Rewiring protein sequence and structure generative models to enhance protein stability prediction

Ziang Li[1] and Yunan Luo[1,*]

[1] School of Computational Science and Engineering, Georgia Institute of Technology
[*] Corresponding author: yunan@gatech.edu

Abstract

Predicting changes in protein thermostability due to amino acid substitutions is essential for understanding human diseases and engineering useful proteins for clinical and industrial applications. While recent advances in protein generative models, which learn probability distributions over amino acids conditioned on structural or evolutionary sequence contexts, have shown impressive performance in predicting various protein properties without task-specific training, their strong unsupervised prediction ability does not extend to all protein functions. In particular, their potential to improve protein stability prediction remains underexplored. In this work, we present SPURS, a novel deep learning framework that adapts and integrates two general-purpose protein generative models–a protein language model (ESM) and an inverse folding model (ProteinMPNN)–into an effective stability predictor. SPURS employs a lightweight neural network module to rewire per-residue structure representations learned by ProteinMPNN into the attention layers of ESM, thereby informing and enhancing ESM's sequence representation learning. This rewiring strategy enables SPURS to harness evolutionary patterns from both sequence and structure data, where the sequence likelihood distribution learned by ESM is conditioned on structure priors encoded by ProteinMPNN to predict mutation effects. We steer this integrated framework to a stability prediction model through supervised training on a recently released mega-scale thermostability dataset. Evaluations across 12 benchmark datasets showed that SPURS delivers accurate, rapid, scalable, and generalizable stability predictions, consistently outperforming current state-of-the-art methods. Notably, SPURS demonstrates remarkable versatility in protein stability and function analyses: when combined with a protein language model, it accurately identifies protein functional sites in an unsupervised manner. Additionally, it enhances current low-N protein fitness prediction models by serving as a stability prior model to improve accuracy. These results highlight SPURS as a powerful tool to advance current protein stability prediction and machine learning-guided protein engineering workflows. The source code of SPURS is available at https://github.com/luo-group/SPURS.

1 Introduction

Thermodynamic stability, measured by changes in Gibbs free energy (ΔG), is a fundamental property of proteins. Characterizing protein stability is crucial for addressing biomedical challenges, from interpreting disease mutations to guiding protein engineering in industrial and clinical applications[1–3]. Although experimental techniques such as directed evolution are successful in identifying stabilizing mutations, they require substantial experimental effort to screen numerous mutants, as stabilizing mutations are rare and the mutation exploration is often unguided. This makes computational approaches, particularly machine learning (ML) methods, attractive for predicting changes in protein stability ($\Delta\Delta G$) upon amino acid substitutions, offering faster and scalable solutions for engineering stabilized proteins.

While deep learning has revolutionized protein structure prediction with models like AlphaFold[4], no similarly transformative methods have emerged for protein stability prediction. This gap largely stems from data scarcity and the limitations of current computational models. Existing ML methods for stability prediction[5–13] are often trained on small to moderate datasets[7,8,14–17], each consisting of only hundreds to thousands of mutants and covering tens to hundreds of proteins. The mismatch between the limited data and the large demands of modern ML models results in poor generalization to unseen mutations or proteins. While database efforts[15,18,19] have consolidated individual datasets, their biases toward destabilizing mutations, certain protein domains, and experimental conditions hindered progress in stability prediction.

Recent advances in mutagenesis experiments, such as complementary DNA (cDNA) proteolysis assays, offer new opportunities for ML-based protein stability prediction. For example, Tsuboyama et al. released a mega-scale dataset (hereafter “Megascale" dataset) with 776k protein folding stability measurements covering all single and selected double amino acid variants across 479 small protein domains[20]. This dataset, derived consistently from the same assay, represents an unparalleled resource for training ML models to predict protein stability. However, leveraging such data requires novel models capable of capturing the hidden thermodynamics underlying protein stability. Protein generative models, including language models (pLMs)[21–24] and inverse-folding models (IFMs)[25,26], have recently emerged as “foundation models" for various ML tasks in protein informatics. These models are pre-trained to predict masked residue amino acid using context from unmasked residues, thereby capturing evolutionary patterns or sequence likelihood distributions from vast natural protein sequences or 3D structures. Studies have shown that these models can predict mutation effects in a zero-shot manner by calculating the log-likelihood ratio between mutants and wild-type proteins[27–29], which correlates well with various measures of protein fitness, including pathogenicity[30], binding affinity[31], and thermostability[29], even without task-specific training.

Given the unsupervised stability prediction capabilities of these protein generative models, it is thus expected that combining them with large-scale stability data for supervised training would boost the prediction accuracy. However, early efforts have shown that fine-tuning pLMs on the Megascale dataset yielded only comparable or worse results than regular ML models trained from scratch[32,33]. This underperformance is partly attributed to the challenge of fine-tuning large models like pLMs, which contain $\sim 10^9$ parameters and are prone to overfitting[34] despite the availability of mega-scale datasets. Furthermore, pLMs focus only on linear amino acid sequences, lacking structural information such as 3D residue interactions that are crucial for stability prediction.

To address this, Dieckhaus et al. fine-tuned ProteinMPNN[25], a protein IFM trained on natural structures, into ThermoMPNN, leveraging structure representations for stability prediction[35]. ThermoMPNN demonstrated strong generalization to unseen proteins and has since been recognized as a leading stability predictor[36,37]. However, inverse folding models like ProteinMPNN are constrained by the availability of high-resolution protein structure data—ProteinMPNN was pre-trained on ∼20k curated structures from the CATH dataset, far fewer than the 420M proteins in UniRef[38]. Thus, it may not fully exploit evolutionary information from massive sequence data. We hypothesize that integrating pre-trained pLMs and IFMs, which harness complementary sequence and structure data, could further enhance stability prediction. However, unifying these models is a complex challenge due to differences in data modalities, model architectures, and training scales.

In this work, we introduce SPURS (stability prediction using a rewired strategy), a deep learning framework that integrates sequence- and structure-based protein generative models for protein stability prediction. To facilitate this integration, we propose a novel rewiring strategy that implants a lightweight neural network module called Adapter[39,40] into a pLM (ESM[23]), enabling it to incorporate structural priors learned by the IFM (ProteinMPNN[25]). This approach enables data-efficient adaptation of the two models to a stability predictor without overfitting. SPURS is also highly scalable, as it predicts stability changes for all point mutations at once, conditioning on the wild-type sequence and structure, contrasting to most deep learning models that predict stability changes for one mutated sequence at a time. This scalability enabled us to perform large-scale stability predictions across numerous human protein domains.

We benchmarked SPURS on 12 datasets measuring changes of thermostability ($\Delta\Delta G$) or melt temperature (ΔT_m) upon mutations and found that it outperformed state-of-the-art stability prediction methods. Notably, SPURS also excels in identifying stabilizing mutations, a challenge for most methods due to the imbalance between stabilizing and destabilizing mutations in current datasets[41]. The superior stability prediction ability of SPURS makes it a versatile method for protein function analysis. We showed that SPURS's stability predictions can be combined with pLMs for protein functional sites identification and used to enhance low-N protein fitness prediction. Overall, SPURS provides a high-performance, pre-trained model for predicting stability in unseen proteins or mutations, with wide applicability to protein engineering tasks, such as functional site discovery and fitness prediction.

2 Results

2.1 SPURS: thermostability prediction leveraging protein generative models

SPURS is a deep learning framework designed to predict changes in protein thermostability ($\Delta\Delta G$) upon point mutations. It takes as input the wild-type sequence of a target protein, $\boldsymbol{x} = (x_1, \ldots, x_L)$, where L is the protein length, $x_i \in \Sigma$ is the i-th amino acid, and Σ is the set of 20 canonical amino acids (AAs). In addition to the sequence, SPURS incorporates the 3D structure of the wild-type protein to inform its prediction. The structure is described by the coordinates of its atoms, $\mathcal{S} = \{\boldsymbol{c}_i \in \mathbb{R}^{N_b \times 3}\}_{i=1}^{L}$, where $\boldsymbol{c}_i$ is the 3D coordinates of the N_b backbone atoms (e.g., C, C_α, O, and N atoms) in the i-th residue. This structure can be abstracted as a graph, where nodes represent backbone atoms and edges are formed based on an atom-pair distance cutoff. If an experimentally determined structure is unavailable for the input protein, SPURS employs AlphaFold[4] to predict the structure.

The neural network architecture of SPURS is an effective integration of two pre-trained generative models: ESM[23], a Transformer-based protein language model (pLM), and ProteinMPNN[25], a graph neural network-based inverse-folding model (IFM). SPURS utilizes ProteinMPNN as a structure encoder to extract geometric features important to protein stability and leverage the sequence evolutionary priors learned by ESM to dissect the mutation effect on stability (Methods). To bridge these two models, SPURS employs a lightweight neural network module, called Adapter[39,40], which wires the structure embeddings from ProteinMPNN into the sequence embedding of ESM (Methods). During training, only the Adapter layer and ProteinMPNN parameters are updated, while ESM's parameters remain fixed. This integration strategy introduces only minimal architecture alterations to ESM and ProteinMPNN, preserving the rich representations and evolutionary priors learned from pre-training while avoiding overfitting. This Adapter-based approach also makes SPURS more data-efficient for fine-tuning to stability prediction, as it requires updating 98.5% fewer parameters compared to fine-tuning the entire ESM model used in previous studies[33,37]. Although this work specifically uses ESM and ProteinMPNN, SPURS is a model-agnostic framework and can integrate other sequence- and structure-based generative models[24,42] for stability prediction.

At the output layer, SPURS predicts the $\Delta\Delta G$ for all possible point mutations in a protein in a single forward pass of the neural network, generating an $L \times 20$ matrix (Methods). This scalability is a major advancement over existing models. Previous stability prediction methods[32,33,43–45] often require a single forward pass for each mutant sequence, necessitating $\mathcal{O}(L \times 20)$ forward passes to predict stability changes for all single substitutions in a protein of length L. In contrast, SPURS generalizes this one-prediction-per-pass approach to an all-prediction-per-pass approach. Instead of taking the mutant as input, SPURS predicts stability changes for all possible substitutions simultaneously by conditioning on the wild-type sequence and structure. This is achieved by learning per-residue latent representations and using a decoder, whose parameters are shared across all residues, to predict the effect of mutating each residue to all 20 AAs (including the wild-type). Consequently, SPURS reduces the number of forward passes from $\mathcal{O}(L \times 20)$ to $\mathcal{O}(1)$, enabling

