PNAS RESEARCH ARTICLE | BIOPHYSICS AND COMPUTATIONAL BIOLOGY

Transfer learning to leverage larger datasets for improved prediction of protein stability changes

Henry Dieckhaus[a,b], Michael Brocidiacono[b], Nicholas Z. Randolph[a,c], and Brian Kuhlman[a,c,d,1]

Edited by William DeGrado, University of California, San Francisco, CA; received August 27, 2023; accepted December 26, 2023

Amino acid mutations that lower a protein's thermodynamic stability are implicated in numerous diseases, and engineered proteins with enhanced stability can be important in research and medicine. Computational methods for predicting how mutations perturb protein stability are, therefore, of great interest. Despite recent advancements in protein design using deep learning, in silico prediction of stability changes has remained challenging, in part due to a lack of large, high-quality training datasets for model development. Here, we describe ThermoMPNN, a deep neural network trained to predict stability changes for protein point mutations given an initial structure. In doing so, we demonstrate the utility of a recently released megascale stability dataset for training a robust stability model. We also employ transfer learning to leverage a second, larger dataset by using learned features extracted from ProteinMPNN, a deep neural network trained to predict a protein's amino acid sequence given its three-dimensional structure. We show that our method achieves state-of-the-art performance on established benchmark datasets using a lightweight model architecture that allows for rapid, scalable predictions. Finally, we make ThermoMPNN readily available as a tool for stability prediction and design.

protein stability | transfer learning | deep learning | point mutation

Significance

Single amino acid mutations can have a dramatic effect on protein thermostability and, therefore, function. Mutations that unfold tumor suppressors are implicated in a variety of cancers, and mutations that stabilize proteins can be used to increase their usefulness as therapeutics or research reagents. Accurate in silico prediction of the effect of point mutations on protein stability would be useful for understanding mutations implicated in disease and for engineering protein-based medicines. In this work, we present a deep learning–based method for predicting stability changes quickly and accurately given only an initial protein structure.

Proteins are a diverse, useful class of molecules that have found roles in a variety of clinical, industrial, and research settings (1–3). Thermodynamic stability is a key property of any protein, in part because naturally evolved proteins are typically only marginally stable under ambient conditions (4). This is often sufficient for evolutionary purposes but is frequently a limiting factor in the utility of recombinant proteins for therapeutics, biocatalysis, and research applications. As a result, chemical biology investigations often seek to identify advantageous point mutations that may stabilize a candidate protein. While experimental stability optimization can be achieved using directed evolution (5), computational stability models offer the potential for faster, cheaper, and more easily scalable optimization protocols (4).

Many in silico methods have been developed to predict the effect of point mutations on protein stability given an initial structure. Historically, most methods have used empirical energy functions that model covalent and noncovalent interactions between atoms to evaluate mutations (6–8). Other widely used protocols also incorporate sequence-derived information such as a position-specific substitution matrix (9) or evolutionary consensus scoring (10). Recent efforts have attempted to apply deep learning to this problem, each with its own limitations. Convolutional neural networks (CNNs) such as ThermoNet (11) and RaSP (12) require voxelization and computationally expensive convolutions, while methods using more efficient graph neural networks (GNNs) such as ProS-GNN (13) and BayeStab (14) require generation of a modeled mutant structure for inference, adding substantial runtime and introducing opportunity for error and model bias. There is, therefore, a need for fast, robust in silico methods to predict changes in thermodynamic stability ($\Delta\Delta G°$) of potential point mutations given an initial wild-type experimental structure or high-confidence computational model.

Recent achievements using large language models (LLMs) for protein structure prediction have inspired models using prelearned sequence embeddings to train models for various protein design tasks via transfer learning (15), including for sequence-based stability prediction (16, 17). At the same time, Dauparas et al. released ProteinMPNN, a message-passing neural network (MPNN) trained on 19,700 protein clusters comprising the entire Protein Data Bank (PDB) (after quality filtering) to recover native-like sequences from a given protein backbone (18). To achieve this goal, ProteinMPNN predicts the probability of all 20 amino acids being the native residue for a given position based on learned structural patterns found in natural proteins from the PDB. Native protein sequences with known structures are assumed to be at least marginally stable under normal

Author affiliations: [a]Department of Biochemistry and Biophysics, University of North Carolina School of Medicine, Chapel Hill, NC 27599; [b]Division of Chemical Biology and Medicinal Chemistry, University of North Carolina Eshelman School of Pharmacy, Chapel Hill, NC 27599; [c]Department of Bioinformatics and Computational Biology, University of North Carolina School of Medicine, Chapel Hill, NC 27599; and [d]Lineberger Comprehensive Cancer Center, University of North Carolina School of Medicine, Chapel Hill, NC 27599

Author contributions: H.D., M.B., N.Z.R., and B.K. designed research; H.D. and M.B. performed research; H.D., M.B., N.Z.R., and B.K. analyzed data; and H.D. and B.K. wrote the paper.

The authors declare no competing interest.

This article is a PNAS Direct Submission.

[1]To whom correspondence may be addressed. Email: bkuhlman@email.unc.edu.

This article contains supporting information online at https://www.pnas.org/lookup/suppl/doi:10.1073/pnas.2314853121/-/DCSupplemental.

Published January 29, 2024.

conditions, meaning that the amino acid probabilities predicted by ProteinMPNN should correlate with the relative stabilities of the corresponding point mutants. However, evolution seldom optimizes for stability alone. Other properties such as solubility and enzymatic activity may necessitate compromises when evolving native sequences, and even when they do not, protein stability has diminishing returns beyond the bare minimum needed to ensure survival under ambient conditions (4). In a recent large-scale experimental analysis of protein stability, Tsuboyama et al. found that certain amino acids were favored over others by several orders of magnitude, even in positions where they would provide identical contributions to folding stability (19). In the same study, a basic linear regression model trained to predict native sequences based on ΔΔG° measurements achieved a sequence recovery of less than 40%. Based on these findings, we anticipated that using ProteinMPNN to naively predict ΔΔG° values is likely to be insufficient to achieve competitive performance. We, therefore, hypothesized that a model such as ProteinMPNN trained on sequence recovery may be amenable to transfer learning to enable accurate ΔΔG° prediction.

Our method, ThermoMPNN, takes advantage of knowledge overlap between the tasks of sequence recovery and stability optimization by using pretrained ProteinMPNN embeddings as input features for transfer learning. In doing so, we effectively leverage two complementary large-scale datasets: the sequence recovery dataset used to train ProteinMPNN and a dataset consisting of experimental stability measurements on hundreds of thousands of mutations from several hundred proteins (19). Until recently, the most comprehensive datasets available for training and evaluating protein stability predictors consisted of <10,000 mutations compiled from separate literature studies. Due to the combination of transfer learning and training with the Megascale dataset, ThermoMPNN learns generalizable structural determinants of stability and achieves state-of-the-art performance on a variety of benchmarks. We also investigate different training regimes to determine the contribution of each dataset and model component. Finally, we profile the prediction patterns and preferences of ThermoMPNN in comparison with its parent sequence recovery model.

Results

ThermoMPNN Architecture. ThermoMPNN (Fig. 1*A*) consists of two modules: a pretrained ProteinMPNN model (18) and a stability prediction module. As input features, ProteinMPNN uses pairwise distances between the backbone atoms of the target residue and the 48 nearest neighboring residues, encoded as Gaussian radial basis functions (RBFs). In our implementation, the wild-type (WT) amino acid sequence of the protein is also passed as a feature to ProteinMPNN. ProteinMPNN (Fig. 1 *A*, *Left box*) is a graph neural network that includes three encoder layers (light blue box) followed by three decoder layers (light gold box). During encoding, message passing between nodes (one node for each residue) and edges (each residue pair close in 3D space is connected by an edge) allows each residue to learn about its structural environment. The decoder layers incorporate the WT sequence embedding along with node and edge embeddings to predict favorable amino acids for selected residues in the input structure, with intermediate information stored in 128-dimensional embeddings in each decoder layer (dark gold boxes). In ThermoMPNN, instead of passing predictions all the way through to the sequence recovery output layer, we extract these decoder embeddings from each layer, keeping only the embeddings for the residue that is being mutated (purple bars). Due to the message-passing scheme employed in ProteinMPNN, this single residue embedding also contains information about nearby residues in structure space. This is then concatenated with the corresponding sequence embedding (green box) to produce the input vector for the stability prediction module (Fig. 1 *A*, *Right box*).

The first component of the stability prediction module is a light attention (LA) block (Fig. 1*A*, purple block). This allows ThermoMPNN to reweight the input vector using contextual information via self-attention. Light attention has recently been shown to improve sequence-based protein localization (15) and ΔΔG° prediction (16) from LLM sequence embeddings, but this work utilizes light attention for refinement of structural embeddings. The adjusted embedding is then passed through a small multilayer perceptron (MLP) with two hidden layers (Fig. 1*A*, red block) to produce a ΔΔG° value, which is calculated by subtracting a predicted wild-type ΔG° from a predicted mutant ΔG°.

