Regardless, there is good indication that these can generate viable sequences. Variational autoencoders have already been shown to produce functional sequences (98, 99) and have an added benefit that the clustering produced by the method allows selective generation of sequence variants with desired functional properties. Cluster-guided sequence generation can easily be augmented with methods such as Potts models (88) to provide secondary sources of information to aid successful generation. Recent methods for designing immunomodulators (100) combine high-throughput in vivo assessment of VAE generated peptide sequences, where the results of these assessments are used to train a radial basis function classifier which shapes which regions of the VAE latent space would be better targeted for the generation of peptides to achieve specific design goals.

Downloaded from https://www.pnas.org by "UNIVERSITY OF TEXAS AT DALLAS, LIB SERIALS ONLINE CAPITAL" on November 12, 2024 from IP address 129.110.241.53.

Structure focused generative methods. Early successes in structure-focused de novo protein design combined known protein backbone structures and physics-based energy functions, such as globally optimal rotamer selection through dead-end elimination algorithms (101) or later Monte Carlo based optimization like RosettaDesign (102). Recent advances have improved the rate of successful designs through combining modern learning algorithms with both physical constraints on structure and evolution-informed constraints on primary sequence. One example is the trRosetta model (103), which uses a deep residual neural network to transform coevolutionary information from a multiple sequence alignment into maps of angles/distances for rigid body transformations between residues in a target structure, which is then combined with coarse-grained and full-atom energy-based methods for resolving a final predicted structure. They saw significant prediction accuracy even on sequences which folded and were created entirely de novo. Additionally, a transformer-based (Fig. 1*C*) method was developed to "paint" around a known partial structure (like a motif) (59); using a transformer-based model which, after training on structure and sequence information, can fill out (predict the previous/next amino acids) a user-input template in order to build a full structure. There are many design options here, such as designing a protein which binds to another protein, or designing a scaffold to support an active site. In a follow-up work (104), it was noted that this design process has a low in silico (Alphafold2 prediction) success rate, which led in part to the inclusion of a diffusion-based statistical method (105) into the de novo design process, which when combined with the RoseTTAFold model from ref. 59 and ideas from work on "hallucinated" proteins (106) improved in silico hit rate and also produced structures which folded in vivo.

Bridging the Gap between Natural and Artificial Sequences

In the previous section, we discuss recent advances on protein design aided by current learning methodologies. In this section, we present insights into a protein design framework that is strongly rooted in physical models. At the same time, we explore the possibility of learning such models to improve the generality of our approach. In this context, it is essential to delve deeper into the interplay between structure and function of protein sequences, a topic intimately connected with our previous discussions on modern machine learning methods as well as their relation to protein folding, structure, dynamics, and function. DCA in particular, or Potts models in general (Fig. 1*B*), offers intriguing insights into co-evolutionary signals within proteins, drawing a parallel to Boltzmann distribution-based models. Here, we employed the Caterpillar protein model (107) that can effectively bridge the gap between artificial sequence creation and accurate protein structure representation. Inspired by the simplicity and adaptability of its namesake, the Caterpillar model incorporates a full-atomistic backbone and uses a spherically symmetric potential, generating a variety of artificial sequences that can fold into protein structures.

Features of the Caterpillar Model. We employ the Caterpillar model driven by the goal to generate artificial sequences that starkly contrast natural ones, with the sole similarity being their ability to fold into the native structure. The model focuses primarily on the full-atomistic backbone and intentionally disregards the side chains. These elements are represented through a spherically symmetric potential centered around the C_α atoms, culminating in a structure that fittingly resembles a caterpillar (see *SI Appendix* for more model details). Its unique representation allows the model to leverage the maximum entropy principle to optimize and validate its predictions against a dataset of over 120 test proteins. The model captures the complex interplay between hydrogen bonds and side-chain interactions through the use of Lennard–Jones and sigmoidal (C_α–C_α) spherical potentials, respectively. In the context of the Potts model, these interactions are captured by the spin couplings and external fields, as described by the energy in Eq. **5** and Fig. 1*B*. This aspect of the Caterpillar model also makes it an ideal candidate for testing the validity of the direct coupling mean-field hypothesis under optimal conditions.

Generating the Families of Artificial Sequences. In an effort to ascertain the capabilities of a learning framework, e.g., DCA, for reconstructing underlying interactions between residues, we analyzed correlations and energy distributions within the Caterpillar Hamiltonian model and the DCA Hamiltonian. To this end, we considered three protein families, namely PDZ, FKBP, and Response regulator receiver domains (with Pfam codes PF00595, PF00254, PF00072 respectively). In a previous study (108), one representative member of each family was chosen as target structure from the Protein Data Bank (PDB) specifically 1WI2 (PDZ), 2PPN (FKBP), and 1NXW (Response regulator) and resulted in good designability for the caterpillar model.