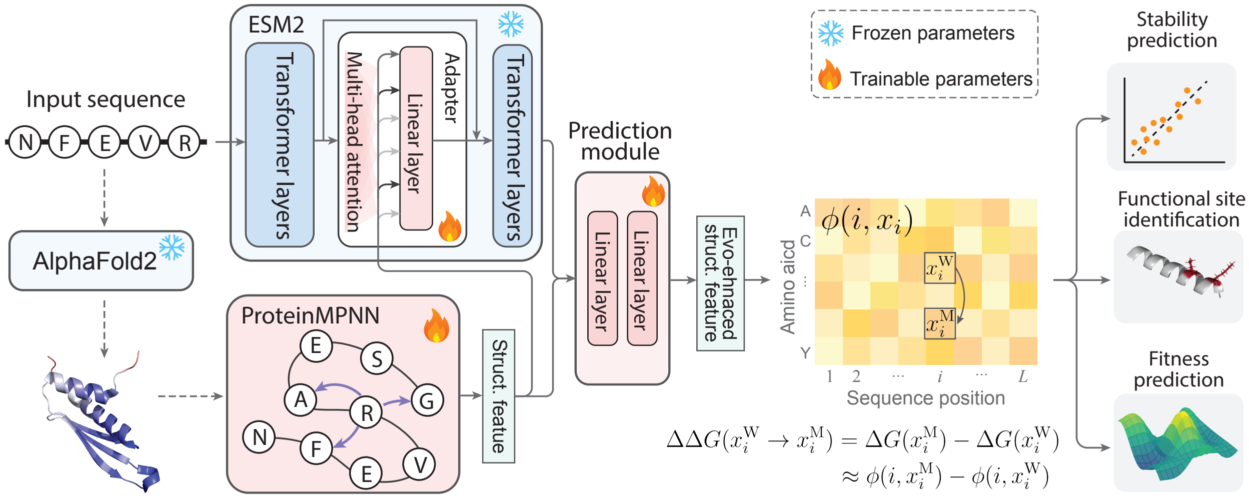

Figure 1: SPURS architecture. SPURS is a deep learning framework that rewires pre-trained protein generative models, including a protein language model (ESM2) and an inverse folding model (ProteinMPNN), to predict stability changes ($\Delta\Delta G$) upon sequence mutations. It takes a protein's wild-type sequence as input and, using AlphaFold2 for structure prediction if an experimental structure is unavailable, conditions on both of sequence and structure to predict the $\Delta\Delta G$ for all possible single-mutation variants. Through a rewiring strategy, SPURS integrates evolutionary and structural priors learned by ESM2 and ProteinMPNN to learn structure-enhanced evolutionary features, which are passed to a prediction module that outputs a matrix ϕ, allowing efficient decoding of $\Delta\Delta G$ predictions for all single mutations. SPURS's performance is demonstrated in tasks of stability prediction, functional site identification, and fitness prediction. Abbreviations: Struct=Structure; Evo=Evolution.

efficient large-scale protein stability analysis. In our experiments, SPURS predicted stability changes for all single substitutions in 118 full-length proteins from the ProteinGym database[28] (average length: 492 AAs) in just 20 seconds on an NVIDIA A40 GPU.

We trained SPURS on the Megascale stability dataset, which contains measurements of stability changes for 230,337 mutations across 241 proteins. The unprecedented size of this dataset allows SPURS to learn generalizable representations for unseen proteins and mutations. Additionally, the dense sampling of mutations per protein in the Megascale dataset (covering all single mutations) enables SPURS to effectively learn rich representations for all $L \times 20$ possible mutations in a single forward pass.

2.2 SPURS enables accurate protein stability prediction

To evaluate SPURS's performance in predicting protein stability changes upon mutations, we curated 12 datasets of stability measurements from published studies (Supplementary Methods). These datasets vary in terms of the number of proteins and the coverage of mutants measured (Fig. 2a), collectively forming a comprehensive benchmark for assessing stability prediction models. We began by using the Megascale splits[20] constructed by the ThermoMPNN study[35], which contains 272,721 single-substitution mutants with corresponding $\Delta\Delta G$ measurements across 298 proteins. These sequences were split into the training, validation, and test sets at a ratio of 80/10/10 and filtered the training and validation sets to remove sequences with greater than 25% sequence identity to those in the test, and other independent datasets (Supplementary Methods). Throughout our experiments, SPURS was trained on this filtered Megascale training set.

Model architecture ablation: We first compared SPURS with ThermoMPNN (re-trained using our training set), the state-of-the-art ML model for stability prediction, on the Megascale test set covering 28,312 mutant sequences of 28 proteins (Fig. 2b). SPURS outperformed ThermoMPNN in Spearman correlation (0.83 v.s. 0.77). This improvement can be attributed to the effective integration of ESM and ProteinMPNN by SPURS, which incorporates both sequence and structural priors, in contrast to ThermoMPNN that only fine-tunes the structure-based ProteinMPNN. An ablation study, where we fine-tuned ESM and ProteinMPNN individually using multi-layer perceptrons (MLPs) on top of frozen ESM or ProteinMPNN layers, confirmed that fine-tuning either model alone led to performance drops compared to SPURS (Fig. 2b). Interestingly, we observed that even the sequence likelihood predicted by the unsupervised ESM and ProteinMPNN exhibited a non-

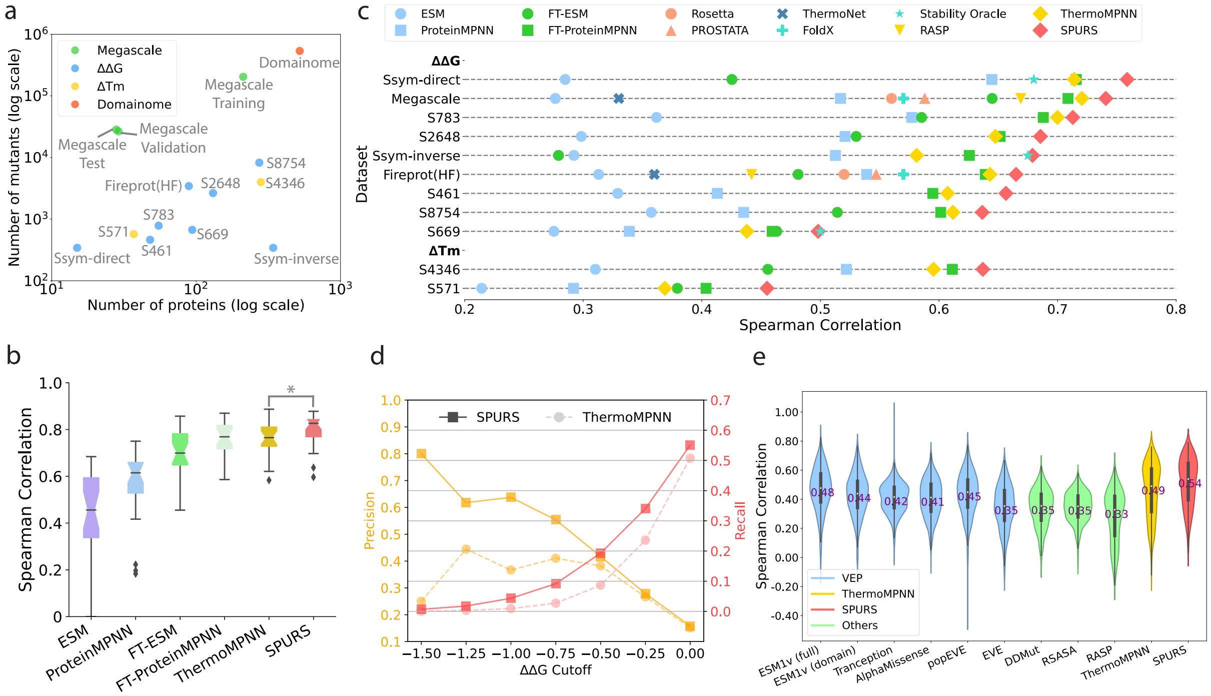

Figure 2: SPURS achieves accurate prediction for protein stability. a, The numbers of proteins and mutants across the datasets used for training and evaluation. **b,** Performance comparison of SPURS with baseline methods on the Megascale test set. ESM and ProteinMPNN use the sequence probability difference between the mutant and wild type as zero-shot stability predictors, while FT-ESM and FT-ProteinMPNN are their fine-tuned (FT) versions with supervised regression. All supervised models were trained on the filtered Megascale training set. *: Mann-Whitney U test $P < 0.05$. **c,** SPURS's performance against baselines on the Megascale test set and ten independent test datasets. **d,** Precision-recall results of stabilizing mutation identification on the Megascale test set. **e,** Comparison of SPURS with baseline methods on the Domainome dataset. ESM1v (full) takes the full-length protein sequence as input, whereas ESM1v (domain) takes domain subsequences. Abbreviations: VEP=Variant effect predictors; RSASA=Relative solvent-accessible surface area.

trivial Spearman correlation with the $\Delta\Delta G$ measurements in Megascale (0.46 and 0.61, respectively). This observation is consistent with previous studies[29] and suggests that protein generative models, even without explicit training on stability data, capture evolutionary features predictive of stability. This forms the basis of our hypothesis and other works[33,35] that supervised fine-tuning of protein generative models improves stability prediction.

Overall performance: Next, we evaluated SPURS's generalizability using eight test sets from other studies[7,8,14,17,35,43,46]. We excluded sequences in the Megascale training and validation sets that had more than 25% sequence identity with the test sets. We additionally included six leading baselines, including biophysical models (FoldX[47], Rosetta[48]), and ML methods (PROSTATA[44], RASP[32], Stability Oracle[49], ThermoNet[6]). SPURS showed significantly improved (for 7/8 test sets) or comparable performance across all datasets compared to these baselines (Fig. 2c and Supplementary Tables S1 and S2).

We then explored SPURS's ability to generalize to melting temperature (ΔT_m) prediction, another measure of protein stability. Even though SPURS was only trained on $\Delta\Delta G$ data, it demonstrated improved Spearman correlations on two ΔT_m datasets[43], S4346 and S571 (Fig. 2c). This highlights SPURS's broad capability to capture stability-related features beyond its training data.

Prioritizing stabilizing mutations: Given that stabilizing mutations are of particular interest in protein engineering, we evaluated SPURS's performance in identifying such mutations. Since destabilizing mutations dominate stability datasets, many existing methods tend to optimize predictions for these overrepresented destabilizing variants, which can lead to inflated accuracy. To address this imbalance, we evaluated SPURS's ability to prioritize stabilizing mutations (N=1,178) from a much larger pool of destabilizing mutations

(N=27,139) in the Megascale test set, where stabilizing mutations are defined as those with $\Delta\Delta G < -0.5$ kcal/mol[50]. Across various prediction cutoffs, SPURS consistently outperformed ThermoMPNN in both precision and recall (Fig. 2d), indicating its robustness in identifying stabilizing mutations.

Domainome dataset benchmark: Finally, we applied SPURS to the recently released Human Domainome dataset[36], which quantifies the impact of human missense variants on protein stability by protein abundance in cells. This dataset contains 563,534 variants across 522 proteins, offering the largest diversity and coverage among our benchmarks (Fig. 2a). The original Domainome study reported that ThermoMPNN outperformed other stability models, including general variant effect predictors, such as AlphaMissense[51] and EVE[30], structural features (relative solvent accessibility), and dedicated stability predictors. When we evaluated SPURS on this dataset, it significantly improved upon the best baseline, ThermoMPNN (correlation 0.54 vs. 0.49), further demonstrating SPURS's generalizability for protein stability prediction (Fig. 2e).

Taken together, our benchmark results demonstrate that SPURS achieved state-of-the-art performance for protein stability prediction, with superior generalizability and less bias than existing models.