Dataset Preparation. The Megascale dataset used in this study was derived from a recent study by Tsuboyama et al. in which point mutation ΔΔG° values were inferred from protease sensitivity experiments (19, 20). This approach enables derivation of reasonably accurate ΔΔG° values at megascale (776,000 data points), which is orders of magnitude larger than any previously utilized thermodynamic stability dataset derived from low-throughput biophysical assays (<10,000 data points). The ΔΔG° values must be inferred using a kinetic model, but they correlate well with previous ΔΔG° measurements, with observed Pearson correlations between 0.75 and 0.96 (19).

For this study, data curation (Fig. 1*B*) was performed as follows. In Tsuboyama et al., a significant fraction of inferred ΔΔG° values were labeled as unreliable due to various factors, such as poor expression, measurements exceeding the assay dynamic range, or poor correlation between the two proteases (19). Removal of these points left 607,839 mutations with "reliable" measurements. Next, we removed data points corresponding to deletions, insertions, and double mutants, leaving 391,090 single-substitution point mutations. Finally, about 12% of the assayed proteins consisted of modified wild-type backgrounds in which the original wild-type domain was too stable to be accurately assayed, so one or more destabilizing mutations were added (19). We chose to omit these proteins from our analysis due to concerns that the modified wild-type structure may not accurately reflect the actual structure. This is because all protein structures provided in the Megascale dataset are computationally predicted models rather than experimental structures. While multiple studies have found that structure-based stability models remain effective on high-confidence predicted structures (12, 21), they often do not accurately predict structural changes caused by point mutations (21). In addition, mutations significant enough to cause a sizable shift in wild-type stability may induce nonlocal conformational changes. After removal of these modified wild-type proteins, we obtain a final dataset of 272,712 mutations across 298 proteins (Fig. 1*B*, blue).

To evaluate model performance on thermodynamic stability data gathered with more traditional biophysical techniques and experimentally obtained structures, we aggregated an additional dataset by matching sequence-based stability data in the FireProtDB database (22) to experimental structures in the PDB (23). Data curation consisted of removing duplicates and points with missing information and then selecting the measurement for each mutation nearest to biological pH. The final dataset, which we will refer to as the Fireprot dataset, consisted of 3,438 mutations for 100 unique proteins (Fig. 1*B*, red). It should be noted that this dataset is a similar size to common literature training sets (e.g., S2648, Q3421, Q3488) with significant overlap in protein homology.

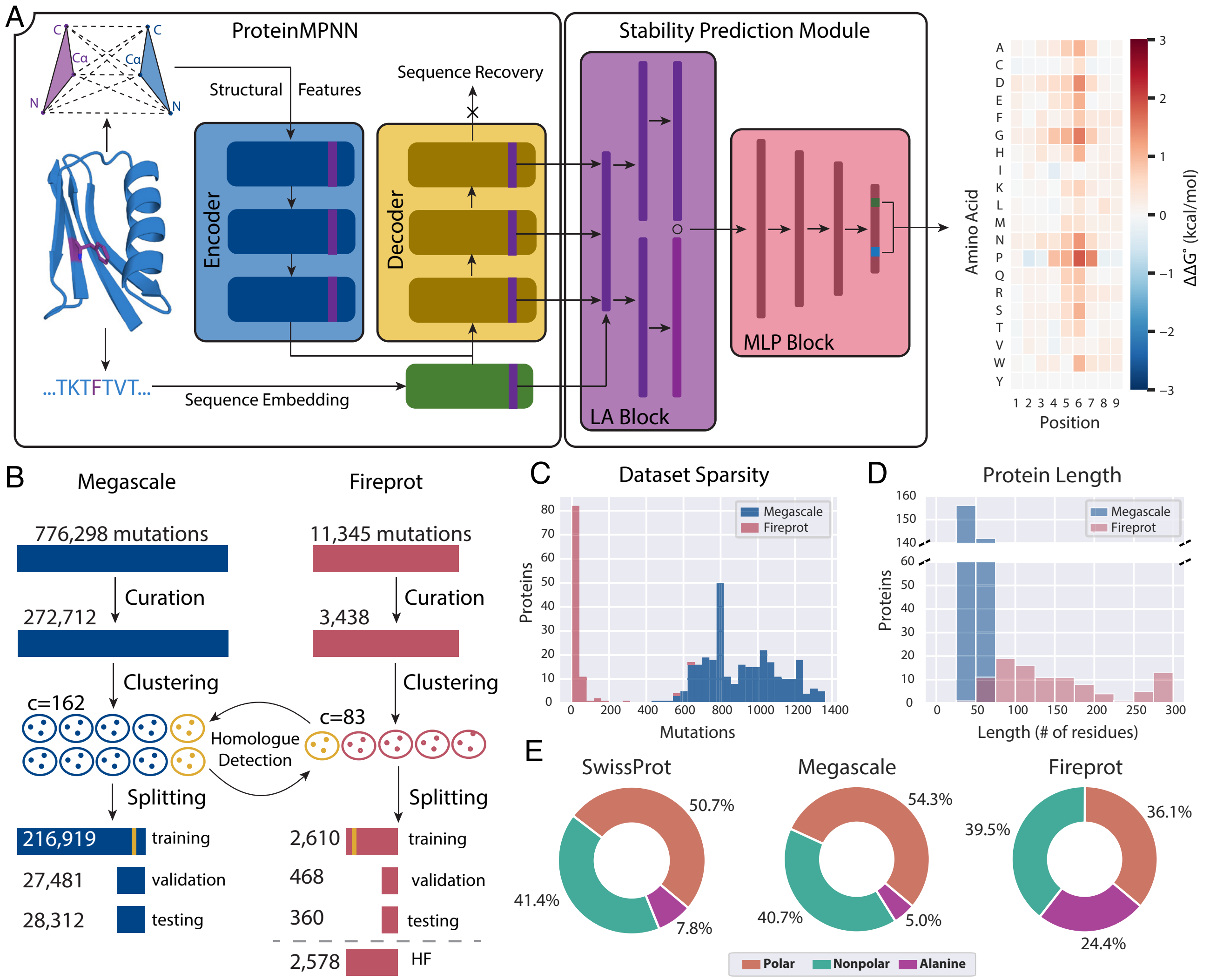

Fig. 1. ThermoMPNN architecture and primary dataset statistics. (*A*) Model architecture of ThermoMPNN, a graph neural network trained on embeddings extracted from a pretrained sequence recovery model (ProteinMPNN, *Left* panel) to predict thermostability changes caused by protein point mutations. The input protein is passed through ProteinMPNN, where the learned embeddings from each decoder layer are extracted and concatenated with the learned sequence embedding to create a vector representation of the residue environment. This vector is passed through a light attention block (LA, purple block) which uses self-attention to reweight the vector based on learned context. Finally, a multilayer perceptron (MLP, red block) predicts a ΔΔG° by subtracting the predicted ΔG° for the wild-type amino acid from the predicted ΔG° of the mutant amino acid. (*B*) Curation, clustering, and data splitting procedure for the Megascale and Fireprot datasets used in this study. Each split is labeled with its total number of mutations, and homologues are shown in yellow. Each clustering result is labeled with the number of clusters in each dataset. (*C*) Distribution of mutations per protein for each dataset. (*D*) Distribution of protein length for each dataset. (*E*) Percentage of mutations to alanine compared to other polar and nonpolar residues for each dataset, along with natural residue abundance for all proteins in the SwissProt database for comparison.

Both curated datasets were then clustered using MMseqs2 (24) with a stringent sequence identity cutoff of 25%. The two datasets were then cross-referenced to detect any homology overlap between the datasets (Fig. 1*B*, yellow), and any protein clusters with a homology match were automatically assigned to their respective training dataset. This ensured that none of the proteins in either Megascale or Fireprot test sets have any homologues in either training set. The remaining clusters were then randomly split to produce an approximately 80/10/10 split of mutations for each dataset. When splitting the Fireprot dataset, proteins with >250 data points were also automatically assigned to the training set, since their inclusion in the validation or test sets could have dominated any performance estimation, skewing the results. Finally, a fourth split, Fireprot "homologue-free" (HF), was made for evaluating models trained solely on Megascale data, by retaining all Fireprot data except for those homologous with Megascale proteins.

Dataset Statistics. The Megascale and Fireprot datasets used in this study presented multiple key differences that contribute to their combined utility for assessing model performance. First, all Megascale proteins were represented by at least 400 measurements (Fig. 1*C*), with a maximum of around 1,400 measurements representing near-total site-saturation mutagenesis coverage of a miniprotein (20 amino acids × 70 residues = 1,400 measurements). On the other hand, 85 of 100 Fireprot proteins had less than 50 measurements each, while just a few well-studied proteins (streptococcal protein G and staphylococcal nuclease) comprised 1/3 of the dataset. The Megascale dataset was also populated entirely by smaller proteins (<75 residues) (Fig. 1*D*) due to the restrictions imposed by oligonucleotide library synthesis. While the Fireprot dataset included some small proteins of a similar length (12 proteins of <75 residues), its members demonstrated both a greater average length and a wider distribution of lengths.