The design simulation process at its core involves assigning random amino acids to each residue in a protein. The simulation proceeds by performing point amino acid mutations and residue swapping (interchanging the positions of two amino acids in a sequence) based on the Metropolis Monte Carlo scheme. This method guides the progression of the simulation, allowing us to explore a wide array of possible protein sequences. As the simulation progresses, it generates a vast number of sequences that are characterized by two collective variables: caterpillar energies $E = E_{ij}\left(A_i, A_j, r_{ij}\right) + E_i^{\mathrm{Sol}}\left(A_i\right) + E_j^{\mathrm{Sol}}\left(A_j\right)$ and the number of permutations $N_P = N!/(n_A!n_B!\ldots)$, which indicates the diversity in the composition of the sequence. In our previous studies (107), we found that sequences $Seq \in [\max(N_P), \min(E)]$ that exhibited a high number of permutations and low total energy were the best at folding into the desired protein structure. In previous work, we have performed extensive tests of the refolding of the artificial sequences generated with the Caterpillar protein model. Typically, the artificial sequences refold with a resolution between 2 and 3 Å RMSD.

In this study, we generated a total of 100,000 sequences from each simulation of the target structures 1WI2, 2PPN, and 1NXW. These sequences should meet two key criteria: They had to have a similarity threshold of less than

Downloaded from https://www.pnas.org by "UNIVERSITY OF TEXAS AT DALLAS, LIB SERIALS ONLINE CAPITAL" on November 12, 2024 from IP address 129.110.241.53.

90% identity within each set, ensuring a good diversity of sequences, and their collective coordinates N_P, E could deviate by no more than 10% from our reference $Seq \in [\max(N_P), \min(E)]$. From each simulation, a set of 10,000 sequences was randomly selected. This subset represented a snapshot of the solution space, giving us a sense of the variety of sequences generated by the design. Some of these sequences were then subjected to folding tests to ensure the validity of our simulation and the feasibility of these sequences to fold into the target protein structures (*SI Appendix*, Figs. S1–S4).

DCA Captures Relevant Residue–Residue Interactions from the Caterpillar Model. We used the mean field implementation of DCA (mfDCA) (14, 15), on 10,000 sequences for each designed protein family. Direct information (DI) offers critical insights by identifying pairs of residues that are highly correlated, which can signify direct interactions or shared evolutionary pressures. When applied to the sequences generated by Caterpillar, which are selected specifically for their ability to fold into the target protein structure, DCA serves as a tool to identify pairs of highly coupled residues, indicative of strong interactions or shared evolutionary pressures. The accuracy of DCA's predictive power is depicted in Fig. 2, where the contact map of protein 1NXW is displayed. Contacts below a 12 Å distance, which represents the interaction cutoff for residue–residue interactions in the Caterpillar model, are highlighted as open black squares.

Fig. 2 further divides the contact map into two triangular sections. The upper triangle overlays Direct Information (DI) values onto the contact map, categorizing them for better visibility. Higher DI values, indicative of stronger correlations or interactions, are denoted in shades closer to yellow. In contrast, lower DI values, potentially representative of weaker or indirect interactions, are displayed in shades closer to white. Notably, high DI values align with these squares, revealing a lack of false positives beyond the 12 Å cutoff. In the lower triangle, Caterpillar residue–residue interaction energies, represented as $E_{ij}\left(A_i, A_j, r_{ij}\right)$, are overlaid on top of the contact map. These energies are color-coded, with lower energies (indicative of stronger, more favorable interactions) in yellow and higher energies (representing weaker or less favorable interactions) in white. We gauged the effectiveness of the DI values as predictors for interacting pairs of residues by calculating Positive Predictive Values (PPV). The results were significant, with PPVs consistently close to 100% (*SI Appendix*, Fig. S5). Also important to note is the observation that the synthetic sequences generated here are not expected to have resemblance to the native family MSA's of the target proteins. We supported this by confirming that structure prediction methods that are trained on natural sequences and structures, e.g., Alphafold, are unable to fold this structure toward its target coordinates (*SI Appendix*, Fig. S6). Further verification was done using BLAST (109), which confirmed that there is no match between the artificial and natural sequences. As expected, the generated sequences are exploring an area of the sequence space that is disconnected from that explored by nature and maintained via evolutionary pressures. This phenomenon has interesting implications and is a topic for further study.