2.3 SPURS identifies functionally important sites in proteins

Proteins perform various cellular functions, largely through interactions with other molecules. Identifying the specific sites or regions responsible for these interactions is essential to understanding biological processes and developing biomedical applications. Stability is just one biophysical property that contributes to protein function, while others, such as binding specificity and enzymatic activity, often trade-off with stability during evolution[52]. Thus, the loss of function due to mutations can be attributed to either direct disruption of molecular interactions or structure destabilization that leads to reduced protein abundance. Mutations at protein binding interfaces, active sites, and allosteric sites tend to have larger effects on function than what changes in stability alone can explain[20,53,54], making it challenging to deconvolve the effects of substitutions on intrinsic function from those on stability[54,55]. Some recent experimental studies attempted to resolve this biophysical ambiguity by quantifying mutation effects on both protein binding and abundance, allowing comprehensive mapping of functional sites[53,56]. Inspired by this, we hypothesized that a similar strategy, using SPURS's stability predictions alongside evolutionary fitness scores from pLMs, could help disentangle mutation effects on function and identify functional sites.

Specifically, we used SPURS to predict $\Delta\Delta G$ and ESM1v[27] to estimate the evolutionary fitness of a protein variant (Supplementary Methods). Here, 'fitness' broadly refers to protein functions like binding affinity, catalytic activity, and more. ESM1v has been shown to be effective for zero-shot predictions of mutation effects on protein fitness[27]. We applied a sigmoid function to model the non-linear relationship between stability and fitness (Fig. 3a; Supplementary Methods), following prior work that employed non-linear Boltzmann distribution to model the relationship between free energy changes caused by mutations in protein folding and those in protein binding[53,56–59]. A recent study showed that the residuals (errors) from the fitted sigmoid curve indicate whether mutations have larger or smaller effects on protein fitness than can be explained by changes in stability[36]. Our approach builds on this study by extending functional site identification beyond the restricted set of 500 protein domains with experimental stability data[36], scaling up to diverse, full-length proteins using SPURS's accurate stability predictions. We computed the fit residuals for all single mutations in a given protein (Fig. 3b) and defined a per-site *function score* by averaging residuals across all mutations at each site (Supplementary Methods). Residues with higher function scores are more likely to be functionally important[36] (Fig. 3b,c).

To evaluate SPURS's ability to identify function sites, we used 239 proteins from the Domainome dataset which have functional site annotation in the Conserved Domain Database (CDD)[60]. Out of the total 14,434 residues across these proteins, 3,516 were labeled as functional, while the rest were considered non-functional. SPURS's function score significantly distinguished functional from non-functional sites (Fig. 3d; t-test $P < 1 \times 10^{-150}$). At the individual protein level, SPURS achieved an average AUROC of 0.69. Given that SPURS was not trained on functional site labels, as previous supervised methods were[61], these results demonstrated its strong unsupervised capability for identifying functional sites in proteins.

Case studies of function site identification We further explored SPURS's predicted functional sites across seven protein domains that ranked high or mid in AUROC (Fig. 3e). These proteins were chosen to cover human domains that represent various sizes from 56 to 97 residues, diverse structural folds, and different

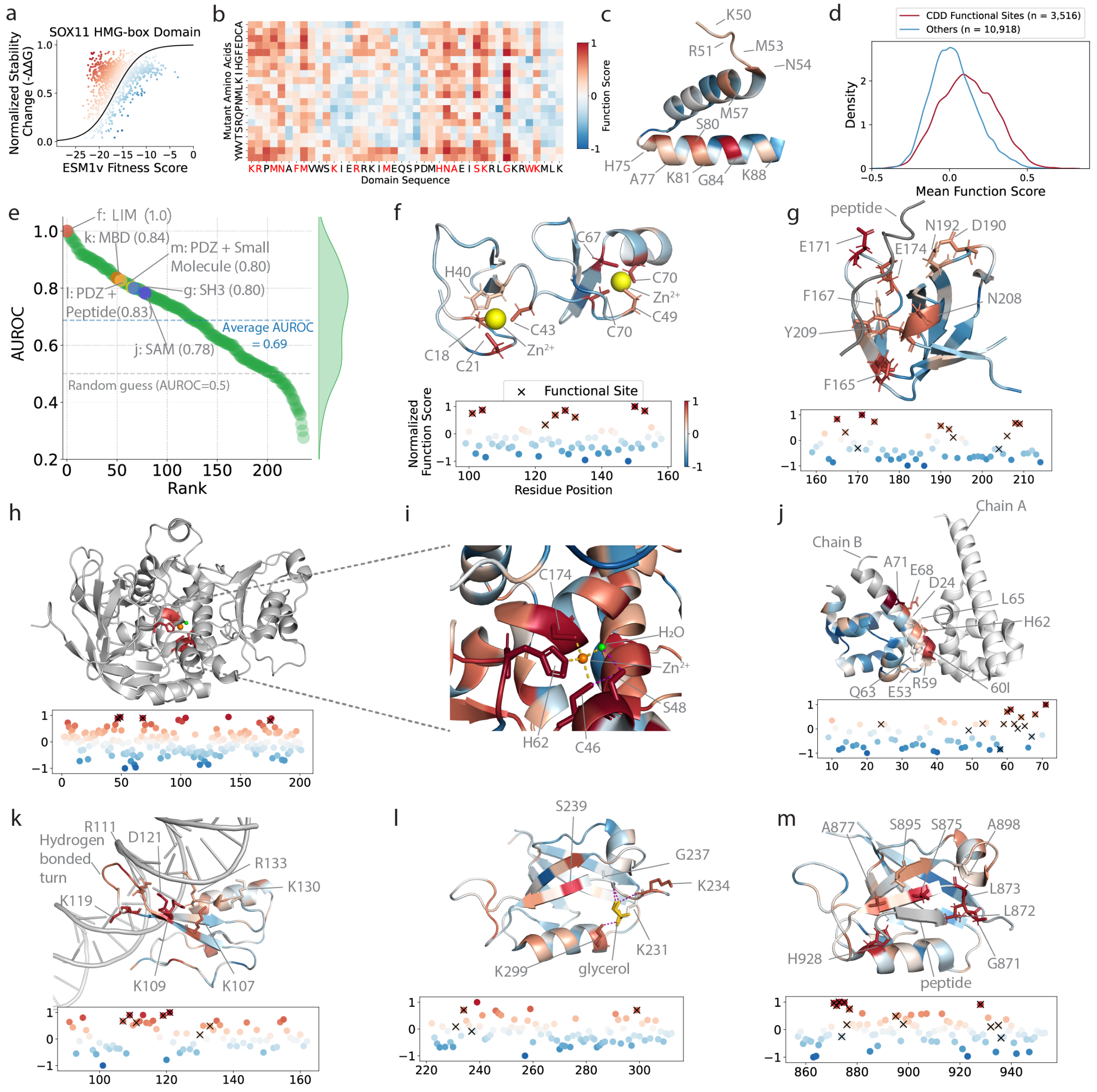

Figure 3: SPURS accurately guides functional site annotation. **a**, ESM1v-predicted fitness scores and min-max normalized stability changes (-$\Delta\Delta G$) predicted by SPURS for the HMG-box domain (UniProt ID: P35716). A sigmoid function is used to fit the relationship between fitness and normalized stability change. Color gradient represents SPURS's predicted function score. **b**, Heatmap depicting function scores of the HMG-box domain, with red letters indicating CDD-annotated DNA-binding sites. **c**, Structure of the HMG-box domain (PDB ID: 6T7C), where residues are colored by function scores and DNA-binding sites are annotated. **d**, Distribution of function scores for functional sites annotated in CDD and other sites across 239 human domains. **e**, SPURS's AUROC performance for the functional site annotations across 239 human domains. **f**-**m**: The structures colored by SPURS's function scores for the six highlighted dots in **e**. Below each structure, a scatter plot shows the predicted function score for each residue, with CDD-annotated functional sites marked as ×. **f**, LIM domain in FHL1 (UniProt ID: Q13642; PDB ID: 1X63). **g**, SH3 domain in GRB2 (P62993, 1IO6). **h**, Alcohol dehydrogenase (P00327, 1QLH). **i**, Zoom-in view of the zinc ion around the catalytic center. **j**, SAM domain in CNKSR2 (Q8WXI2, 3BS5). **k**, MBD domain in MECP2 (P51608, 5BT2), **l**, PDZ domain in DLG3 (Q92796, 2FE5), **m**, PDZ domain in SCRIB (Q14160, 6XA7).

functions, including protein-protein interaction, DNA binding, ion interaction, and enzymatic activity. By mapping SPURS's function score onto their 3D structure, we found high-score regions significantly enriched for functional sites (Figs. 3f-m).

Annotation for diverse functions: In the LIM domain, which contains two four-residue zinc fingers, SPURS assigned high scores exclusively to the zinc-coordinating residues (Fig.3f). Similarly, in the SH3 domain, SPURS accurately captured its critical peptide-binding sites (Figs. 3g). We extended this analysis to proteins not presented in Domainome, such as alcohol dehydrogenase (EC 1.1.1.1), a zinc-dependent enzyme. SPURS successfully identified its key zinc-binding residues (C46, H62, and C174) and a stabilizing residue (S48), which forms hydrogen bonds critical for the enzyme's structure (Fig. 3h-i)

Identification of binding sites: Notably, despite being trained only on single-chain inputs, SPURS identified interaction interfaces and binding sites. For example, it highlighted SAM domain residues near the heterodimer interaction interface (Fig. 3j), even though its interacting partner's sequence or structure was not provided to SPURS. A similar observation was made for the MBD domain, where SPURS not only recovered DNA binding sites but also identified a β turn located between two β-strands and close to the DNA helix (Fig. 3k). While CDD did not annotate this β-turn as functional sites, it appeared to coordinate the protein-DNA binding, suggesting the potential of SPURS for discovering new functional sites.

Annotation consistency: SPURS also demonstrated consistency across different proteins harboring the same domain. For example, in two proteins with PDZ domains (Figs. 3l-m), SPURS assigned high function score to several residues that occupy structurally corresponding positions in the two proteins. A serine residue (S239 in Fig. 3l and S875 in Fig. 3m) consistently received the highest function score in both structures despite their different structural contexts. While SPURS prioritized consistent functional sites for both proteins, it also captured the context-dependent nature of binding and gave higher scores to residues involved in specific ligand interactions unique to each protein. In one case where the domain binds to a small glycerol molecule (Fig.3l), a lysine (K299) interacting with the molecule received a higher score than its counterpart in the other protein, while in the second case where the same domain binds to a peptide (Fig.3m), SPURS prioritized residues (L872, H928) close to the peptide. Importantly, the ligands were only shown for visualization and were not provided as input to SPURS. Nonetheless, SPURS was able to identify both conserved functional sites for the same domain across proteins and context-dependent sites specific to each protein's binding function.

These case studies illustrate SPURS's strong agreement with CDD annotations and its ability to identify both conserved and context-specific functional sites. Unlike previous methods that rely on docking structures[62] or supervised learning[61], our approach is unsupervised, requiring only sequence input, with structural predictions generated by AlphaFold when needed. This approach alleviates data bottlenecks, making it a powerful tool for applications like hotspot identification in protein engineering.

2.4 SPURS improves low-*N* protein fitness prediction

After establishing SPURS's accuracy in predicting protein stability changes, we explored whether it could extend to enhancing the prediction of mutation effects on general protein properties beyond stability. Many laboratory assays have been developed to measure various protein properties like binding affinity, expression, and solubility, often generally referred to as fitness. However, experimental techniques can only probe a tiny fraction of the exponentially large sequence space and screen their fitness, making it critical in protein engineering to develop ML models that generalize well from small-sized (low-N) fitness data to predict for unseen sequences[26,34,63].