Table 1. ThermoMPNN ablation study results (mean ± SD) on Megascale and Fireprot datasets

Dataset	Model	RMSE (kcal/mol)	PCC	SCC
Megascale (test)	ProteinMPNN	1.295 ± 0.008	0.43 ± 0.01	0.487 ± 0.006
	ThermoMPNN	0.708 ± 0.007	0.754 ± 0.004	0.725 ± 0.003
	No light attention	0.716 ± 0.001	0.742 ± 0.002	0.711 ± 0.004
	No pretraining	0.789 ± 0.005	0.689 ± 0.005	0.642 ± 0.005
	Added fine-tuning	0.69 ± 0.01	0.780 ± 0.002	0.747 ± 0.001
Fireprot (HF)	ProteinMPNN	2.13 ± 0.02	0.41 ± 0.02	0.49 ± 0.01
	ThermoMPNN	1.51 ± 0.01	0.650 ± 0.005	0.657 ± 0.003
	No light attention	1.562 ± 0.006	0.630 ± 0.001	0.634 ± 0.001
	No pretraining	1.66 ± 0.04	0.54 ± 0.02	0.50 ± 0.02
	Added fine-tuning	1.53 ± 0.01	0.657 ± 0.006	0.666 ± 0.004

All models were independently trained and evaluated in triplicate.

Additionally, while the Fireprot dataset consists entirely of natural proteins, the Megascale dataset is composed of both natural (N = 181) and de novo–designed (N = 109) proteins, the latter subset originating from either unconstrained trRosetta hallucination (N = 19) or RosettaRemodel conditioned on several different secondary structure topologies (N = 90) (19). Secondary structure assignment using DSSP (25) (*SI Appendix*, Fig. S1) confirmed that the de novo–designed proteins exhibit a diverse set of distinct folds. The Fireprot dataset also exhibited a significant bias toward mutations to alanine (Fig. 1*E*), which comprised 24.4% of the dataset, despite appearing at a rate of only 7.8% in naturally occurring proteins, as calculated from all proteins in the SwissProt database (25). The Megascale dataset, on the other hand, contained only 5.0% mutations to alanine, resulting in a mutation profile that much more closely resembles the composition observed in SwissProt.

Ablation Study. To determine key contributions to ThermoMPNN's performance, we conducted an ablation study by removing several components of the pipeline (Table 1). We found that using the published, pretrained ProteinMPNN model alone for rank-ordering mutations was reasonably effective, with a Pearson correlation (PCC) of 0.43 ± 0.01 for the Megascale dataset and 0.41 ± 0.02 for the Fireprot dataset (mean ± SD, three independent seeds during training). The transfer-learned ThermoMPNN model improved substantially over this baseline, with corresponding PCCs of 0.754 ± 0.004 and 0.650 ± 0.005, respectively. Importantly, starting from the pretrained ProteinMPNN weights optimized for sequence recovery training was crucial for obtaining strong performance, with a significant drop-off observed (PCCs of 0.689 ± 0.005 and 0.54 ± 0.02, respectively) when training the model from naïve weights compared to transfer learning. However, attempting to optimize the pretrained ProteinMPNN weights via fine-tuning produced mixed results. Although fine-tuning improved scores on the Megascale test set (PCC of 0.780 ± 0.002), this failed to translate to the Fireprot dataset (PCC of 0.657 ± 0.006), with similar results to the transfer learning regime (RMSE of 1.53 ± 0.01, compared to 1.51 ± 0.01 without fine-tuning). Moreover, significant overfitting was also observed when fine-tuning (*SI Appendix*, Fig. S2). For these reasons, we opted to use transfer learning rather than fine-tuning for all further experiments. The LA module produced a small but consistent performance boost across both datasets, with no overfitting observed despite using a greater number of parameters (2.7 M) than ProteinMPNN (1.7 M).

Effect of Training Data on ThermoMPNN. We next examined the effect of different training datasets on ThermoMPNN performance (Fig. 2*A*) to discriminate between contributions of model architecture and dataset composition. We found that ThermoMPNN metrics were significantly degraded when trained on the Fireprot dataset, with Pearson correlations (PCCs) of 0.49 ± 0.04 and 0.35 ± 0.02 on the Megascale and Fireprot test sets, respectively, compared to 0.754 ± 0.004 and 0.650 ± 0.005 when trained on the Megascale dataset. Two methods of cotraining with both datasets were explored: concurrent training, wherein each epoch included batches from both datasets, and sequential training, wherein training on Megascale was followed by separate training epochs on Fireprot. Neither of these methods yielded any improvement over Megascale-only training. To determine what aspect(s) of Megascale data were most important for this training boost, the Megascale dataset was randomly subsampled to create a version with an equal number of proteins (Megascale SP) and a version with an equal number of total mutations (Megascale SM) to the Fireprot dataset. Models trained on each of these subsampled datasets also outperformed the Fireprot dataset when evaluated on the Megascale test set, although Megascale SM had similar performance on Fireprot test set. Notably, the Fireprot test set was more challenging to predict than the Megascale test set for all models tested.

ThermoMPNN Overall Performance. Fig. 2*B* shows the predictions of the top-performing ThermoMPNN model on the Megascale (test) and Fireprot (HF) datasets plotted against their respective experimental measurements. One important note is that the Megascale data span a substantially smaller dynamic range (−3 to 5 kcal/mol) than that of the Fireprot data (−9 to 12 kcal/mol). This means that ThermoMPNN, trained on the Megascale dataset, struggles to accurately predict $\Delta\Delta G°$ for outlier mutations. However, most Fireprot mutations (96.7%) fall within the dynamic range of the Megascale dataset, for which a fair correlation can be observed between predicted and experimental $\Delta\Delta G°$. These findings indicate a potential limitation of the Megascale dataset as a tool for training stability predictors that are not explicitly built to extrapolate beyond the dynamic range of the training set.

Examination of per-protein prediction RMSE (Fig. 2*C*) found that most proteins with <100 residues maintained an RMSE below 1.0 kcal/mol, while larger proteins were more likely to produce higher RMSEs. Despite this, 13 of 17 proteins with >100 residues still achieved an RMSE below 1.5 kcal/mol. De novo proteins achieved slightly lower average RMSE and a narrower score distribution than natural proteins, but there were also no large de novo proteins in our datasets, biasing a direct comparison. By comparison, small natural proteins (up to 63 residues, max de novo protein length) had only slightly elevated RMSE (0.66) compared to the de novo proteins (0.60).