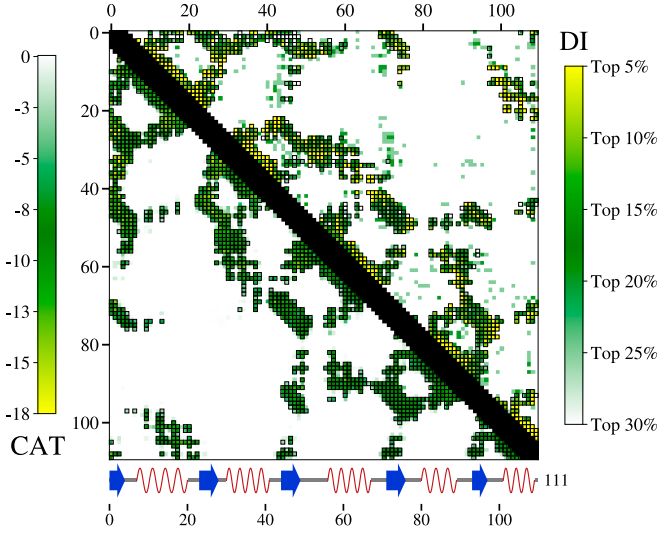

Fig. 2. Visualization of Contact Maps. A depiction of the contact map of protein 1NXW, with contacts below 12 Å highlighted as black squares. We chose 12 Å because it is the same cutoff used in the Caterpillar for the (C_α–C_α) interactions. The upper triangle of the map overlays ranked DI values onto the contact map, with higher DI values denoted in shades of yellow and lower DI values in shades closer to white. The lower triangle of the map overlays Caterpillar Res–Res interactions onto the contact map, color-coded with lower energies in yellow and higher energies in white.

Robust Correlation between DCA and Caterpillar Interaction Matrices Across Protein Families. We compared the interaction parameter coupling matrix inferred by DCA (S_{DCA}) with the actual Caterpillar matrix used in protein design. In principle, $S_{DCA}\left(A_i, A_j\right)$ should approximate the average of the interaction energy between residues i and j of type A_i and A_j, $\varepsilon_{ij}\left(A_i, A_j\right)$ over all the residue pairs $\langle \ldots \rangle_{ij}$:

$$S_{DCA}\left(A_i, A_j\right) = \left\langle \varepsilon_{ij}\left(A_i, A_j\right) / \Gamma\left(r_{ij}\right) \right\rangle_{ij} \approx S_{CAT}\left(A_i, A_j\right). \quad [13]$$

Our attempts to recover the initial interaction matrix demonstrated a good correlation between the estimated values and the ones originally used. In *SI Appendix*, Table S1, we show the Pearson correlation coefficients between the S_{CAT} and S_{DCA} across all tested protein families. The observed correlations are significant; moreover, the recovered values were consistent across the three cases studied (Fig. 3), indicating a robust correlation even when combined. Notably, no significant improvement was observed when the fit was performed on the mean values, supporting that a very similar interaction matrix was recovered regardless of the protein family. In future research, we aim to extract interaction matrices from natural sequence alignments, hoping to design proteins with natural-like properties rather than arbitrary ones. By incorporating the residue–residue interaction derived from natural sequences into the Caterpillar model, we aim to generate natural-like artificial sequences and enhance the structure prediction power of the model. This approach is supported by our belief that it is possible to extract a universal S matrix from natural sequences, a hypothesis reinforced by our findings that the S_{DCA} matrix, maintains its characteristics irrespective of the target protein, demonstrating its consistency across various

Downloaded from https://www.pnas.org by "UNIVERSITY OF TEXAS AT DALLAS, LIB SERIALS ONLINE CAPITAL" on November 12, 2024 from IP address 129.110.241.53.

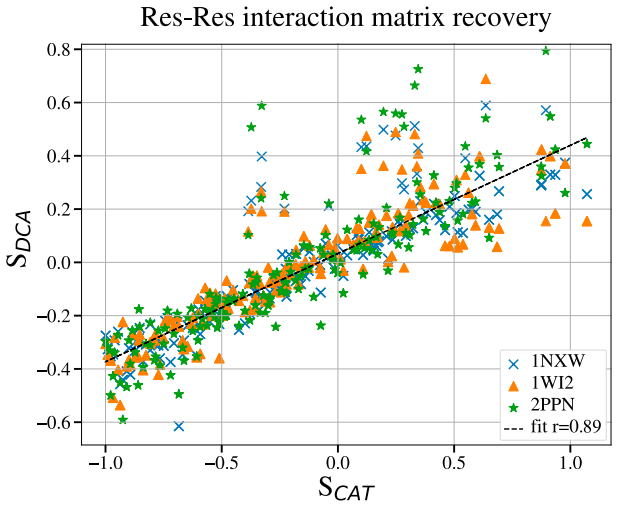

Fig. 3. Comparative analysis of S_{CAT} and S_{DCA} matrices. Comparison of the Caterpillar and the DCA interaction matrices, calculated with long-range interaction energies ([6–12]Å), across three different proteins: 1NXW, 1WI2, and 2PPN. Each data point represents a residue pair from one of the three proteins. The high correlation coefficient (0.89) suggests a strong connection between the two matrices.

protein targets. This analytical process substantiates the robustness of utilizing a proper learning scheme when trying to infer important features of molecular interactions.