Here, we aim to improve low-N fitness prediction models with SPURS. Proteins need to be structurally stable to perform functions. We thus hypothesized that SPURS's stability predictions could serve as a strong prior for fitness prediction. We propose a simple yet effective approach that incorporates SPURS's $\Delta\Delta G$ prediction to improve protein fitness prediction. Our approach was inspired by a leading supervised low-N fitness prediction model called 'Augmented model'[26]. To predict the fitness of a protein variant, the Augmented model uses the one-hot-encoded sequence and an evolutionary density score, which is the likelihood ratio between the mutant and the wild-type sequences from pLMs like ESM or other sequence density models (e.g., DeepSequence[64] or EVE[30]), to train a Ridge regressor on fitness data. We extended the Augmented model by incorporating SPURS's $\Delta\Delta G$ predictions as an additional feature in the regressor (Fig. 4a bottom;

Supplementary Methods). We referred to our enhanced model as SPURS-augmented ESM if ESM is used as the sequence density model, or similarly if other models are used.

We compared SPURS-augmented models with a series of Augmented models, including Augmented-{ESM1b[21], EVMutation[65], Evotuned UniRep[66], DeepSequence[64]}, on the fitness data of 12 proteins from the original study[26]. Using 240 mutants for training and the rest for testing, we found that SPURS-augmented models outperformed their counterparts for most proteins, with a 7% improvement in Spearman correlation (Fig. 4b). To further assess performance, we used the ProteinGym benchmark[28], which includes 200+ deep mutational scanning (DMS) datasets of protein fitness, spanning different functions such as enzyme catalytic activity, binding affinity, stability, and organismal fitness. We excluded the DMS datasets measuring stability, since we focused on the performance improvements enabled by SPURS for general fitness beyond stability. Across the resulting 133 DMS datasets, our SPURS-augmented DeepSequence model outperformed Augmented DeepSequence–the most competitive method reported[26]–in 123 cases, with an overall 10% improvement in Spearman correlation (Fig. 4c). Performance improvements were highest in datasets measuring expression (13.0%) and organismal fitness (11.8%) compared to activity (6.5%) and binding (7.3%) (Fig. 4d). These improvements were consistent across varying training set size N from 48 to 240 samples, and to 80% of the total sequences in a DMS dataset (Fig. 4e).

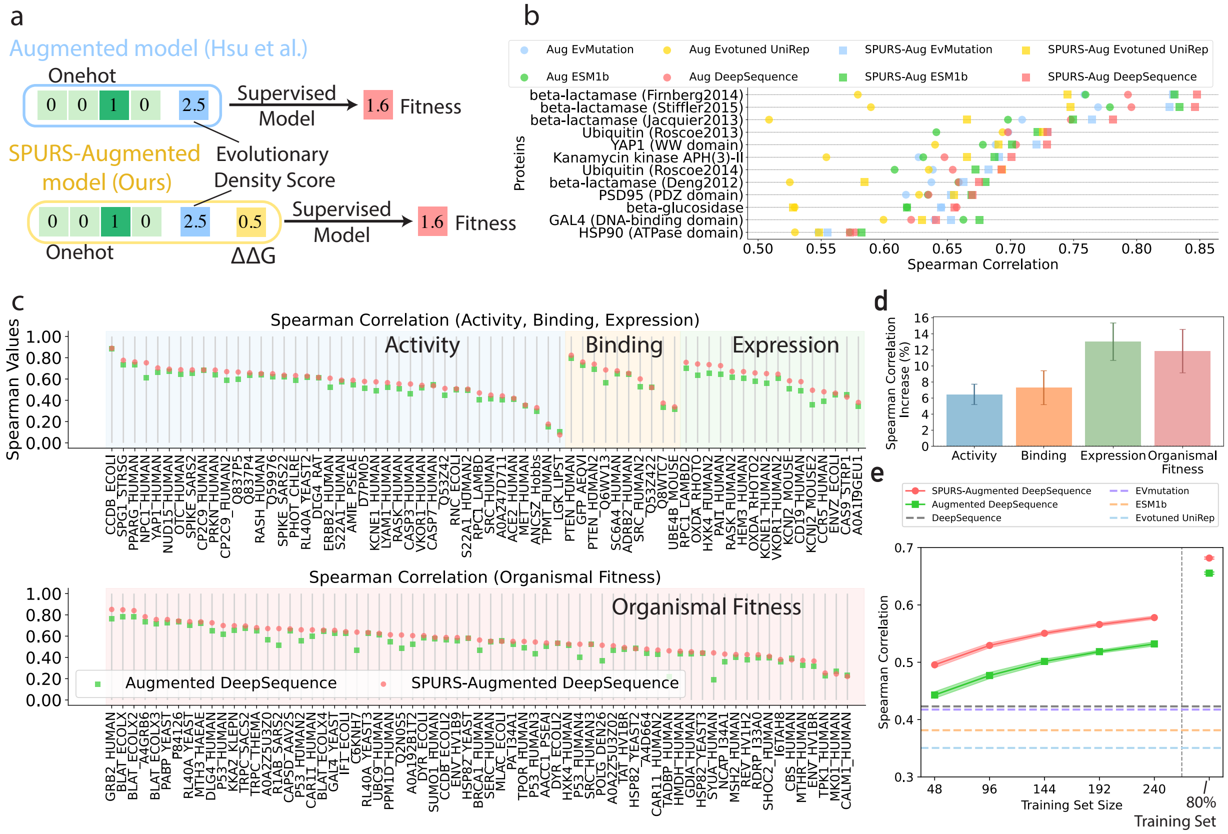

Figure 4: SPURS-augmented models enhances low-N protein fitness prediction. **a**, Architecture of the SPURS-augmented model. Compared to the Augmented models developed by Hsu et al.[26], our approach incorporates SPURS's predicted stability change as an additional feature. **b**, Comparison of SPURS-augmented models (squares) and corresponding Augmented models (circles) on 12 proteins benchmarked in Hsu et al.[26]. Aug=Augmented. **c**, Performances of SPURS-augmented DeepSequence and Augmented DeepSequence across all Datasets in ProteinGym, excluding stability-related DMS datasets or those with fewer than 240 mutants. DMS dataset names on x-axis were abbreviated, and a mapping to their full names defined by ProteinGym is provided in Table S3. **d,** Relative Spearman correlation improvement of SPURS-augmented DeepSequence compared to Augmented DeepSequence across different fitness categories from **c**. **e,** Low-N prediction performance of SPURS-augmented DeepSequence and Augmented DeepSequence across varying training set size N. In **d**-**e**, the dot, bar, or line plots represent the mean±s.d. of the test performance of 20 repetitions of model training, with each trained on a randomly sampled training set of 240 mutants using different random seeds.

In summary, we showed that the SPURS's $\Delta\Delta G$ predictions provide useful priors for protein fitness prediction, consistently enhancing leading low-N models across various DMS datasets and training data sizes. We note that our SPURS-augmented model itself is not designed to be a stand-alone state-of-the-art predictor that outperforms a large volume of existing low-N models that leverage sophisticated deep learning models (e.g., pLMs)[26,34,63]. Instead, the evaluation here was designed to show that SPURS offers a simple yet effective enhancement to already-competitive models.

3 Conclusion

We have presented SPURS, a novel deep learning framework for protein stability prediction. SPURS effectively integrates ESM and ProteinMPNN to predict $\Delta\Delta G$ values using a neural network rewiring strategy. Our experiments showed that SPURS outperforms many state-of-the-art stability predictors across multiple benchmarks, providing accurate, rapid, scalable, and generalizable stability predictions. We also demonstrated that SPURS, when combined with a protein language model (pLM), can accurately identify protein functional sites in an unsupervised manner. Furthermore, SPURS serves as a stability prior, enhancing the accuracy of existing low-N protein fitness prediction models. Although we showcased its integration with ProteinMPNN and ESM2, the SPURS framework is versatile and can be adapted to incorporate other inverse folding models (IFMs)[42] and pLMs[21,67]. While this work focused on single-mutation variants due to the availability of stability data, future extensions could include predictions for multi-mutation or indel variants by aggregating individual mutation effects, as done in other methods[68,69].

4 Methods

4.1 Representations of input sequence and structure

SPURS takes a protein's wild-type sequence as input and predicts thermostability change ($\Delta\Delta G$) for all possible single-mutation variants (Fig. 1). SPURS additionally incorporates the wild-type 3D structure of the target protein when available. If the experimentally determined structure is not accessible, SPURS uses AlphaFold[4] to predict the structure. In our experiments, experimental structures were used for the proteins in Fig. 4b and AlphaFold-predicted structures for other experiments, in line with previous studies[28,35].

SPURS integrates ProteinMPNN[25] as a structure encoder and ESM2[23] as both a sequence encoder and mutation effect predictor. ProteinMPNN is a message-passing neural network (MPNN)[70,71], a type of graph neural network, which learns representations from a protein structure. It treats the structure as a graph, where nodes are backbone atoms (N, C_α, C, O, and C_β) and edges are formed by connecting each atom to its 48 nearest neighbors. This structure graph is passed through the MPNN consisting of three encoder and three decoder layers, producing a per-residue embedding in $\mathbb{R}^{L\times 128}$, where L is the protein length. To improve representation learning, we replaced the autoregressive decoding in the original ProteinMPNN model with a masked decoding scheme (Supplementary Methods). We concatenated MPNN's output embedding with its internal residue embeddings ($\mathbb{R}^{L\times 128}$) learned based on the amino acid identity, resulting in a combined structure features $\boldsymbol{z}_s \in \mathbb{R}^{L\times 256}$. ESM2, a 33-layer Transformer[72], encodes a protein sequence as a per-residue embedding $\boldsymbol{z}_e \in \mathbb{R}^{L\times 1280}$. Since ESM2 is pre-trained on massive natural sequence data, the output embeddings $\boldsymbol{z}_e$ capture evolutionary patterns of protein language, referred to as evolutionary features.

4.2 Model rewiring to learn structure-enhanced evolutionary features

To inject the structure feature $\boldsymbol{z}_s \in \mathbb{R}^{L\times 1280}$ captured by ProteinMPNN into the evolutionary feature $\boldsymbol{z}_e \in \mathbb{R}^{L\times 1280}$ learned by ESM2, SPURS rewires ProteinMPNN into ESM2 to create interactions between the structure and evolutionary features, resulting in a structure-enhanced evolutionary feature $\boldsymbol{z}_a \in \mathbb{R}^{L\times 1280}$. This rewiring mechanism is achieved by a parameter-efficient neural network module called Adapter[40], which has been proven effective for fine-tuning large language models and for protein sequence design[39,73]. The Adapter layer can be represented as

$$\boldsymbol{z}_a = \boldsymbol{W}_2 \cdot \mathrm{GELU}(\boldsymbol{W}_1 \cdot (\mathrm{MultiHead}(\boldsymbol{z}_e, \boldsymbol{z}_s, \boldsymbol{z}_s) + \boldsymbol{z}_e) + \boldsymbol{b}_1) + \boldsymbol{b}_2, \tag{1}$$

where $\text{MultiHead}(Q, K, V)$ is the multi-head attention layer[72] with query embedding Q, key embedding K, and value embedding V. Here, we used $\boldsymbol{z}_e$ as the query and $\boldsymbol{z}_s$ as both key and value. The attention output is added back to $\boldsymbol{z}_e$ through a residual connection, followed by a two-layer MLP parameterized by $\boldsymbol{W}_{\{1,2\}}$ and $\boldsymbol{b}_{\{1,2\}}$, with a GELU activation function[74], to produce the structure-enhanced evolutionary features $\boldsymbol{z}_a$. By learning to extract and integrate features from both structural and evolutionary contexts, SPURS is expected to provide more informed stability prediction with the enhanced feature $\boldsymbol{z}_a$. The Adapter is inserted after the 31st layer of ESM2 (out of 33 layers), with the position determined by a hyperparameter search on the validation set. After applying the Adapter, $\boldsymbol{z}_a$ is passed through the remaining layers of ESM2, then projected from $\mathbb{R}^{L\times 1280}$ to $\mathbb{R}^{L\times 128}$ via a linear layer. It is then concatenated with the structure feature $\boldsymbol{z}_s \in \mathbb{R}^{L\times 256}$ to reinforce the structure prior, resulting in the final protein embedding $\boldsymbol{z}_o \in \mathbb{R}^{L\times 384}$.