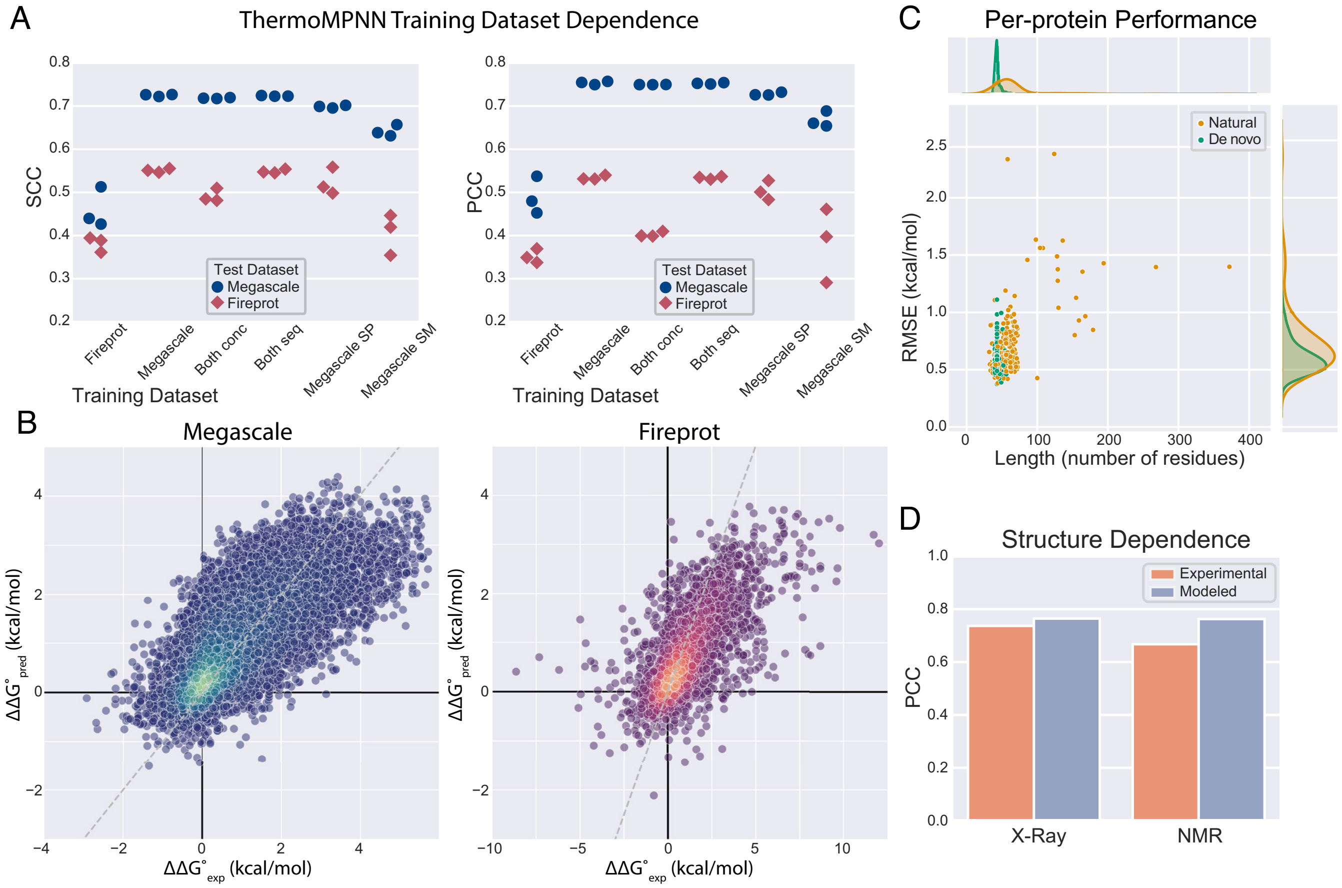

Fig. 2. ThermoMPNN performance analysis. (*A*) Spearman correlation coefficient (SCC) and PCC calculated on predictions for models trained (in triplicate) using different datasets on Megascale (blue circle) and Fireprot (red diamond) test datasets. Models were also evaluated after either concurrent (Both conc) or sequential (Both seq) cotraining, as well as training on only a subset of Megascale data including a similar number of proteins (Megascale SP) or a similar number of mutations (Megascale SM) to the Fireprot data. (*B*) ThermoMPNN predictions ($\Delta\Delta G^{\circ}_{pred}$) on the Megascale test (*Left*, *N* = 28,312) and Fireprot HF (*Right*, *N* = 2,578) datasets plotted vs. their respective experimental values ($\Delta\Delta G^{\circ}_{exp}$) with identity line (dashed gray line) for reference. Points are colored by local density of mutations (lighter = denser) using a fitted Gaussian kernel density estimate (KDE) function. (*C*) ThermoMPNN RMSE for all Megascale and Fireprot (HF) proteins (minimum 25 mutations, *N* = 320 proteins) plotted by length, with joint kernel density estimate (KDE) plots of length (*Top*) and RMSE (*Right*) distributions of natural (*N* = 211) and de novo (*N* = 109) proteins. (*D*) ThermoMPNN PCC on experimental and computationally modeled (AlphaFold) structures for all Megascale and Fireprot (HF) proteins solved by X-ray crystallography (*Left*, *N* = 113 proteins) and NMR (*Right*, *N* = 152 proteins).

ThermoMPNN Structure Dependence. Although ThermoMPNN was trained using computationally modeled structures, its performance remains strong on experimentally resolved structures (Fig. 2*D*). Similar metrics are observed for structures obtained from X-ray crystallography (PCC = 0.74) and modeled structures for the same proteins (PCC = 0.76), although NMR structures are slightly worse (PCC = 0.67) than their modeled counterparts (PCC = 0.76). No significant correlation was observed between performance (mean RMSD) and either modeled structure confidence (mean pLDDT) or crystal structure resolution (*SI Appendix*, Fig. S3). However, it should be noted that all structures included in our test sets demonstrated relatively high confidence (pLDDT > 0.75) and resolution (< 3 Å), so ThermoMPNN performance on low-quality structures remains undetermined.

ThermoMPNN Performance on Stabilizing and Inverse Mutations. We also specifically examined the capability of ThermoMPNN to predict stabilizing mutations, defined as positive predictive value (PPV), or the likelihood that a mutation predicted to be stabilizing by ThermoMPNN is indeed stabilizing. This required selection of a suitable ΔΔG° threshold for stabilizing mutations, which we observed to vary across different literature applications (26–28). If we consider a $\Delta\Delta G^{\circ} < -0.5$ kcal/mol to indicate a stabilizing mutation, ThermoMPNN achieves a PPV of 56% (34/61 predicted stabilizing mutations) on the Fireprot (HF) dataset and 46% (1,312/2,852) on the Megascale dataset. *SI Appendix*, Table S1 reports ThermoMPNN PPVs for several common ΔΔG° thresholds.

Another important property of a thermodynamic stability predictor is antisymmetry (i.e., $\Delta\Delta G^{\circ}_{direct} = -\Delta\Delta G^{\circ}_{inverse}$ where inverse is the reverse mutation at a specific residue position, for instance F35R vs. R35F). While ThermoMPNN is not explicitly constructed to be antisymmetric, performance on inverse mutation datasets is only slightly degraded, with both direct and inverse mutation performance remaining above PCC = 0.60 for Ssym, p53, and myoglobin test datasets (*SI Appendix*, Table S2). This implicit symmetrization is achieved by the final output layer of the model, which subtracts a predicted ΔG° for the wild-type amino acid from the predicted ΔG° of the mutant amino acid. A retrained "single target" ThermoMPNN shows drastically degraded inverse mutation performance (*SI Appendix*, Table S2).

We explored data augmentation strategies using modeled mutant structures to provide synthetic "inverse mutation" training examples, which modestly improved results on Ssym-inverse (PCC = 0.63) but did not improve the PPV of the model, indicating that overall capability to predict stabilizing mutations is not significantly better. We also trained a Siamese variant of ThermoMPNN which takes both a wild-type and (modeled) mutant structure as input and predicts both direct and inverse mutations simultaneously. This network used a custom loss function to penalize deviations from antisymmetric predictions, similar to the method described in Benevenuta at al. (29). Inference with this model demonstrated perfect antisymmetry and boosted Ssym-inverse performance (PCC = 0.68), but only when both wild-type and mutant structures were provided. Predictions using only the wild-type structure produced slightly better Ssym-inverse results than default ThermoMPNN (PCC = 0.62 vs. 0.60), but metrics for other datasets such as Fireprot (HF) degraded under these constraints (PCC = 0.56 vs. 0.65 and PPV = 33% vs. 56%).

Comparison of ThermoMPNN with Other Models. We next conducted several comparisons to benchmark ThermoMPNN against other published models (Fig. 3 and Tables 2 and 3). First, we tested different published methods with readily accessible code against our Megascale (test split, N = 28,312 mutations) and Fireprot (homologue-free split, N = 2,578) datasets (Table 2). We found ThermoMPNN to be the most robust predictor that we evaluated on these two datasets, with a Pearson correlation 0.04-0.06 higher than any other method on each dataset. Retraining on the Megascale dataset significantly improved both RaSP and PROSTATA scores on the Megascale test set but did not produce any improvement on the Fireprot HF set (*SI Appendix*, Table S3). More in-depth analysis of these three methods trained on the Megascale dataset revealed that the sequence-based PROSTATA outperformed structure-based RaSP but not ThermoMPNN, with larger RMSE differences observed for residues with more neighbors (i.e., in the core) (*SI Appendix*, Fig. S4, *Left*). In addition, pairwise comparison of ThermoMPNN and PROSTATA found that the former offered improvements across all mutation types, with larger gains for mutations toward hydrophobic amino acids and away from hydrophilic ones (*SI Appendix*, Fig. S4, *Right*). Taken together, these results indicate that both structure- and sequence-based methods can be trained as robust predictors with the Megascale dataset, with the former offering even greater benefit for certain types of mutations.