In this perspective, we provide an overview of the interplay between machine learning and biological physics. We made an effort to capture the progression and state of the art of learning formulations into a consistent and unified mathematical framework. We show how broad applications in the fields of protein folding, structure prediction, dynamics, evolution, and design can be connected to such learning representations. We hope this perspective could help scientists in physics, biology, and computer sciences to communicate through this unified language and accelerate multidisciplinary collaborations and novel applications.

Data, Materials, and Software Availability. All study data are included in the article and/or *SI Appendix*. Code and data used to generate our results can be found here: https://bitbucket.org/ivan_coluzza/caterpillar-protein-design-and-folding/src/main/ (110).

ACKNOWLEDGMENTS. Work at the Center for Theoretical Biological Physics was sponsored by the NSF (Grant PHY-2019745 and PHY-2210291) and by the Welch Foundation (Grant C-1792) J.N.O. is a CPRIT Scholar in Cancer Research sponsored by the Cancer Prevention and Research Institute of Texas. I.C. thanks support of i2basque, Research and Academic Network (i2BASQUE); the HPC facility at DIPC (ATLAS) High Performance Computing (HPC) Cluster at the Donostia International Physics Center (DIPC); the support provided by SGIker (Universidad del País Vasco/Euskal Herriko Unibertsitatea/European Regional Development Fund (ERDF), EU); the HPC Europe program (EHPC-BEN-2023B07-015 and EHPC-DEV-2023D06-018); the Spanish Ministerio de Ciencia e Innovación (PID2022-139467OB-I00). This study is part of the Advanced Materials programme supported by Ministerio de Ciencia, Innovación y Universidades (MCIN), European Union NextGenerationEU (PRTR-C17.I1), and by IKUR Strategy through collaboration between Ikerbasque Foundation and Fundación BCMaterials, the Department of Education of the Basque Government. J.M. and F.M. acknowledge support from the NIH (NIH R35GM133631). F.M. acknowledges support from the NSF CAREER award (MCB-1943442). J.O. would also like to the thank the support from the Donostia International Physics Center where part of this work was performed.