Building on previous findings that fine-tuning only the Adapter layer can achieve comparable performance to fine-tuning the entire Transformer model[39,40], we froze the ESM2 parameters (650 million parameters) and optimized only the Adapter and ProteinMPNN parameters (9.9 million parameters). This approach (Fig. 1) reduced training cost by 98.5% compared to updating the full SPURS model, without compromising prediction accuracy.

4.3 Efficient stability prediction module

To predict the stability changes, the embedding $\boldsymbol{z}_o$ is passed through a multi-layer perceptron (MLP) $g : \mathbb{R}^{384} \to \mathbb{R}^{20}$, with each output dimension corresponds to one of the 20 amino acids (AAs). This MLP, shared across all L positions in the protein sequence, projects $\boldsymbol{z}_o \in \mathbb{R}^{L\times 384}$ to a matrix $\boldsymbol{\phi} = g(\boldsymbol{z}_o) \in \mathbb{R}^{L\times 20}$. The element of this $L \times 20$ matrix $\boldsymbol{\phi}$ is indexed by the sequence position and AA type, in which $\boldsymbol{\phi}(i, a)$ represents a trainable weight that approximates the thermostability (ΔG) when the residue at position i is mutated to amino acid a.

This matrix $\boldsymbol{\phi}$ can be used to derive the change in ΔG (i.e., $\Delta\Delta G$) upon single mutations. Let $\boldsymbol{x} = (x_1, \ldots, x_L)$ be a protein sequence of length L, where $x_i \in \Sigma$ is the i-th AA, and Σ is the set of 20 canonical AAs. Denote $\boldsymbol{x}^{\text{WT}}$ and $\boldsymbol{x}^{\text{MT}}$ as the wild-type sequence and its single-mutation variant resulting from the substitution $x_i^{\text{WT}} \to x_i^{\text{MT}}$ at position i. The stability change of $\boldsymbol{x}^{\text{MT}}$ with respect to $\boldsymbol{x}^{\text{WT}}$ is defined as

$$\Delta\Delta G(\boldsymbol{x}^{\text{MT}}|\boldsymbol{x}^{\text{WT}}) = \Delta G(x_i = x_i^{\text{MT}}|\boldsymbol{x}^{\text{WT}}) - \Delta G(x_i = x_i^{\text{WT}}|\boldsymbol{x}^{\text{WT}}). \tag{2}$$

In analogy, SPURS (denoted as f_θ parameterized by θ) predicts the $\Delta\Delta G$ for the single-mutation variant $\boldsymbol{x}^{\text{MT}}$, conditioned on the protein's wild-type sequence $\boldsymbol{x}^{\text{WT}}$ and structure $\mathcal{S}^{\text{WT}}$, as following:

$$f_\theta(\boldsymbol{x}^{\text{MT}}|\boldsymbol{x}^{\text{WT}}, \mathcal{S}^{\text{WT}}) = \boldsymbol{\phi}(i, x_i^{\text{MT}}) - \boldsymbol{\phi}(i, x_i^{\text{WT}}). \tag{3}$$

This formulation enables efficient and scalable $\Delta\Delta G$ prediction: the matrix $\boldsymbol{\phi}$ only needs to be computed once in a single forward pass of the neural network, and then can be reused efficiently to derive the $\Delta\Delta G$ for all single mutations using Eq. 3. In contrast, many existing methods use the mutant sequence as input, which requires $\mathcal{O}(L \times 20)$ forward passes to predict $\Delta\Delta G$ for all $L \times 20$ single-mutation variants. By conditioning on the wild-type sequence and structure, SPURS predicts all single-mutation variants altogether in one pass, significantly improving prediction efficiency. This approach, also used by some recent studies[34,69], is well-suited for large-scale protein stability analysis.

SPURS was trained using a mean squared error (MSE) loss to minimize the difference between predicted and experimentally measured $\Delta\Delta G$ values. Each batch contained all mutants corresponding to a single wild-type protein. The model was trained for a maximum of 200 epochs on an NVIDIA A40 GPU using the AdamW optimizer[75] with a learning rate of 0.0001. A plateau scheduler was used for adaptive learning rate adjustment, and early stopping was employed to prevent overfitting by terminating training once the validation performance was not improved for 30 epochs. Hyperparameters such as batch size, learning rate, and optimizer settings were fine-tuned using the Megascale validation set.

Acknowledgements This work is supported in part by the National Institute of General Medical Sciences of the National Institutes of Health under award R35GM150890 and a Seed Grant Program from GaTech IDEaS. This work used the Delta GPU Supercomputer at NCSA of UIUC through allocation CIS230097 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program, which is supported by NSF grants #2138259, #2138286, #2138307, #2137603, and #2138296. The authors acknowledge the computational resources provided by Microsoft Azure through the Cloud Hub program at GaTech IDEaS and the Microsoft Accelerate Foundation Models Research (AFMR) program.

References

[1] Mirfath Sultana Mesbahuddin, Aravindhan Ganesan, and Subha Kalyaanamoorthy. Engineering stable carbonic anhydrases for co2 capture: a critical review. *Protein Engineering, Design and Selection*, 34:gzab021, 2021.

[2] Shuke Wu, Radka Snajdrova, Jeffrey C Moore, Kai Baldenius, and Uwe T Bornscheuer. Biocatalysis: enzymatic synthesis for industrial applications. *Angewandte Chemie International Edition*, 60(1):88–119, 2021.

[3] Sisi Shan, Shitong Luo, Ziqing Yang, Junxian Hong, Yufeng Su, Fan Ding, Lili Fu, Chenyu Li, Peng Chen, Jianzhu Ma, et al. Deep learning guided optimization of human antibody against sars-cov-2 variants with broad neutralization. *Proceedings of the National Academy of Sciences*, 119(11):e2122954119, 2022.

[4] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589, 2021.

[5] Lijun Quan, Qiang Lv, and Yang Zhang. Strum: structure-based prediction of protein stability changes upon single-point mutation. *Bioinformatics*, 32(19):2936–2946, 2016.

[6] Bian Li, Yucheng T Yang, John A Capra, and Mark B Gerstein. Predicting changes in protein thermodynamic stability upon point mutation with deep 3d convolutional neural networks. *PLoS computational biology*, 16(11):e1008291, 2020.

[7] Yuting Chen, Haoyu Lu, Ning Zhang, Zefeng Zhu, Shuqin Wang, and Minghui Li. Premps: Predicting the impact of missense mutations on protein stability. *PLoS computational biology*, 16(12):e1008543, 2020.

[8] Corrado Pancotti, Silvia Benevenuta, Giovanni Birolo, Virginia Alberini, Valeria Repetto, Tiziana Sanavia, Emidio Capriotti, and Piero Fariselli. Predicting protein stability changes upon single-point mutation: a thorough comparison of the available tools on a new dataset. *Briefings in Bioinformatics*, 23(2):bbab555, 2022.

[9] Dmitriy Umerenkov, Fedor Nikolaev, Tatiana I Shashkova, Pavel V Strashnov, Maria Sindeeva, Andrey Shevtsov, Nikita V Ivanisenko, and Olga L Kardymon. Prostata: a framework for protein stability assessment using transformers. *Bioinformatics*, 39(11):btad671, 2023.

[10] Gen Li, Sijie Yao, and Long Fan. Prostage: predicting effects of mutations on protein stability by using protein embeddings and graph convolutional networks. *Journal of Chemical Information and Modeling*, 64(2):340–347, 2024.

[11] S Benevenuta, C Pancotti, P Fariselli, G Birolo, and T Sanavia. An antisymmetric neural network to predict free energy changes in protein variants. *Journal of Physics D: Applied Physics*, 54(24):245403, 2021.

[12] Corrado Pancotti, Silvia Benevenuta, Valeria Repetto, Giovanni Birolo, Emidio Capriotti, Tiziana Sanavia, and Piero Fariselli. A deep-learning sequence-based method to predict protein stability changes upon genetic variations. *Genes*, 12(6):911, 2021.

[13] Ludovica Montanucci, Emidio Capriotti, Yotam Frank, Nir Ben-Tal, and Piero Fariselli. Ddgun: an untrained method for the prediction of protein stability changes upon single and multiple point variations. *BMC bioinformatics*, 20:1–10, 2019.

[14] Yves Dehouck, Aline Grosfils, Benjamin Folch, Dimitri Gilis, Philippe Bogaerts, and Marianne Rooman. Fast and accurate predictions of protein stability changes upon mutations using statistical potentials and neural networks: Popmusic-2.0. *Bioinformatics*, 25(19):2537–2543, 2009.

[15] Joicymara S Xavier, Thanh-Binh Nguyen, Malancha Karmarkar, Stephanie Portelli, Pâmela M Rezende, Joao PL Velloso, David B Ascher, and Douglas EV Pires. Thermomutdb: a thermodynamic database for missense mutations. *Nucleic acids research*, 49(D1):D475–D479, 2021.

[16] Rahul Nikam, A Kulandaisamy, K Harini, Divya Sharma, and M Michael Gromiha. Prothermdb: thermodynamic database for proteins and mutants revisited after 15 years. *Nucleic acids research*, 49(D1):D420–D424, 2021.

[17] Jan Stourac, Juraj Dubrava, Milos Musil, Jana Horackova, Jiri Damborsky, Stanislav Mazurenko, and David Bednar. Fireprotdb: database of manually curated protein stability data. *Nucleic acids research*, 49(D1):D319–D324, 2021.

[18] M Michael Gromiha, Jianghong An, Hidetoshi Kono, Motohisa Oobatake, Hatsuho Uedaira, Ponraj Prabakaran, and Akinori Sarai. Protherm, version 2.0: thermodynamic database for proteins and mutants. *Nucleic acids research*, 28(1):283–285, 2000.

[19] Connie Y Wang, Paul M Chang, Marie L Ary, Benjamin D Allen, Roberto A Chica, Stephen L Mayo, and Barry D Olafson. Protabank: A repository for protein design and engineering data. *Protein Science*, 27(6):1113–1124, 2018.