To further evaluate the performance of ThermoMPNN on unseen test data, we performed predictions for the widely used Ssym and S669 benchmark datasets (Table 3). These benchmarks were evaluated without filtering or other modifications, with the exception that any homologues (>25% sequence identity) detected in the Megascale training set were removed prior to retraining to prevent data leakage. Ssym is a previously curated dataset of 342 mutations with validated crystal structures of both wild-type and mutant proteins (357 total structures), enabling evaluation in both "direct" and "inverse" directions (29). We found that ThermoMPNN achieves state-of-the art performance on both direct and inverse mutations from Ssym, albeit with higher absolute scores on the direct mutations (PCCs of 0.72 and 0.60, respectively). Only one prior method, recently released graph transformer Stability Oracle, outperformed ThermoMPNN on Ssym inverse mutations (PCC = 0.72), also achieving the same score (PCC = 0.72) on direct mutations. Stability Oracle was also trained on the Megascale dataset, although the code and data splits for this method are not yet publicly available for further comparison with ThermoMPNN. The S669 dataset is another established benchmark dataset of 669 mutations from 95 proteins chosen for dissimilarity (by sequence identity) to commonly used training datasets (30). Consistent with previous works, the S669 dataset was more challenging for our model as well, with a PCC of 0.43 outperforming many structure-based methods

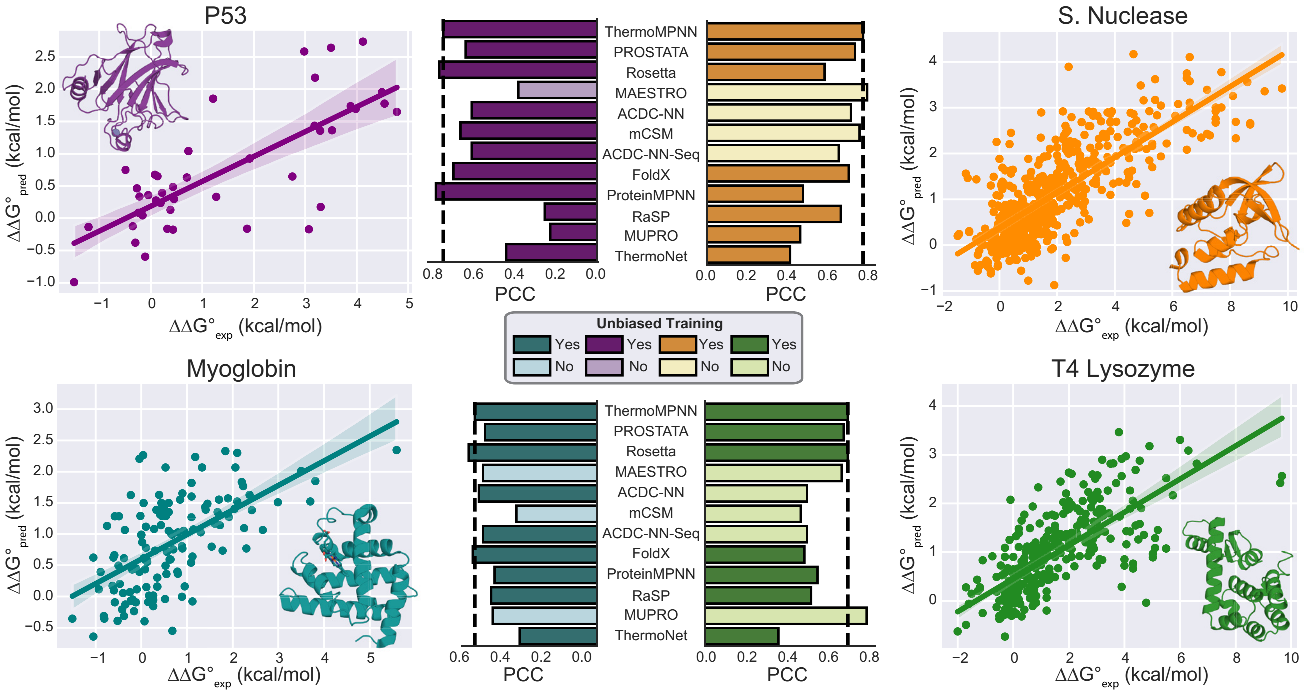

Fig. 3. Literature comparison with ThermoMPNN predictions on selected case study proteins. At middle, bar plot of PCC for ThermoMPNN and selected literature methods. Methods trained/parameterized without homologous proteins ("unbiased training") are indicated with dark bars, while those with data leakage concerns are indicated with light bars. RaSP and PROSTATA models were retrained on the Megascale dataset for this comparison. At *Left* and *Right*, scatter plots of stability change predictions ($\Delta\Delta G°_{pred}$) vs. experimental measurements ($\Delta\Delta G°_{exp}$) for external test proteins p53 (PDB ID: 2OCJ, N = 42 mutations), staphylococcus nuclease (1EY0, N = 559), myoglobin (1BZ6, N = 134), and T4 lysozyme (2LZM, N = 309) with fitted regression line (shaded region = 95% CI). The cartoon representation of each protein is included for reference.

Table 2. Comparison of ThermoMPNN performance with literature methods on Megascale and Fireprot datasets

	Megascale (test)			Fireprot (HF)		
Model	RMSE (kcal/mol)	PCC	SCC	RMSE (kcal/mol)	PCC	SCC
ProteinMPNN	1.30	0.43	0.49	2.14	0.41	0.49
ThermoMPNN	0.71	0.75	0.73	1.55	0.65	0.66
Rosetta	5.18	0.53	0.56	4.19	0.45	0.52
RaSP[†]	1.08	0.71	0.67	1.86	0.47	0.44
PROSTATA[†]	0.83	0.64	0.59	1.68	0.59	0.55
FoldX	3.87	0.40	0.57	2.77	0.43	0.57
ACDC-NN	1.05	0.52	0.45	1.69*	0.57*	0.51*
ACDC-NN-Seq	1.08	0.48	0.44	1.71*	0.54*	0.48*
ThermoNet	1.20	0.33	0.27	1.99*	0.37*	0.36*
MAESTRO	1.04	0.55	0.47	1.49*	0.62*	0.60*
mCSM	0.97	0.49	0.41	1.53*	0.59*	0.57*
MUPRO	1.08	0.31	0.29	1.54*	0.58*	0.57*

*Score may be inflated due to the presence of close homologues (>25% sequence identity) of Fireprot proteins in the training dataset.
[†]Retrained using the Megascale dataset.

(RaSP, Rosetta, etc.) but falling short of some other recent efforts (PROSTATA, ACDC-NN, etc.). Next, we evaluated ThermoMPNN in detail on four case studies (Fig. 3), including common literature examples p53 and myoglobin, as well as the two Fireprot-HF dataset proteins with the most mutations (staphylococcal nuclease, PDB ID: 1EY0; and T4 lysozyme, PDB ID: 2LZM). We found ThermoMPNN performed well on all these examples, with myoglobin producing the lowest PCC of 0.60, compared to between 0.70 and 0.77 for the other three proteins. Among methods without data leakage concerns, ThermoMPNN placed in the top 1 to 3 on all four case studies.

Exploration of ThermoMPNN Prediction Trends. To evaluate ThermoMPNN prediction trends on a large set of residues, we performed five-fold cross-validation on the Megascale dataset and analyzed the aggregated ThermoMPNN predictions obtained for all Megascale and Fireprot (HF) proteins ($N = 387$). Comparison of mutation preferences for ThermoMPNN against ProteinMPNN (Fig. 4) revealed that ThermoMPNN significantly favors mutations toward hydrophobic residues, including on the protein surface. The largest individual changes were increased preferences for isoleucine, tryptophan, and surface arginine, along with reductions in surface lysine and glutamic acid residues. This pattern is consistent with findings from Tsuboyama et al., which found that certain amino acids including isoleucine and tryptophan are underrepresented in natural protein domains relative to their thermodynamic contributions (19), while lysine and glutamic acid were the most overrepresented residues by the same metric.

Discussion

Our results support our initial hypothesis that models trained for sequence recovery can be utilized via transfer learning to achieve accurate prediction of thermodynamic stability changes. We have demonstrated how ProteinMPNN, itself a weak $\Delta\Delta G°$ predictor, can be combined with a relatively simple stability prediction module to obtain a robust $\Delta\Delta G°$ model without relying on evolutionary information or modeled mutant structures. Our optimized model, ThermoMPNN, achieved state-of-the-art performance both on our own curated datasets and on established literature

Table 3. ThermoMPNN comparison with literature methods on Ssym and S669 datasets

	Ssym (direct)		Ssym (inverse)		S669	
Model	RMSE (kcal/mol)	PCC	RMSE (kcal/mol)	PCC	RMSE (kcal/mol)	PCC
ProteinMPNN	3.38	0.26	3.20	0.56	3.32	0.26
ThermoMPNN	1.12	0.72	1.53	0.60	1.52*	0.43*
Rosetta (29, 30)	2.31	0.69	2.61	0.43	2.70	0.39
RaSP (12)	1.27	0.57	1.97	0.23	1.63	0.39
ThermoNet (11, 30)	1.56	0.47	1.55	0.47	1.62	0.39
Stability Oracle (28)	1.22	0.72	1.19	0.72	1.43	0.52
ACDC-NN (30, 36)	1.42	0.58	1.47	0.55	1.60	0.46
PROSTATA (17)	1.42	0.51	1.42	0.50	1.44	0.48
ABYSSAL (16)		0.46		0.44		0.37[†]
MAESTRO (29, 30)	1.36	0.52	2.09	0.32	1.44	0.50
FoldX (29, 30)	1.56	0.63	2.13	0.39	2.30	0.22
mCSM (29, 30)	1.23	0.61	2.43	0.14	1.54	0.36
MUPRO (29, 30)	0.94	0.79	2.51	0.07	1.61	0.25

*Model trained with a filtered homologue-free Megascale training set to avoid data leakage.
[†]Results on filtered S669 due to training set homology (420 of original 669 variants).