1. A. Vaswani *et al.*, Attention is all you need. arXiv [Preprint] (2017). https://doi.org/10.48550/arXiv.1706.03762 (Accessed 31 August 2023).
2. OpenAI, GPT-4 Technical Report. arXiv [Preprint] (2023). https://doi.org/10.48550/arXiv.2303.08774 (Accessed 31 August 2023).
3. J. Jumper *et al.*, Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).
4. J. J. Hopfield, Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. U.S.A.* **79**, 2554–2558 (1982).
5. H. Ramsauer *et al.*, Hopfield Networks is All You Need. arXiv [Preprint] (2020). https://doi.org/10.48550/arXiv.2008.02217 (Accessed 31 August 2023).
6. S.-I. Amari, "Learning patterns and pattern sequences by self-organizing nets of threshold elements" in *IEEE Transactions on Computers* (1972), vol. C-21, pp. 1197–1206.
7. D. E. Rumelhart, J. L. McClelland, CORPORATE PDP Research Group, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition Foundations* (MIT Press, 1986), vol. 1.
8. D. E. Rumelhart, G. E. Hinton, R. J. Williams, Learning representations by back-propagating errors. *Nature* **323**, 533–536 (1986).
9. E. Ising, Beitrag zur theorie des ferromagnetismus. *Z. Angew. Phys.* **31**, 253–258 (1925).
10. L. Onsager, Crystal statistics. I. A two-dimensional model with an order–disorder transition. *Phys. Rev.* **65**, 117–149 (1944).
11. R. B. Potts, Some generalized order-disorder transformations. *Math. Proc. Cambridge Philos. Soc.* **48**, 106–109 (1952).
12. M. S. Friedrichs, P. G. Wolynes, Toward protein tertiary structure recognition by means of associative memory Hamiltonians. *Science* **246**, 371–373 (1989).
13. A. Davtyan *et al.*, AWSEM-MD: Protein structure prediction using coarse-grained physical potentials and bioinformatically based local structure biasing. *J. Phys. Chem. B* **116**, 8494–8503 (2012).
14. M. Weigt, R. A. White, H. Szurmant, J. A. Hoch, T. Hwa, Identification of direct residue contacts in protein–protein interaction by message passing. *Proc. Natl. Acad. Sci. U.S.A.* **106**, 67–72 (2009).
15. F. Morcos *et al.*, Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proc. Natl. Acad. Sci. U.S.A.* **108**, E1293–E1301 (2011).
16. F. Morcos, N. P. Schafer, R. R. Cheng, J. N. Onuchic, P. G. Wolynes, Coevolutionary information, protein folding landscapes, and the thermodynamics of natural selection. *Proc. Natl. Acad. Sci. U.S.A.* **111**, 12408–12413 (2014).
17. J. N. Onuchic, Z. Luthey-Schulten, P. G. Wolynes, Theory of protein folding: The energy landscape perspective. *Annu. Rev. Phys. Chem.* **48**, 545–600 (1997).
18. R. R. Cheng, F. Morcos, H. Levine, J. N. Onuchic, Toward rationally redesigning bacterial two-component signaling systems using coevolutionary information. *Proc. Natl. Acad. Sci. U.S.A.* **111**, E563–E571 (2014).
19. S. Ovchinnikov, H. Kamisetty, D. Baker, Robust and accurate prediction of residue–residue interactions across protein interfaces using evolutionary information. *eLife* **3**, e02030 (2014).
20. D. Krotov, J. J. Hopfield, Dense associative memory for pattern recognition. arXiv [Preprint] (2016). https://doi.org/10.48550/arXiv.1606.01164 (Accessed 31 August 2023).
21. M. Demircigil, J. Heusel, M. Löwe, S. Upgang, F. Vermet, On a model of associative memory with huge storage capacity. *J. Stat. Phys.* **168**, 288–299 (2017).
22. H. E. Stanley, Dependence of critical properties on dimensionality of spins. *Phys. Rev. Lett.* **20**, 589–592 (1968).
23. B. Hoover *et al.*, Energy Transformers. arXiv [Preprint] (2023). https://doi.org/10.48550/arXiv.2302.07253 (Accessed 31 August 2023).
24. Y. Yang, Z. Huang, D. Wipf, Transformers from an optimization perspective. arXiv [Preprint] (2022). https://doi.org/10.48550/arXiv.2205.13891 (Accessed 31 August 2023).
25. O. Press, N. A. Smith, M. Lewis, Train short, test long: Attention with linear biases enables input length extrapolation. arXiv [Preprint] (2021). https://doi.org/10.48550/arXiv.2108.12409 (Accessed 31 August 2023).
26. Z. Dai *et al.*, Transformer-XL: Attentive language models beyond a fixed-length context. arXiv [Preprint] (2019). https://doi.org/10.48550/arXiv.1901.02860 (Accessed 31 August 2023).
27. G. Delétang *et al.*, Neural networks and the Chomsky hierarchy. arXiv [Preprint] (2022). https://doi.org/10.48550/arXiv.2207.02098 (Accessed 31 August 2023).
28. M. Bal, Deep implicit attention: A mean-field theory perspective on attention mechanisms. Archive.org. https://web.archive.org/web/20230912063141/; https://mcbal.github.io/post/deep-implicit-attention-a-mean-field-theory-perspective-on-attention-mechanisms. Deposited 12 September 2023.
29. H. Attias, A variational Bayesian framework for graphical models. *Adv. Neural. Inf. Process. Syst.* **12**, 209–215 (1999).
30. M. A. Kramer, Nonlinear principal component analysis using autoassociative neural networks. *AIChE J.* **37**, 233–243 (1991).
31. D. P. Kingma, M. Welling, Auto-encoding variational Bayes. arXiv [Preprint] (2013). https://doi.org/10.48550/arXiv.1312.6114 (Accessed 31 August 2023).
32. Y. Nagano, S. Yamaguchi, Y. Fujita, M. Koyama, A wrapped normal distribution on hyperbolic space for gradient-based learning. arXiv [Preprint] (2019). https://doi.org/10.48550/arXiv.1902.02992 (Accessed 31 August 2023).
33. T. R. Davidson, L. Falorsi, N. De Cao, T. Kipf, J. M. Tomczak, Hyperspherical variational auto-encoders. arXiv [Preprint] (2018). https://doi.org/10.48550/arXiv.1804.00891 (Accessed 31 August 2023).
34. M. Kshirsagar, H. Yuan, J. L. Ferres, C. Leslie, BindVAE: Dirichlet variational autoencoders for de novo motif discovery from accessible chromatin. *Genome Biol.* **23**, 1–24 (2022).

Downloaded from https://www.pnas.org by "UNIVERSITY OF TEXAS AT DALLAS, LIB SERIALS ONLINE CAPITAL" on November 12, 2024 from IP address 129.110.241.53.