[20] Kotaro Tsuboyama, Justas Dauparas, Jonathan Chen, Elodie Laine, Yasser Mohseni Behbahani, Jonathan J Weinstein, Niall M Mangan, Sergey Ovchinnikov, and Gabriel J Rocklin. Mega-scale experimental analysis of protein folding stability in biology and design. *Nature*, 620(7973):434–444, 2023.

[21] Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C Lawrence Zitnick, Jerry Ma, et al. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15):e2016239118, 2021.

[22] Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, et al. Prottrans: Toward understanding the language of life through self-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):7112–7127, 2021.

[23] Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Sal Candido, et al. Language models of protein sequences at the scale of evolution enable accurate structure prediction. *BioRxiv*, 2022:500902, 2022.

[24] Tomas Hayes, Roshan Rao, Halil Akin, Nicholas J Sofroniew, Deniz Oktay, Zeming Lin, Robert Verkuil, Vincent Q Tran, Jonathan Deaton, Marius Wiggert, et al. Simulating 500 million years of evolution with a language model. *bioRxiv*, pages 2024–07, 2024.

[25] Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J Ragotte, Lukas F Milles, Basile IM Wicky, Alexis Courbet, Rob J de Haas, Neville Bethel, et al. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378(6615):49–56, 2022.

[26] Chloe Hsu, Hunter Nisonoff, Clara Fannjiang, and Jennifer Listgarten. Learning protein fitness models from evolutionary and assay-labeled data. *Nature biotechnology*, 40(7):1114–1122, 2022.

[27] Joshua Meier, Roshan Rao, Robert Verkuil, Jason Liu, Tom Sercu, and Alex Rives. Language models enable zero-shot prediction of the effects of mutations on protein function. *Advances in neural information processing systems*, 34:29287–29303, 2021.

[28] Pascal Notin, Aaron W Kollasch, Daniel Ritter, Lood van Niekerk, Steffanie Paul, Hansen Spinner, Nathan Rollins, Ada Shaw, Ruben Weitzman, Jonathan Frazer, et al. Proteingym: Large-scale benchmarks for protein design and fitness prediction. *bioRxiv*, 2023.

[29] Shawn Reeves and Subha Kalyaanamoorthy. Zero-shot transfer of protein sequence likelihood models to thermostability prediction. *Nature Machine Intelligence*, 6(9):1063–1076, 2024.

[30] Jonathan Frazer, Pascal Notin, Mafalda Dias, Aidan Gomez, Joseph K Min, Kelly Brock, Yarin Gal, and Debora S Marks. Disease variant prediction with deep generative models of evolutionary data. *Nature*, 599(7883):91–95, 2021.

[31] Brian L Hie, Varun R Shanker, Duo Xu, Theodora UJ Bruun, Payton A Weidenbacher, Shaogeng Tang, Wesley Wu, John E Pak, and Peter S Kim. Efficient evolution of human antibodies from general protein language models. *Nature Biotechnology*, 42(2):275–283, 2024.

[32] Lasse M Blaabjerg, Maher M Kassem, Lydia L Good, Nicolas Jonsson, Matteo Cagiada, Kristoffer E Johansson, Wouter Boomsma, Amelie Stein, and Kresten Lindorff-Larsen. Rapid protein stability prediction using deep learning representations. *Elife*, 12:e82593, 2023.

[33] Simon KS Chu, Kush Narang, and Justin B Siegel. Protein stability prediction by fine-tuning a protein language model on a mega-scale dataset. *PLOS Computational Biology*, 20(7):e1012248, 2024.

[34] Junming Zhao, Chao Zhang, and Yunan Luo. Contrastive fitness learning: Reprogramming protein language models for low-n learning of protein fitness landscape. In *International Conference on Research in Computational Molecular Biology*, pages 470–474. Springer, 2024.

[35] Henry Dieckhaus, Michael Brocidiacono, Nicholas Z Randolph, and Brian Kuhlman. Transfer learning to leverage larger datasets for improved prediction of protein stability changes. *Proceedings of the National Academy of Sciences*, 121(6):e2314853121, 2024.

[36] Toni Beltran, Xianger Jiang, Yue Shen, and Ben Lehner. Site saturation mutagenesis of 500 human protein domains reveals the contribution of protein destabilization to genetic disease. *bioRxiv*, pages 2024–04, 2024.

[37] Talal Widatalla, Rafael Rafailov, and Brian Hie. Aligning protein generative models with experimental fitness via direct preference optimization. *bioRxiv*, pages 2024–05, 2024.

[38] UniProt. Uniprot: the universal protein knowledgebase in 2021. *Nucleic acids research*, 49(D1):D480–D489, 2021.

[39] Zaixiang Zheng, Yifan Deng, Dongyu Xue, Yi Zhou, Fei Ye, and Quanquan Gu. Structure-informed language models are protein designers. In *International conference on machine learning*, pages 42317–42338. PMLR, 2023.

[40] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pages 2790–2799. PMLR, 2019.

[41] Tiziana Sanavia, Giovanni Birolo, Ludovica Montanucci, Paola Turina, Emidio Capriotti, and Piero Fariselli. Limitations and challenges in protein stability prediction upon genome variations: towards future applications in precision medicine. *Computational and structural biotechnology journal*, 18:1968–1979, 2020.

[42] Chloe Hsu, Robert Verkuil, Jason Liu, Zeming Lin, Brian Hie, Tom Sercu, Adam Lerer, and Alexander Rives. Learning inverse folding from millions of predicted structures. In *International conference on machine learning*, pages 8946–8970. PMLR, 2022.

[43] Yunxin Xu, Di Liu, and Haipeng Gong. Improving the prediction of protein stability changes upon mutations by geometric learning and a pre-training strategy. *bioRxiv*, pages 2023–05, 2023.

[44] Dmitriy Umerenkov, Tatiana I Shashkova, Pavel V Strashnov, Fedor Nikolaev, Maria Sindeeva, Nikita V Ivanisenko, and Olga L Kardymon. Prostata: protein stability assessment using transformers. *BioRxiv*, pages 2022–12, 2022.

[45] Marina A Pak, Nikita V Dovidchenko, Satyarth Mishra Sharma, and Dmitry N Ivankov. New mega dataset combined with deep neural network makes a progress in predicting impact of mutation on protein stability. *BioRxiv*, pages 2022–12, 2023.

[46] Fabrizio Pucci, Katrien V Bernaerts, Jean Marc Kwasigroch, and Marianne Rooman. Quantification of biases in predictions of protein stability changes upon mutations. *Bioinformatics*, 34(21):3659–3665, 2018.

[47] Joost Schymkowitz, Jesper Borg, Francois Stricher, Robby Nys, Frederic Rousseau, and Luis Serrano. The foldx web server: an online force field. *Nucleic acids research*, 33(suppl_2):W382–W388, 2005.

[48] Rebecca F Alford, Andrew Leaver-Fay, Jeliazko R Jeliazkov, Matthew J O'Meara, Frank P DiMaio, Hahnbeom Park, Maxim V Shapovalov, P Douglas Renfrew, Vikram K Mulligan, Kalli Kappel, et al. The rosetta all-atom energy function for macromolecular modeling and design. *Journal of chemical theory and computation*, 13(6):3031–3048, 2017.

[49] Daniel J Diaz, Chengyue Gong, Jeffrey Ouyang-Zhang, James M Loy, Jordan Wells, David Yang, Andrew D Ellington, Alexandros G Dimakis, and Adam R Klivans. Stability oracle: a structure-based graph-transformer framework for identifying stabilizing mutations. *Nature Communications*, 15(1):6170, 2024.

[50] Emidio Capriotti, Piero Fariselli, Ivan Rossi, and Rita Casadio. A three-state prediction of single point mutations on protein stability changes. *BMC bioinformatics*, 9:1–9, 2008.

[51] Jun Cheng, Guido Novati, Joshua Pan, Clare Bycroft, Akvilė Žemgulytė, Taylor Applebaum, Alexander Pritzel, Lai Hong Wong, Michal Zielinski, Tobias Sargeant, et al. Accurate proteome-wide missense variant effect prediction with alphamissense. *Science*, 381(6664):eadg7492, 2023.

[52] Nobuhiko Tokuriki and Dan S Tawfik. Stability effects of mutations and protein evolvability. *Current opinion in structural biology*, 19(5):596–604, 2009.

[53] Andre J Faure, Júlia Domingo, Jörn M Schmiedel, Cristina Hidalgo-Carcedo, Guillaume Diss, and Ben Lehner. Mapping the energetic and allosteric landscapes of protein binding domains. *Nature*, 604(7904):175–183, 2022.

[54] Magnus Haraldson Høie, Matteo Cagiada, Anders Haagen Beck Frederiksen, Amelie Stein, and Kresten Lindorff-Larsen. Predicting and interpreting large-scale mutagenesis data using analyses of protein stability and conservation. *Cell reports*, 38(2), 2022.

[55] Xianghua Li and Ben Lehner. Biophysical ambiguities prevent accurate genetic prediction. *Nature communications*, 11(1):4923, 2020.

[56] Chenchun Weng, Andre J Faure, Albert Escobedo, and Ben Lehner. The energetic and allosteric landscape for kras inhibition. *Nature*, 626(7999):643–652, 2024.

[57] Jakub Otwinowski. Biophysical inference of epistasis and the effects of mutations on protein stability and function. *Molecular biology and evolution*, 35(10):2345–2354, 2018.

[58] Andre J Faure, Aina Martí-Aranda, Cristina Hidalgo-Carcedo, Antoni Beltran, Jörn M Schmiedel, and Ben Lehner. The genetic architecture of protein stability. *Nature*, pages 1–9, 2024.

[59] Andre J Faure and Ben Lehner. Mochi: neural networks to fit interpretable models and quantify energies, energetic couplings, epistasis and allostery from deep mutational scanning data. *bioRxiv*, pages 2024–01, 2024.

[60] Jiyao Wang, Farideh Chitsaz, Myra K Derbyshire, Noreen R Gonzales, Marc Gwadz, Shennan Lu, Gabriele H Marchler, James S Song, Narmada Thanki, Roxanne A Yamashita, et al. The conserved domain database in 2023. *Nucleic Acids Research*, 51(D1):D384–D388, 2023.

[61] Matteo Cagiada, Sandro Bottaro, Søren Lindemose, Signe M Schenstrøm, Amelie Stein, Rasmus Hartmann-Petersen, and Kresten Lindorff-Larsen. Discovering functionally important sites in proteins. *Nature communications*, 14(1):4175, 2023.

[62] PC Agu, CA Afiukwa, OU Orji, EM Ezeh, IH Ofoke, CO Ogbu, EI Ugwuja, and PM Aja. Molecular docking as a tool for the discovery of molecular targets of nutraceuticals in diseases management. *Scientific Reports*, 13(1):13398, 2023.

[63] Surojit Biswas, Grigory Khimulya, Ethan C Alley, Kevin M Esvelt, and George M Church. Low-n protein engineering with data-efficient deep learning. *Nature methods*, 18(4):389–396, 2021.

[64] Adam J Riesselman, John B Ingraham, and Debora S Marks. Deep generative models of genetic variation capture the effects of mutations. *Nature methods*, 15(10):816–822, 2018.

[65] Thomas A Hopf, John B Ingraham, Frank J Poelwijk, Charlotta PI Schärfe, Michael Springer, Chris Sander, and Debora S Marks. Mutation effects predicted from sequence co-variation. *Nature biotechnology*, 35(2):128–135, 2017.