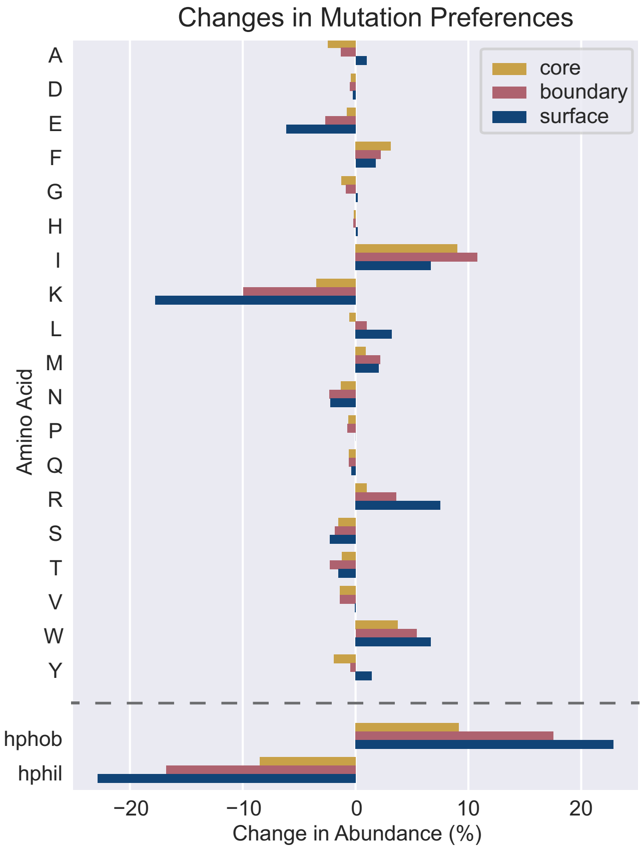

Fig. 4. Comparison of ThermoMPNN and ProteinMPNN amino acid preferences. Change in amino acid preferences of ThermoMPNN compared to ProteinMPNN if the most favorable mutation is chosen at every position for all Megascale and Fireprot (HF) proteins. Cysteine mutations were excluded due to potential disulfide formation (*SI Appendix*, Fig. S5).

benchmarks, with particularly strong performance on the symmetrical Ssym dataset (PCC = 0.72/0.60 on direct/inverse mutations) and common case study p53 (PCC = 0.76).

One advantage of a transfer learning approach is that we leverage a far larger dataset than those available for thermodynamic stability modeling. The PDB-derived dataset used to train ProteinMPNN includes 19,700 protein clusters (18), which is 100 times larger than even the Megascale stability dataset (162 clusters). This enables training of larger, deeper models such as ProteinMPNN that would be overfit on any stability dataset. Indeed, we found that naïvely retraining ThermoMPNN for ΔΔG° prediction led to overfitting, even on the Megascale dataset. This may also explain why fine-tuning the entire ProteinMPNN model from pretrained weights failed to improve ThermoMPNN predictions. By transfer learning with only the lightweight ThermoMPNN stability prediction module, we were able to avoid overfitting and fully leverage the available data to learn generalizable structural patterns. At the same time, the introduction of the Megascale dataset represents a significant opportunity for training deep neural networks for ΔΔG° prediction, as demonstrated in the training dataset exploration section. As expected, Megascale-trained models outperformed those trained on the sparse, unbalanced Fireprot dataset, and that advantage was more dependent on the total number of mutations included (sparsity) than on the number of unique proteins in the training dataset. Another recent structure-based transfer-learning method, Stability Oracle, observed similar performance boosts from both pretraining for sequence recovery and transfer learning using the larger, more robust Megascale dataset (28).

One area in which ThermoMPNN demonstrated strong performance was predicting stabilizing mutations. Among mutations in the Fireprot dataset with a predicted ΔΔG° < −0.5 kcal/mol, 56% were indeed stabilizing. While the success rate for predicted stabilizing mutations varies widely by protein and selection threshold, ThermoMPNN compares favorably based on what approximate comparisons are possible. Prior in silico studies on data derived from ProTherm (like our Fireprot dataset) report similar PPVs. Stability Oracle achieves a 53% success rate on their T2837 dataset with the same ΔΔG° threshold as ThermoMPNN (28), while an earlier study reported success rates of 32 to 53% among six other widely used ΔΔG° prediction tools (27) on a ProTherm-derived dataset. Studies involving experimental validation generally report lower success rates. A recent meta-analysis of 13 different mutagenesis studies using FoldX reported an overall success rate of 29.4% (26). Another recent study using Rosetta achieved a 50% success rate for 50 proposed mutations across three protein systems (32), although these were filtered by an expert prior to testing, meaning this is likely an optimistic estimate. In this work, we focus on ΔΔG° prediction conditioned on only wild-type structural information, which enables the impressive speed and broad utility of ThermoMPNN. Even with this constraint, ThermoMPNN achieves strong performance on inverse mutation datasets in addition to its competitive PPV on stabilizing mutations.

A few potential limitations of the Megascale dataset as a machine learning training dataset became apparent during our analysis. First, the dynamic range of the proteolysis assay is limited to ~5 kcal/mol (19), while experimental stability datasets such as our Fireprot dataset may include mutations with up to ±10 kcal/mol ΔΔG°. This means models trained on Megascale have limited capability to predict large changes in stability, a property that we also observe in other recently published models utilizing the Megascale dataset (16, 28). Second, we found that surface mutations to cysteine were often observed to be highly stabilizing in the Megascale dataset, such that ThermoMPNN would heavily favor surface cysteine mutations unless omitted from the permitted residue options (*SI Appendix*, Fig. S5). This phenomenon is a known artifact of the assay in which intermolecular disulfide bonds are introduced, disrupting detection of backbone cleavage events (33). More generally, the theoretical maximum performance of models trained on proteolysis-derived ΔΔG° values remains indeterminate, since individual protein correlations with biophysical measurements range from near-perfect (PCC=0.96) to merely good (PCC=0.75) (19).

We also examined how ThermoMPNN amino acid preferences differ from those of its parent model, ProteinMPNN. Models trained to prioritize thermodynamic stability typically favor hydrophobic mutations, particularly on the protein surface (34), while sequence recovery models must balance stability with solubility, function, and other constraints. ThermoMPNN predictions follow this trend, shifting significantly toward hydrophobic mutations in both core and surface regions when compared to ProteinMPNN. The increased preference of ThermoMPNN for surface hydrophobic residues is consistent with directed mutagenesis studies that show that surface hydrophobics can bury appreciable nonpolar surface area to lower the free energy of folding (35). To minimize the potential for hydrophobicity-induced misfolding or aggregation, we expect that ThermoMPNN will be most useful when utilized to select a small number of key point mutations to optimize natural or de novo–designed proteins with marginal stability, rather than to perform more comprehensive sequence design. Importantly, our model is similarly effective on both (small) natural and de novo–designed proteins, since it does not rely upon evolutionary information (i.e., multiple sequence alignment). ThermoMPNN is also fast, capable of running site-saturation mutagenesis in just 2 s for a small (105 residue) protein and 8 s for a large (693 residue) protein on a single GPU (*SI Appendix*, Table S4). This could enable ThermoMPNN to be integrated into iterative in silico design protocols in the future. To this end, we have released ThermoMPNN as both a standalone Python package and a web accessible Colab notebook (*SI Appendix*, Fig. S6) so that it may be useful to both developers and biologists (https://github.com/Kuhlman-Lab/ThermoMPNN). We hope that this work will serve as an example of the power of transfer learning for protein design objectives and the importance of leveraging all available data to enable widely applicable, readily repurposed models.

Methods

Dataset Curation. The Megascale dataset used in this study was derived from the updated April 2023 version released by Tsuboyama et al. (https://zenodo.org/record/7992926) (20). The raw data were curated by removing points with ΔΔG° values marked as "unreliable" (ddG_ML = "-"), as well as all insertions, deletions, and multiple mutants. Proteins were then filtered to remove any data derived from perturbed wild-type measurements to avoid using inaccurate structures. This yielded a final Megascale dataset of 272,712 mutations across 298 proteins, including 181 natural protein domains with structures modeled by AlphaFold, all of which had mean pLDDT > 0.75. The remaining proteins consisted of 90 de novo proteins generated via unconstrained trRosetta hallucination, and 19 de novo proteins generated using RosettaDesign with a variety of secondary structure topology blueprints (19). Secondary structure profiles of all Megascale dataset proteins can be found in *SI Appendix*, Fig. S1. All protein structures in the Megascale dataset were predicted computational models rather than experimental structures.