35. D. J. Rezende, S. Mohamed, Variational inference with normalizing flows. arXiv [Preprint] (2015). https://doi.org/10.48550/arXiv.1505.05770 (Accessed 31 August 2023).
36. I. Kobyzev, S. J. D. Prince, M. A. Brubaker, Normalizing flows: An introduction and review of current methods. arXiv [Preprint] (2019). https://doi.org/10.48550/arXiv.1908.09257 (Accessed 31 August 2023).
37. A. Lou *et al*., Neural manifold ordinary differential equations. *Adv. Neural. Inf. Process. Syst.* **33**, 17548–17558 (2020).
38. I. Tolstikhin, O. Bousquet, S. Gelly, B. Schoelkopf, Wasserstein auto-encoders. arXiv [Preprint] (2017). https://doi.org/10.48550/arXiv.1711.01558 (Accessed 31 August 2023).
39. A. Tong *et al*., Improving and generalizing flow-based generative models with minibatch optimal transport. arXiv [Preprint] (2023). https://doi.org/10.48550/arXiv.2302.00482 (Accessed 31 August 2023).
40. I. J. Goodfellow *et al*., Generative adversarial networks. arXiv [Preprint] (2014). https://doi.org/10.48550/arXiv.1406.2661 (Accessed 31 August 2023).
41. D. Repecka *et al*., Expanding functional protein sequence spaces using generative adversarial networks. *Nat. Mach. Intell.* **3**, 324–333 (2021).
42. N. Anand, P. Huang, Generative modeling for protein structures. *Adv. Neural Inf. Process. Syst.* **31**, 7494–7505 (2018).
43. M. Arjovsky, S. Chintala, L. Bottou, Wasserstein GAN. arXiv [Preprint] (2017). https://doi.org/10.48550/arXiv.1701.07875 (Accessed 31 August 2023).
44. C. Cecconi, E. A. Shank, C. Bustamante, S. Marqusee, Direct observation of the three-state folding of a single protein molecule. *Science* **309**, 2057–2060 (2005).
45. A. Miranker, C. V. Robinson, S. E. Radford, R. T. Aplin, C. M. Dobson, Detection of transient protein folding populations by mass spectrometry. *Science* **262**, 896–900 (1993).
46. M. Ekeberg, T. Hartonen, E. Aurell, Fast pseudolikelihood maximization for direct-coupling analysis of protein structure from many homologous amino-acid sequences. *J. Comput. Phys.* **276**, 341–356 (2014).
47. D. T. Jones, D. W. A. Buchan, D. Cozzetto, M. Pontil, PSICOV: Precise structural contact prediction using sparse inverse covariance estimation on large multiple sequence alignments. *Bioinformatics* **28**, 184–190 (2012).
48. S. Balakrishnan, H. Kamisetty, J. G. Carbonell, S.-I. Lee, C. J. Langmead, Learning generative models for protein fold families. *Proteins: Struct., Funct., Bioinf.* **79**, 1061–1078 (2011).
49. T. D. Schneider, R. Michael Stephens, Sequence logos: A new way to display consensus sequences. *Nucleic Acids Res.* **18**, 6097–6100 (1990).
50. C. Clementi, H. Nymeyer, J. Nelson Onuchic, Topological and energetic factors: What determines the structural details of the transition state ensemble and "en-route" intermediates for protein folding? An investigation for small globular proteins. *J. Mol. Biol.* **298**, 937–953 (2000).
51. J. K. Noel *et al*., SMOG2: A versatile software package for generating structure-based models. *PLoS Comput. Biol.* **12**, e1004794 (2016).
52. J. I. Sułkowska, F. Morcos, M. Weigt, T. Hwa, J. N. Onuchic, Genomics-aided structure prediction. *Proc. Natl. Acad. Sci. U.S.A.* **109**, 10340–10345 (2012).
53. R. R. Cheng, M. Raghunathan, J. K. Noel, J. N. Onuchic, Constructing sequence-dependent protein models using coevolutionary information. *Protein Sci.* **25**, 111–122 (2016).
54. D. S. Marks *et al*., Protein 3D structure computed from evolutionary sequence variation. *PLoS ONE* **6**, e28766 (2011).
55. S. Ovchinnikov *et al*., Improved de novo structure prediction in CASP11 by incorporating coevolution information into Rosetta. *Proteins: Struct., Funct., Bioinf.* **84**, 67–75 (2016).
56. B. J. Sirovetz, N. P. Schafer, P. G. Wolynes, Protein structure prediction: Making AWSEM AWSEM-ER by adding evolutionary restraints. *Proteins* **85**, 2127 (2017).
57. M. Michel *et al*., PconsFold: Improved contact predictions improve protein models. *Bioinformatics* **30**, i482–i488 (2014).
58. A. Kryshtafovych, T. Schwede, M. Topf, K. Fidelis, J. Moult, Critical assessment of methods of protein structure prediction (CASP)-Round XIV. *Proteins: Struct., Funct., Bioinf.* **89**, 1607–1617 (2021).
59. J. Wang *et al*., Scaffolding protein functional sites using deep learning. *Science* **377**, 387–394 (2022).