[66] Ethan C Alley, Grigory Khimulya, Surojit Biswas, Mohammed AlQuraishi, and George M Church. Unified rational protein engineering with sequence-based deep representation learning. *Nature methods*, 16(12):1315–1322, 2019.

[67] Yijia Xiao, Jiezhong Qiu, Ziang Li, Chang-Yu Hsieh, and Jie Tang. Modeling protein using large-scale pretrain language model. *arXiv preprint arXiv:2108.07435*, 2021.

[68] Nadav Brandes, Grant Goldman, Charlotte H Wang, Chun Jimmie Ye, and Vasilis Ntranos. Genome-wide prediction of disease variant effects with a deep protein language model. *Nature Genetics*, 55(9):1512–1522, 2023.

[69] Jeffrey Ouyang-Zhang, Daniel Diaz, Adam Klivans, and Philipp Krähenbühl. Predicting a protein's stability under a million mutations. *Advances in Neural Information Processing Systems*, 36, 2024.

[70] John Ingraham, Vikas Garg, Regina Barzilay, and Tommi Jaakkola. Generative models for graph-based protein design. *Advances in neural information processing systems*, 32, 2019.

[71] Bowen Jing, Stephan Eismann, Patricia Suriana, Raphael JL Townshend, and Ron Dror. Learning from protein structure with geometric vector perceptrons. *arXiv preprint arXiv:2009.01411*, 2020.

[72] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[73] Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, et al. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3):220–235, 2023.

[74] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[75] I Loshchilov. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[76] Iván Martín Hernández, Yves Dehouck, Ugo Bastolla, José Ramón López-Blanco, and Pablo Chacón. Predicting protein stability changes upon mutation using a simple orientational potential. *Bioinformatics*, 39(1):btad011, 2023.

[77] Martin Steinegger and Johannes Söding. Mmseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nature biotechnology*, 35(11):1026–1028, 2017.

[78] Adam J Riesselman, John B Ingraham, and Debora S Marks. Deep generative models of genetic variation capture mutation effects. *arXiv preprint arXiv:1712.06527*, 2017.

Supplementary Information

A Supplementary Methods

A.1 Datasets

We used the training, validation, test splits of the Megascale dataset created by Dieckhaus et al.[35], which ensured no sequences sharing $> 25\%$ identity across splits. In addition, we collected ten independent test sets, including Fireprot (HF)[17,35], Ssym-direct[46], Ssym-inverse[46], S669[8], S783[7], S2648[14], S461[76], S8754[43], S4346[43], S571[43]. Among these, Fireprot (HF), Ssym-direct, Ssym-inverse and S669 were pre-filtered and provided by Dieckhaus et al.[35], while the remaining six datasets were collected from Xu et al.[43]. For S8754, the mutants without known wild-type structures were excluded.

We used the Megascale training and validation sets for model training, and the Megascale test set and the ten independent test sets for evaluation. Sequences in the training split with more than 25% sequence similarity to any sequences in the ten independent test sets were removed using MMseqs2[77] for similarity calculation. Only single-mutation variants were used for training and evaluation. The statistics of each dataset are provided in Table S4.

Functional site annotations for Domainome sequences were obtained from the Conserved Domain Database (CDD) by parsing the National Center for Biotechnology Information (NCBI) protein entry page (e.g., `https://www.ncbi.nlm.nih.gov/protein/UNIPROT_ID/`, where “`UNIPROT_ID`" is the protein's Uniprot ID). The functional sites for the enzyme (Uniprot ID: P00327, PDB ID: 1QLH) were downloaded from the Mechanism and Catalytic Site Atlas (M-CSA; `https://www.ebi.ac.uk/thornton-srv/m-csa/`). Experimental fitness data and baseline method predictions for the proteins in Fig. 4 were collected from the ProteinGym study[28](`https://proteingym.org/`).

A.2 Functional sites identification

For a given wild-type sequence, we used SPURS to predict $\Delta\Delta G$ and ESM1v[27] to predict the sequence likelihood change for all of its single-mutation variants. ESM's sequence likelihood change is defined as the log-likelihood difference (delta log-likelihood, ΔLL) between the mutant sequences $\boldsymbol{x}^{\text{MT}}$ compared with the wild type $\boldsymbol{x}^{\text{WT}}$:

$$s_{\text{ESM}}(x_i = x_i^{\text{MT}}|\boldsymbol{x}^{\text{WT}}) = \log p_{\text{ESM}}(x_i = x_i^{\text{MT}}|\boldsymbol{x}^{\text{WT}}) - \log p_{\text{ESM}}(x_i = x_i^{\text{WT}}|\boldsymbol{x}^{\text{WT}}), \tag{4}$$

where $p_{\text{ESM}}(x_i = x_i^{\text{MT}}|\boldsymbol{x}^{\text{WT}})$ is the ESM-predicted probability of AA x_i^{MT} occurring at position i, given the wild-type sequence as context[27]. The score $s_{\text{ESM}}(\boldsymbol{x}^{\text{MT}}|\boldsymbol{x}^{\text{WT}})$ can be interpreted as the relative evolutionary fitness of mutant $\boldsymbol{x}^{\text{MT}}$, which has been shown to be an effective zero-shot predictor of experimentally measured fitness data[27,28].

Inspired by prior studies modeling the relationship between the free energy changes due to mutations in protein folding and those in protein binding through a non-linear Boltzmann distribution[53,56–59], we applied a sigmoid function to fit the non-linear relationship between a mutant's stability change and evolutionary fitness. Specifically, we defined a stability change score $s_{\text{stab}}(x_i = x_i^{\text{MT}}|\boldsymbol{x}^{\text{WT}})$ as the negative SPURS-predicted $\Delta\Delta G$ (i.e., $-f_\theta(\cdot)$) and min-max normalized this score across all single-mutation variants, with the minimum set to the 0.1% percentile and the maximum to the 99.9% percentile. In this way, stabilizing mutations are given s_{stab} scores close to 1, and destabilizing mutations are close to 0. Next, we fit a sigmoid function on the s_{stab} and s_{ESM} scores of all single-mutation variants for each protein:

$$\hat{s}_{\text{stab}}(x_i = x_i^{\text{MT}}|\boldsymbol{x}^{\text{WT}}) = \sigma\left(\frac{s_{\text{ESM}}(x_i = x_i^{\text{MT}}|\boldsymbol{x}^{\text{WT}}) - \mu}{\tau}\right), \tag{5}$$

where $\sigma(z) = 1/(1+e^{-z})$ is the standard sigmoid function, μ and τ are the learnable parameters that control the sigmoid curve's shape, and $\hat{s}_{\text{stab}}$ is the sigmoid-fit values of s_{stab}. Following the Domainome study[36], to prioritize fitting the low-stability variants, we weighted variant by SPURS's $\Delta\Delta G$ prediction:

$$w_i^{\text{MT}} = \max(s_{\text{stab}}) - \min(s_{\text{stab}}) - s_{\text{stab}}(x_i = x_i^{\text{MT}}|\boldsymbol{x}^{\text{WT}}) \tag{6}$$

The residue of the fit is defined as:

$$\epsilon(x_i = x_i^{\text{MT}}|\boldsymbol{x}^{\text{WT}}) = s_{\text{stab}}(x_i = x_i^{\text{MT}}|\boldsymbol{x}^{\text{WT}}) - \hat{s}_{\text{stab}}(x_i = x_i^{\text{MT}}|\boldsymbol{x}^{\text{WT}}). \tag{7}$$

A recent study showed that the residuals indicate mutations with larger or smaller effects on fitness than that can be counted for by stability changes[36]. We thus define a per-site importance score as the average residuals across the 20 AA mutations at a site, referred to as *function score*:

$$s_{\text{func}}(i) = \frac{1}{20}\sum_{a\in\Sigma}\epsilon(x_i = a|\boldsymbol{x}^{\text{WT}}). \tag{8}$$

As shown in the previous study[36] and confirmed by our results (Fig. 3), a larger $s_{\text{func}}(i)$ value suggests that site i is more likely a functional site.

A.3 Enhanced protein fitness prediction with SPURS

To demonstrate that SPURS can enhance supervised protein fitness prediction, we used the "Augmented models," one of the leading low-N fitness prediction methods, developed by Hsu et al.[26] as our base model. To predict the quantitative fitness value of a variant, Augmented models represent the input sequence as a concatenation of one-hot encoding of its amino acids in the sequence, resulting in a flattened vector with $L \times |\mathcal{A}|$ dimensions, where $\mathcal{A}$ is the set of possible amino acids. This vector is augmented to a $(L \times |\mathcal{A}| + 1)$-dimension feature vector by incorporating the delta likelihood score predicted by a protein sequence generative model, such as a pLM (e.g., s_{ESM} in Eq. 4) or other sequence likelihood models (e.g., EVmutation[78], Evotuned UniRep[66], DeepSequence[64]). The augmented vector is then used as the input feature to train a Ridge regression model to predict the experimental fitness value.

To enhance this approach, we incorporated SPURS's $\Delta\Delta G$ prediction for input protein into the feature vector, leading to a $(L \times |\mathcal{A}| + 2)$-dimension vector for the Ridge regression. This modified model is referred to as SPURS-augmented model. For each DMS dataset, we first set aside 20% randomly sampled mutants as the test set. From the remaining data, we sampled N mutants as the training set, where N was set to 48, 96, 144, 192, 240, or $N = 80\%$ of total sequences in a DMS dataset, following the setup from Hsu et al.[26]. For each DMS dataset, we performed 20 repetitions of random training data sampling and model training with different random seeds, reporting the average prediction performance as the final score.

A.4 Improved ProteinMPNN decoding for structural feature learning

The original ProteinMPNN was designed as an inverse folding model to generate sequences compatible with a given backbone structure. It employed an autoregressive (sequential) decoding scheme to generate each amino acid one at a time, conditioning on the input structure and previously decoded amino acids: $p(x_i|\boldsymbol{x}_{<i};\mathcal{S})$ where $\boldsymbol{x}_{<i} = x_1 \dots x_{i-1}$. While this design aligns with ProteinMPNN's initial purpose, it inherently limits its ability to fully learn structural features for stability prediction in our work, as the model cannot access information from the entire sequence during decoding. To overcome this limitation, we adopted a one-shot decoding strategy, allowing each position in the sequence to access information from all other amino acids simultaneously: $p(x_i|\boldsymbol{x}_{-i};\mathcal{S})$ where $\boldsymbol{x}_{-i}$ refers to all amino acids except at position i, thereby providing richer sequence context and improving structure representation learning. To fully leverage the advantages of this new strategy, we fine-tuned the ProteinMPNN model weights during SPURS's training, rather than keeping them fixed, allowing the model to better adapt to the refined decoding scheme and enhancing its representation learning effectiveness.

A.5 Data availability

We used *esm2_t33_650M_UR50D* checkpoint of ESM2, *v_48_020* checkpoint of ProteinMPNN, and *esm1v_t33_650M_UR90S_1* checkpoint of ESM1v (`https://dl.fbaipublicfiles.com/fair-esm/models/esm1v_t33_650M_UR90S_1.pt`). The Megascale dataset was downloaded from `https://zenodo.org/records/7844779`. Megascale split, Fireprot(HF), S669, Ssym-direct, and Ssym-inverse were downloaded from `https://github.com/Kuhlman-Lab/ThermoMPNN`. S783, S2648, S461, S8754, S4346 and S571 were downloaded from `https://github.com/Gonglab-THU/GeoStab`. Domainome dataset and the performance of baseline models were downloaded from `https://zenodo.org/records/11260616`.