The Fireprot dataset used in this study was obtained by downloading the full FireProtDB database (https://loschmidt.chemi.muni.cz/fireprotdb/) (22). Duplicate entries and those missing key data fields (ΔΔG°, PDB ID, UniProt ID, position, wild-type residue, and mutant residue) were removed; then, the wild-type sequences were aligned to the sequences of experimental structures from the PDB (https://www.rcsb.org/) (23). Entries with multiple PDB IDs were manually disambiguated, oligomers were removed, and a small number of additional proteins were removed due to quality control issues (*SI Appendix*, Table S5). For mutations with multiple measurements, the entry (or entries) obtained under pH closest to biological pH (7.4) was selected (or averaged, as appropriate). The final Fireprot dataset consisted of 3,438 mutations across 100 proteins.

Dataset Splitting. The MMseqs2 easy-cluster tool (24) was used to cluster each dataset with a minimum sequence identity cutoff of 25%. This produced 163 clusters of 1-27 members for the Megascale dataset and 83 clusters of 1 to 3 members for the Fireprot dataset. The MMseqs2 easy-search tool was used to detect homology matches between Megascale and Fireprot, again with a 25% identity cutoff. This returned 41 total matches between 32 Megascale and 11 Fireprot proteins. Random splitting produced 80/10/10 training/validation/testing splits containing 239/31/28 proteins for Megascale and 57/15/28 proteins for Fireprot (the homologue-free split contained 89 proteins). The Fireprot proteins with >250 data points were streptococcal protein G (PDB ID: 1PGA), staphylococcal nuclease (1EY0), and T4 lysozyme (2LZM).

Dataset Analysis. Amino acid abundances for natural proteins in the SwissProt database were obtained from the literature (36) and totaled according to the following groups: polar (CDEHKNQRSTY), nonpolar (FGILMPVW), and alanine (A). Secondary structure assignments were calculated using the DSSP algorithm (25).

ThermoMPNN Architecture. ThermoMPNN (Fig. 1*A*) is a graph neural network that treats the input protein as a connected graph of nodes (residues) and edges (distances). Its first module consists of a pretrained ProteinMPNN network, which we treat as a feature extractor by freezing all parameters. Pairwise atomic distances are calculated between all backbone atoms (N, Cα, C, O, and Cβ), and the 48 nearest neighbors are collected for each residue. These distance features are passed through three message-passing encoder layers in which information is "passed" between connected nodes (i.e., nearby residues). The encoder is followed by a decoder, also composed of three layers, which utilize features from the encoder as well as any available sequence embeddings to "decode" residues one by one for sequence recovery. In ThermoMPNN, the intermediate embeddings for the target mutation position stored in each decoder layer are extracted and concatenated with the sequence embedding for the wild-type residue. This produces a vector of size 128 + (128 × *N*), where *N* is the number of decoder layers included (we use *N* = 2 for all described experiments).

The next component of ThermoMPNN is a light attention block. First, two independent padded convolutions (size = 9, stride = 1) are performed on the one-dimensional input vector. The feature convolution is fed through a dropout layer (probability = 0.25), while the attention convolution is rescaled with a softmax layer. These two outputs are then multiplied elementwise to produce a reweighted feature vector of the same size according to learned attention patterns. Finally, this vector is passed through an MLP with two hidden layers (sizes 64 and 32) to predict a ΔΔG° which is obtained by subtracting the predicted wild-type ΔG° from a predicted mutant ΔG° at the same position.

ThermoMPNN Training. Unless otherwise stated, all ThermoMPNN models were trained using an AdamW optimizer (learning rate = 0.001, weight decay = 0.01) with mean squared error (MSE) loss for 100 epochs. The ProteinMPNN model "v_48_020.pt" (48 nearest neighbors, 0.2 Å training noise) was used for all knowledge transfer experiments. Each training batch consisted of all mutations for a given protein, and all batches were sampled in each epoch. After training, the best model was selected based on the highest Pearson correlation achieved on the validation set. Training ThermoMPNN on the Megascale dataset took approximately 18 h on a single V100 GPU with 16 dedicated CPUs.

Ablation Study. ProteinMPNN rank-order mutation scores were obtained by extracting the normalized log probabilities for each mutation and multiplying them by −1. These were converted to kcal/mol by fitting a linear regression to the Megascale (training) dataset and applying this to the flipped log probabilities. To fine-tune ProteinMPNN, the parameters were unfrozen and cotrained according to the procedure described above. A hyperparameter sweep (from 1×10^{-2} to 1×10^{-6}) identified an optimal learning rate of 1×10^{-4} for the ProteinMPNN layers. All models were trained and evaluated in triplicate using three different random seeds to calculate the mean and SD of the performance metrics.

ThermoMPNN Modified Training Experiments. ThermoMPNN concurrent cotraining was implemented by training on all batches of Megascale and Fireprot training data on each epoch with all other hyperparameters held constant. For sequential cotraining, 50 epochs of Megascale training were completed as per the above procedure. Next, the learning rate was decreased from 0.001 to 0.0001, and the light attention block was frozen to only permit the final MLP layers to further train. Then, 50 epochs of training on the Fireprot dataset were completed.

For the ThermoMPNN symmetry study, single-target ThermoMPNN was trained using both wild-type and mutant sequence embeddings and a single ΔΔG° output node. Data augmentation was performed by generating mutant structures using Rosetta for the entire Megascale dataset and then adding inverse mutations using these structures and inverted ΔΔG° values. We also generated "thermodynamic permutations" as described in ref. 28 and added these to our dataset. During training, a different random 25% subset of this augmented dataset was sampled for each epoch. We followed the same procedure for our lazy data augmentation training, with the caveat that instead of using the Rosetta-modeled mutant structure, we simply threaded the mutant sequence onto the wild-type structure to create a synthetic mutant structure. To evaluate the p53 inverse and myoglobin inverse mutations, we used modeled structures from Rosetta.

To train a Siamese variant of ThermoMPNN, we reformulated our network to accept both a wild-type and mutant structure. This combined input was then passed through a shared stability prediction module to predict $\Delta\Delta G°_{direct}$ and $\Delta\Delta G°_{inverse}$, which were used to calculate a custom antisymmetric loss as described in ref. 31. To evaluate this model using only a wild-type structure, we generated a synthetic mutant structure during inference as previously described.

Comparison with Literature Methods. Several literature methods were evaluated on our Megascale and Fireprot datasets. Unless otherwise stated, all methods were used with default model weights and inference procedures. For our Rosetta benchmark, we used the recently detailed point mutation stabilization protocol (26), while the FoldX benchmark used the *PositionScan* protocol (37).

Code for RaSP (12) (https://github.com/KULL-Centre/_2022_ML-ddG-Blaabjerg), ACDC-NN including ACDC-NN-Seq (31, 38) (https://github.com/compbiomed-unito/acdc-nn), ThermoNet (11) (https://github.com/gersteinlab/ThermoNet), and PROSTATA (17) (https://github.com/mitiau/PROSTATA) were downloaded from their respective GitHub repositories. ACDC-NN inference was performed using only wild-type structure ("struct" mode) for all datasets, while ThermoNet used Rosetta-generated structures during inference. mCSM predictions were obtained from the publicly accessible webserver (39). MAESTRO (40) (https://pbwww.services.came.sbg.ac.at/) and MUpro (41) (https://download.igb.uci.edu) packages were obtained from their respective websites.

RaSP and PROSTATA were initially evaluated using the default trained weights released by their respective authors but were retrained using our Megascale training dataset for direct comparison with ThermoMPNN (*SI Appendix*, Table S3). For PROSTATA, the best scoring model (or ensemble) was included in the analysis, and a symmetrized Megascale training dataset was used. For RaSP, only the second stage ("Downstream") model was retrained on Megascale data. Performance metrics for literature methods on Ssym and S669 datasets were obtained from prior literature (citations included in Table 2).

ThermoMPNN Mutation Preferences. Protein structure regions (core/boundary/surface) were calculated by labeling each residue according to how many nearby (within 10 Å) neighbors were found (according to Cα-Cα distances). Residues with ≥20 neighbors were classified as core, those with ≤15 neighbors as surface, and those in between 20 and 15 neighbors as boundary. For the ThermoMPNN vs. PROSTATA comparison, amino acids were classified as hydrophobic (IVLFCMAW), neutral (GTSYPH), or hydrophilic (NDQEKR). For the ThermoMPNN structure-dependence study, modeled structures were generated using AlphaFold (42) with three recycles and with multiple sequence alignments (MSAs) and templates enabled.

ThermoMPNN Runtime Analysis. ThermoMPNN runtime analysis (*SI Appendix*, Table S4) was performed by averaging 5 replicate runs for each protein on a single V100 GPU with 16 GB VRAM and 8 CPUs.