60. R. M. Levy, A. Haldane, W. F. Flynn, Potts Hamiltonian models of protein co-variation, free energy landscapes, and evolutionary fitness. *Curr. Opin. Struct. Biol.* **43**, 55–62 (2017).
61. M. Varadi *et al*., AlphaFold Protein Structure Database: Massively expanding the structural coverage of protein-sequence space with high-accuracy models. *Nucleic Acids Res.* **50**, D439–D444 (2022).
62. Z. Lin *et al*., Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science* **379**, 1123–1130 (2023).
63. G. Ahdritz *et al*., OpenFold: Retraining AlphaFold2 yields new insights into its learning mechanisms and capacity for generalization. bioRxiv [Preprint] (2022). https://doi.org/10.1101/2022.11.20.517210 (Accessed 31 August 2023).
64. M. Mirdita *et al*., ColabFold: Making protein folding accessible to all. *Nat. Methods* **19**, 679–682 (2022).
65. M. AlQuraishi, End-to-end differentiable learning of protein structure. *Cell Syst.* **8**, 292–301.e3 (2019).
66. M. AlQuraishi, P. K. Sorger, Differentiable biology: Using deep learning for biophysics-based and data-driven modeling of molecular mechanisms. *Nat. Methods* **18**, 1169–1180 (2021).
67. X. Ding, B. Zhang, Contrastive learning of coarse-grained force fields. *J. Chem. Theory Comput.* **18**, 6334–6344 (2022).
68. F. Noé, S. Olsson, J. Köhler, H. Wu, Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science* **365**, eaaw1147 (2019).
69. J. Köhler, Y. Chen, A. Krämer, C. Clementi, F. Noé, Flow-matching: Efficient coarse-graining of molecular dynamics without forces. *J. Chem. Theory Comput.* **19**, 942–952 (2023).
70. M. Weiel *et al*., Dynamic particle swarm optimization of biomolecular simulation parameters with flexible objective functions. *Nat. Mach. Intell.* **3**, 727–734 (2021).
71. H. Sidky, W. Chen, A. L. Ferguson, Molecular latent space simulators. *Chem. Sci.* **11**, 9459–9467 (2020).
72. X. Krithika Ravishankar, E. M. L. Jiang, F. Morcos, G. A. Cisneros, Computational compensatory mutation discovery approach: Predicting a PARP1 variant rescue mutation. *Biophys. J.* **121**, 3663–3673 (2022).
73. R. R. Cheng *et al*., Connecting the sequence-space of bacterial signaling proteins to phenotypes using coevolutionary landscapes. *Mol. Biol. Evol.* **33**, 3054–3064 (2016).
74. H. Matteo Figliuzzi, A. S. Jacquier, O. Tenaillon, M. Weigt, Coevolutionary landscape inference and the context-dependence of mutations in beta-lactamase TEM-1. *Mol. Biol. Evol.* **33**, 268–280 (2016).
75. T. A. Hopf *et al*., Mutation effects predicted from sequence co-variation. *Nat. Biotechnol.* **35**, 128–135 (2017).
76. J. A. de la Paz, C. M. Nartey, M. Yuvaraj, F. Morcos, Epistatic contributions promote the unification of incompatible models of neutral molecular evolution. *Proc. Natl. Acad. Sci. U.S.A.* **117**, 5873–5882 (2020).
77. M. Bisardi, J. Rodriguez-Rivas, F. Zamponi, M. Weigt, Modeling sequence-space exploration and emergence of epistatic signals in protein evolution. *Mol. Biol. Evol.* **39**, msab321 (2022).
78. I. Choudhuri, A. Biswas, A. Haldane, R. M. Levy, Contingency and entrenchment of drug-resistance mutations in HIV viral proteins. *J. Phys. Chem. B* **126**, 10622–10636 (2022).
79. M. Baek *et al*., Accurate prediction of protein structures and interactions using a three-track neural network. *Science* **373**, 871–876 (2021).
80. J. Rodriguez-Rivas, G. Croce, M. Muscat, M. Weigt, Epistatic models predict mutable sites in SARS-CoV-2 proteins and epitopes. *Proc. Natl. Acad. Sci. U.S.A.* **119**, e2113118119 (2022).
81. L. Vigué *et al*., Deciphering polymorphism in 61,157 Escherichia coli genomes via epistatic sequence landscapes. *Nat. Commun.* **13**, 1–14 (2022).
82. A. K. Chakraborty, J. P. Barton, Rational design of vaccine targets and strategies for HIV: A crossroad of statistical physics, biology, and medicine. *Rep. Prog. Phys.* **80**, 032601 (2017).
83. J. K. Mann *et al*., The fitness landscape of HIV-1 Gag: Advanced modeling approaches and validation of model predictions by in vitro testing. *PLoS Comput. Biol.* **10**, e1003776 (2014).
84. W. F. Flynn, A. Haldane, B. E. Torbett, R. M. Levy, Inference of epistatic effects leading to entrenchment and drug resistance in HIV-1 protease. *Mol. Biol. Evol.* **34**, 1291–1306 (2017).
85. J. Tubiana, S. Cocco, R. Monasson, Learning protein constitutive motifs from sequence data. *eLife* **8**, e39397 (2019).
86. B. Bravi *et al*., RBM-MHC: A semi-supervised machine-learning method for sample-specific prediction of antigen presentation by HLA-I alleles. *Cell Syst.* **12**, 195–202.e9 (2021).
87. M. Widrich *et al*., Modern Hopfield networks and attention for immune repertoire classification. arXiv [Preprint] (2020). https://doi.org/10.48550/arXiv.2007.13505 (Accessed 31 August 2023).
88. C. Ziegler, J. Martin, C. Sinner, F. Morcos, Latent generative landscapes as maps of functional diversity in protein sequence space. *Nat. Commun.* **14**, 2222 (2023).
89. R. M. Rao *et al*., "MSA transformer" in *International Conference on Machine Learning* (PMLR, 2021), pp. 8844–8856.
90. A. Wang, K. Cho, BERT has a mouth, and it must speak: BERT as a Markov random field language model. arXiv [Preprint] (2019). https://doi.org/10.48550/arXiv.1902.04094 (Accessed 31 August 2023).
91. U. Lupo, D. Sgarbossa, A.-F. Bitbol, Protein language models trained on multiple sequence alignments learn phylogenetic relationships. *Nat. Commun.* **13**, 6298 (2022).
92. A. Gane *et al*., ProtNLM, UniProt help. UniProt. https://www.uniprot.org/help/ProtNLM. Accessed 8 January 2024.
93. The UniProt Consortium, UniProt: The universal protein knowledgebase in 2023. *Nucleic Acids Res.* **51**, D523–D531 (2023).
94. W. P. Russ *et al*., An evolution-based model for designing chorismate mutase enzymes. *Science* **369**, 440–445 (2020).
95. X.-L. Jiang, R. P. Dimas, C. T. Y. Chan, F. Morcos, Coevolutionary methods enable robust design of modular repressors by reestablishing intra-protein interactions. *Nat. Commun.* **12**, 1–8 (2021).
96. S. Alvarez *et al*., In vivo functional phenotypes from a computational epistatic model of evolution. bioRxiv [Preprint] (2023). https://doi.org/10.1101/2023.05.24.542176 (Accessed 31 August 2023).
97. D. Sgarbossa, U. Lupo, A.-F. Bitbol, Generative power of a protein language model trained on multiple sequence alignments. *eLife* **12**, e79854 (2023).
98. A. Hawkins-Hooker *et al*., Generating functional protein variants with variational autoencoders. *PLoS Comput. Biol.* **17**, e1008736 (2021).
99. S. N. Dean, S. A. Walper, Variational autoencoder for generation of antimicrobial peptides. *ACS Omega* **5**, 20746–20754 (2020).
100. Y. Tang *et al*., Data-driven discovery of innate immunomodulators via machine learning-guided high throughput screening. bioRxiv [Preprint] (2023). https://doi.org/10.1101/2023.06.26.546393 (Accessed 31 August 2023).
101. B. I. Dahiyat, S. L. Mayo, De novo protein design: Fully automated sequence selection. *Science* **278**, 82–87 (1997).
102. C. A. Rohl, C. E. M. Strauss, K. M. S. Misura, D. Baker, Protein structure prediction using Rosetta. *Methods Enzymol.* **383**, 66–93 (2004).
103. J. Yang *et al*., Improved protein structure prediction using predicted interresidue orientations. *Proc. Natl. Acad. Sci. U.S.A.* **117**, 1496–1503 (2020).
104. J. L. Watson *et al*., De novo design of protein structure and function with RFdiffusion. *Nature* **620**, 1089–1100 (2023).
105. J. Sohl-Dickstein, E. A. Weiss, N. Maheswaranathan, S. Ganguli, Deep unsupervised learning using nonequilibrium thermodynamics. arXiv [Preprint] (2015). https://doi.org/10.48550/arXiv.1503.03585 (Accessed 31 August 2023).
106. I. Anishchenko *et al*., De novo protein design by deep network hallucination. *Nature* **600**, 547–552 (2021).
107. I. Coluzza, Transferable coarse-grained potential for de novo protein folding and design. *PLoS ONE* **9**, e112852 (2014).
108. F. Nerattini *et al*., Identification of protein functional regions. *ChemPhysChem* **21**, 335–347 (2020).
109. C. Camacho, G. M. Boratyn, V. Joukov, R. V. Alvarez, T. L. Madden, ElasticBLAST: Accelerating sequence search via cloud computing. *BMC Bioinf.* **24**, 117 (2023).
110. J. Martin, M. Lequerica-Mateos, J. N. Onuhich, I. Coluzza, F. Morcos, Sequences generated with the Caterpillar model, the S_{DCA} raw data and the residue-residue interaction matrix used to design the sequences. Bitbucket. https://bitbucket.org/ivan_coluzza/caterpillar-protein-design-and-folding/src/main/Example_PNAS_Correlation/. Deposited 8 January 2024.

Downloaded from https://www.pnas.org by "UNIVERSITY OF TEXAS AT DALLAS, LIB SERIALS ONLINE CAPITAL" on November 12, 2024 from IP address 129.110.241.53.