Data, Materials, and Software Availability. The code for ThermoMPNN is available at https://github.com/Kuhlman-Lab/ThermoMPNN. We also provide a Colab implementation of ThermoMPNN in the same repository (*SI Appendix*, Fig. S6). The curated Fireprot dataset utilized in this study can be downloaded from the provided Zenodo data repository (https://doi.org/10.5281/zenodo.8169289) (43). Scripts for preprocessing the Fireprot dataset starting from the raw FireProtDB data are also included in the ThermoMPNN GitHub repository. All other datasets can be obtained from the ProtDDG-Bench repository (https://github.com/protddg-bench/protddg-bench) or from their respective literature sources: Ssym (29), S669 (30), p53 (39), myoglobin (44), and Megascale (20).

ACKNOWLEDGMENTS. This work was supported by NIH grant R35GM131923 (B.K.) and by NSF fellowship DGE-2040435 (N.Z.R.). We would like to thank David F. Thieker for his assistance with the Rosetta stabilize_pm protocol.

1. V. Naresh, N. Lee, A review on biosensors and recent development of nanostructured materials-enabled biosensors. *Sensors* **21**, 1109 (2021).
2. S. B. Ebrahimi, D. Samanta, Engineering protein-based therapeutics through structural and chemical design. *Nat. Commun.* **14**, 2411 (2023).
3. C. K. Winkler, J. H. Schrittwieser, W. Kroutil, Power of biocatalysis for organic synthesis. *ACS Cent. Sci.* **7**, 55–71 (2021).
4. A. Goldenzweig, S. J. Fleishman, Principles of protein stability and their application in computational design. *Annu. Rev. Biochem.* **87**, 105–129 (2018).
5. Y. Wang *et al.*, Directed evolution: Methodologies and applications. *Chem. Rev.* **121**, 12384–12444 (2021).
6. R. F. Alford *et al.*, The Rosetta all-atom energy function for macromolecular modeling and design. *J. Chem. Theory Comput.* **13**, 3031–3048 (2017).
7. S. Yin, F. Ding, N. V. Dokholyan, Eris: An automated estimator of protein stability. *Nat. Methods* **4**, 466–467 (2007).
8. R. Guerois, J. E. Nielsen, L. Serrano, Predicting changes in the stability of proteins and protein complexes: A study of more than 1000 mutations. *J. Mol. Biol.* **320**, 369–387 (2002).
9. D. Bednar *et al.*, FireProt: Energy- and evolution-based computational design of thermostable multiple-point mutants. *PLoS Comput. Biol.* **11**, e1004556 (2015).
10. A. Goldenzweig *et al.*, Automated structure- and sequence-based design of proteins for high bacterial expression and stability. *Mol. Cell* **63**, 337–346 (2016).
11. B. Li, Y. T. Yang, J. A. Capra, M. B. Gerstein, Predicting changes in protein thermodynamic stability upon point mutation with deep 3D convolutional neural networks. *PLoS Comput. Biol.* **16**, e1008291 (2020).
12. L. M. Blaabjerg *et al.*, Rapid protein stability prediction using deep learning representations. *eLife* **12**, e82593 (2023).
13. S. Wang, H. Tang, P. Shan, L. Zuo, ProS-GNN: Predicting effects of mutations on protein stability using graph neural networks. bioRxiv [Preprint] (2021). https://doi.org/10.1101/2021.10.25.465658 (Accessed 26 July 2023).
14. S. Wang, H. Tang, Y. Zhao, L. Zuo, BayeStab: Predicting effects of mutations on protein stability with uncertainty quantification. *Protein Sci.* **31**, e4467 (2022).
15. H. Stärk, C. Dallago, M. Heinzinger, B. Rost, Light attention predicts protein location from the language of life. *Bioinf. Adv.* **1**, vbab035 (2021).
16. M. A. Pak, N. V. Dovidchenko, S. M. Sharma, D. N. Ivankov, The new mega dataset combined with a deep neural network makes progress in predicting the impact of single mutations on protein stability. bioRxiv [Preprint] (2023). https://doi.org/10.1101/2022.12.31.522396 (Accessed 26 July 2023).
17. D. Umerenkov *et al.*, PROSTATA: Protein Stability assessment using transformers. bioRxiv [Preprint] (2022). https://doi.org/10.1101/2022.12.25.521875 (Accessed 26 July 2023).
18. J. Dauparas *et al.*, Robust deep learning-based protein sequence design using ProteinMPNN. *Science* **378**, 49–56 (2022).
19. K. Tsuboyama *et al.*, Mega-scale experimental analysis of protein folding stability in biology and design. *Nature* **620**, 434–444 (2023).
20. K. Tsuboyama *et al.*, Mega-scale experimental analysis of protein folding stability in biology and protein design. *Zenodo* (2023), https://doi.org/10.5281/zenodo.7992926.
21. M. A. Pak *et al.*, Using AlphaFold to predict the impact of single mutations on protein stability and function. *PLoS ONE* **18**, e0282689. (2023).
22. J. Stourac *et al.*, FireProtDB: Database of manually curated protein stability data. *Nucleic Acids Res.* **49**, D319–D324 (2021).
23. H. M. Berman *et al.*, The protein data bank. *Nucleic Acids Res.* **28**, 235–242 (2000).
24. M. Steinegger, J. Söding, MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat. Biotechnol.* **35**, 1026–1028 (2017).
25. W. Kabsch, C. Sander, Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers* **22**, 2577–2637 (1983).
26. O. Buß, J. Rudat, K. Ochsenreither, Foldx as protein engineering tool: Better than random based approaches? *Comput. Struct. Biotechnol. J.* **16**, 25–33 (2018).
27. S. Khan, M. Vihinen, Performance of protein stability predictors. *Hum. Mutat.* **31**, 675–684 (2010).
28. D. J. Diaz *et al.*, Stability oracle: A structure-based graph-transformer for identifying stabilizing mutations. bioRxiv [Preprint] (2023), https://doi.org/10.1101/2023.05.15.540857 (Accessed 26 July 2023).
29. F. Pucci, K. V. Bernaerts, J. M. Kwasigroch, M. Rooman, Quantification of biases in predictions of protein stability changes upon mutations. *Bioinformatics* **34**, 3659–3665 (2018).
30. C. Pancotti *et al.*, Predicting protein stability changes upon single-point mutation: A thorough comparison of the available tools on a new dataset. *Brief Bioinf.* **23**, bbab555 (2022).
31. S. Benevenuta, C. Pancotti, P. Fariselli, G. Birolo, T. Sanavia, An antisymmetric neural network to predict free energy changes in protein variants. *J. Phys. D, Appl. Phys.* **54**, 245403 (2021).
32. D. F. Thieker *et al.*, Stabilizing proteins, simplified: A Rosetta-based webtool for predicting favorable mutations. *Protein Sci.* **31**, e4428 (2022).
33. A. Nisthal, C. Y. Wang, M. L. Ary, S. L. Mayo, Protein stability engineering insights revealed by domain-wide comprehensive mutagenesis. *Proc. Natl. Acad. Sci. U.S.A.* **116**, 16367–16377 (2019).
34. S. Mazurenko, Predicting protein stability and solubility changes upon mutations: Data perspective. *ChemCatChem* **12**, 5590–5598 (2020).
35. D. N. Kim, T. M. Jacobs, B. Kuhlman, Boosting protein stability with the computational design of β-sheet surfaces. *Protein Sci.* **25**, 702–710 (2016).
36. S. Shen *et al.*, Probabilistic analysis of the frequencies of amino acid pairs within characterized protein sequences. *Physica A* **370**, 651–662 (2006).
37. J. Schymkowitz *et al.*, The FoldX web server: An online force field. *Nucleic Acids Res.* **33**, W382–W388 (2005).
38. C. Pancotti *et al.*, A deep-learning sequence-based method to predict protein stability changes upon genetic variations. *Genes (Basel)* **12**, 911 (2021).
39. D. E. V. Pires, D. B. Ascher, T. L. Blundell, mCSM: Predicting the effects of mutations in proteins using graph-based signatures. *Bioinformatics* **30**, 335–342 (2014).
40. J. Laimer, H. Hofer, M. Fritz, S. Wegenkittl, P. Lackner, MAESTRO–Multi agent stability prediction upon point mutations. *BMC Bioinf.* **16**, 116 (2015).
41. J. Cheng, A. Randall, P. Baldi, Prediction of protein stability changes for single-site mutations using support vector machines. *Proteins* **62**, 1125–1132 (2006).
42. J. Jumper *et al.*, Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).
43. H. Dieckhaus, M. Brocidiacono, N. Randolph, B. Kuhlman, FireProtDB + PDB structural protein stability dataset, *Zenodo* (2023), 10.5281/zenodo.8169289. Accessed 26 July 2023.
44. K. P. Kepp, Towards a "Golden Standard" for computing globin stability: Stability and structure sensitivity of myoglobin mutants. *Biochim. Biophys. Acta* **1854**, 1239–1248 (2015).