B Supplementary Tables

Dataset	ESM	ProteinMPNN	FT-ESM	FT-ProteinMPNN	ThermoMPNN	SPURS
Megascale	0.33	0.41	0.64	0.70	0.72	**0.74**
Fireprot(HF)	0.43	0.55	0.48	0.64	0.64	**0.67**
Ssym-direct	0.36	0.58	0.43	0.71	0.71	**0.76**
Ssym-inverse	0.38	0.38	0.28	0.63	0.59	**0.68**
S669	0.37	0.43	0.46	0.46	0.44	**0.50**
S461	0.29	0.50	0.65	0.60	0.61	**0.66**
S783	0.30	0.52	0.59	0.69	0.70	**0.71**
S8754	0.16	0.24	0.53	0.60	0.61	**0.64**
S2648	0.18	0.31	0.53	0.65	0.65	**0.69**
S571	0.27	0.43	0.40	0.42	0.39	**0.47**
S4346	0.32	0.52	0.46	0.61	0.60	**0.64**

Table S1: Spearman correlation results for SPURS and baseline models across eleven test sets. The correlations are computed using all mutations within each dataset collectively. This is the raw data underlying Fig. 2.

Dataset	Stability Oracle	PROSTATA	RASP	FoldX	Rosetta	ThermoNet
Megascale	-	0.59[35]	0.67[35]	0.57[35]	0.56[35]	0.33[35]
Fireprot (HF)	-	0.55[35]	0.44[35]	0.57[35]	0.52[35]	0.36[35]
Sym-direct	0.68[49]	-	-	-	-	-
Sym-inverse	0.68[49]	-	-	-	-	-
S669	0.50[49]	-	-	-	-	-

Table S2: Spearman correlation results for additional baseline models across test sets. Performance as reported by Diaz et al.[49] and Dieckhaus et al.[35] of additional baseline models on test sets. This is the raw data underlying Fig. 2.

Abbreviation	ProteinGym Name	Abbreviation	ProteinGym Name
A0A1I9GEU1	A0A1I9GEU1_NEIME_Kennouche_2019	A0A192B1T2	A0A192B1T2_9HIV1_Haddox_2018
A0A247D711	A0A247D711_LISMN_Stadelmann_2021	A0A2Z5U3Z0	A0A2Z5U3Z0_9INFA_Doud_2016
A0A2Z5U3Z02	A0A2Z5U3Z0_9INFA_Wu_2014	A4D664	A4D664_9INFA_Soh_2019
A4GRB6	A4GRB6_PSEAI_Chen_2020	AACC1_PSEAI	AACC1_PSEAI_Dandage_2018
ACE2_HUMAN	ACE2_HUMAN_Chan_2020	ADRB2_HUMAN	ADRB2_HUMAN_Jones_2020
ANCSZ_Hobbs	ANCSZ_Hobbs_2022	AMIE_PSEAE	AMIE_PSEAE_Wrenbeck_2017
BRCA1_HUMAN	BRCA1_HUMAN_Findlay_2018	CALM1_HUMAN	CALM1_HUMAN_Weile_2017
C6KNH7	C6KNH7_9INFA_Lee_2018	CAS9_STRP1	CAS9_STRP1_Spencer_2017_positive
CASP3_HUMAN	CASP3_HUMAN_Roychowdhury_2020	CASP7_HUMAN	CASP7_HUMAN_Roychowdhury_2020
CBS_HUMAN	CBS_HUMAN_Sun_2020	CD19_HUMAN	CD19_HUMAN_Klesmith_2019_FMC_singles
CCDB_ECOLI	CCDB_ECOLI_Adkar_2012	CCDB_ECOLI2	CCDB_ECOLI_Tripathi_2016
CCR5_HUMAN	CCR5_HUMAN_Gill_2023	CP2C9_HUMAN	CP2C9_HUMAN_Amorosi_2021_activity
CP2C9_HUMAN2	CP2C9_HUMAN_Amorosi_2021_abundance	D7PM05	D7PM05_CLYGR_Somermeyer_2022
DYR_ECOLI	DYR_ECOLI_Thompson_2019	DYR_ECOLI2	DYR_ECOLI_Nguyen_2023
ERBB2_HUMAN	ERBB2_HUMAN_Elazar_2016	ENV_HV1B9	ENV_HV1B9_DuenasDecamp_2016
ENV_HV1BR	ENV_HV1BR_Haddox_2016	ENVZ_ECOLI	ENVZ_ECOLI_Ghose_2023
GFP_AEQVI	GFP_AEQVI_Sarkisyan_2016	GAL4_YEAST	GAL4_YEAST_Kitzman_2015
GDIA_HUMAN	GDIA_HUMAN_Silverstein_2021	HEM3_HUMAN	HEM3_HUMAN_Loggerenberg_2023
HMDH_HUMAN	HMDH_HUMAN_Jiang_2019	HXK4_HUMAN	HXK4_HUMAN_Gersing_2022_activity
HXK4_HUMAN2	HXK4_HUMAN_Gersing_2023_abundance	HSP82_YEAST	HSP82_YEAST_Mishra_2016
HSP82_YEAST2	HSP82_YEAST_Cote-Hammarlof_2020_growth-H2O2	HSP82_YEAST3	HSP82_YEAST_Flynn_2019
IF1_ECOLI	IF1_ECOLI_Kelsic_2016	I6TAH8	I6TAH8_I68A0_Doud_2015
KCNJ2_MOUSE	KCNJ2_MOUSE_Coyote-Maestas_2022_surface	KCNJ2_MOUSE2	KCNJ2_MOUSE_Coyote-Maestas_2022_function
KCNE1_HUMAN	KCNE1_HUMAN_Muhammad_2023_function	KCNE1_HUMAN2	KCNE1_HUMAN_Muhammad_2023_expression
KKA2_KLEPN	KKA2_KLEPN_Melnikov_2014	LGK_LIPST	LGK_LIPST_Klesmith_2015
LYAM1_HUMAN	LYAM1_HUMAN_Elazar_2016	MK01_HUMAN	MK01_HUMAN_Brenan_2016
MET_HUMAN	MET_HUMAN_Estevam_2023	MSH2_HUMAN	MSH2_HUMAN_Jia_2020
MTH3_HAEAE	MTH3_HAEAE_RockahShmuel_2015	MTHR_HUMAN	MTHR_HUMAN_Weile_2021
MLAC_ECOLI	MLAC_ECOLI_MacRae_2023	NPC1_HUMAN	NPC1_HUMAN_Erwood_2022_HEK293T
P53_HUMAN	P53_HUMAN_Kotler_2018	P53_HUMAN2	P53_HUMAN_Giacomelli_2018_WT_Nutlin
P53_HUMAN3	P53_HUMAN_Giacomelli_2018_Null_Etoposide	P53_HUMAN4	P53_HUMAN_Giacomelli_2018_Null_Nutlin
PABP_YEAST	PABP_YEAST_Melamed_2013	PA_I34A1	PA_I34A1_Wu_2015
PAI1_HUMAN	PAI1_HUMAN_Huttinger_2021	PPARG_HUMAN	PPARG_HUMAN_Majithia_2016
PPM1D_HUMAN	PPM1D_HUMAN_Miller_2022	PRKN_HUMAN	PRKN_HUMAN_Clausen_2023
POLG_DEN26	POLG_DEN26_Suphatrakul_2023	Q2N0S5	Q2N0S5_9HIV1_Haddox_2018
Q53Z42	Q53Z42_HUMAN_McShan_2019_expression	Q53Z422	Q53Z42_HUMAN_McShan_2019_binding-TAPBPR
Q6WV13	Q6WV13_9MAXI_Somermeyer_2022	Q59976	Q59976_STRSQ_Romero_2015
Q837P4	Q837P4_ENTFA_Meier_2023	Q837P5	Q837P5_ENTFA_Meier_2023
Q8WTC7	Q8WTC7_9CNID_Somermeyer_2022	R1AB_SARS2	R1AB_SARS2_Flynn_2022
RASK_HUMAN	RASK_HUMAN_Weng_2022_binding-DARPin_K55	RASK_HUMAN2	RASK_HUMAN_Weng_2022_abundance
REV_HV1H2	REV_HV1H2_Fernandes_2016	RDRP_I33A0	RDRP_I33A0_Li_2023
RNC_ECOLI	RNC_ECOLI_Weeks_2023	RL40A_YEAST	RL40A_YEAST_Roscoe_2013
RL40A_YEAST2	RL40A_YEAST_Roscoe_2014	RL40A_YEAST3	RL40A_YEAST_Mavor_2016
S22A1_HUMAN	S22A1_HUMAN_Yee_2023_abundance	S22A1_HUMAN2	S22A1_HUMAN_Yee_2023_activity
SC6A4_HUMAN	SC6A4_HUMAN_Young_2021	SPG1_STRSG	SPG1_STRSG_Olson_2014
SRC_HUMAN	SRC_HUMAN_Ahler_2019	SRC_HUMAN2	SRC_HUMAN_Chakraborty_2023_binding-DAS_25uM
SRC_HUMAN3	SRC_HUMAN_Nguyen_2022	SERC_HUMAN	SERC_HUMAN_Xie_2023
SPIKE_SARS2	SPIKE_SARS2_Starr_2020_expression	SPIKE_SARS22	SPIKE_SARS2_Starr_2020_binding
SUMO1_HUMAN	SUMO1_HUMAN_Weile_2017	TADBP_HUMAN	TADBP_HUMAN_Bolognesi_2019
TAT_HV1BR	TAT_HV1BR_Fernandes_2016	TPK1_HUMAN	TPK1_HUMAN_Weile_2017
TPMT_HUMAN	TPMT_HUMAN_Matreyek_2018	TPOR_HUMAN	TPOR_HUMAN_Bridgford_2020
TRPC_SACS2	TRPC_SACS2_Chan_2017	TRPC_THEMA	TRPC_THEMA_Chan_2017
UBE4B_MOUSE	UBE4B_MOUSE_Starita_2013	UBC9_HUMAN	UBC9_HUMAN_Weile_2017
VKOR1_HUMAN	VKOR1_HUMAN_Chiasson_2020_abundance	VKOR1_HUMAN2	VKOR1_HUMAN_Chiasson_2020_activity
YAP1_HUMAN	YAP1_HUMAN_Araya_2012		

Table S3: The abbreviations for DMS dataset names in the ProteinGym substitution benchmark dataset. For formatting purposes, the DMS dataset names used in the ProteinGym dataset are abbreviated in Fig. 4c. Digit suffixes are used to further distinguish between different DMS studies targeting the same protein.

Dataset	Number of mutants	Number of proteins
Megascale (training)	216,919	239
Filtered Megascale (training)	203,789	212
Megascale (validation)	27,481	31
Filtered Megascale (validation)	26,548	29
Megascale (test)	28,312	28
Fireprot (HF)	3,438	89
Ssym-direct	342	15
Ssym-inverse	342	342
S669	669	94
S461	461	48
S783	783	55
S8754	8,236	274
S2648	2,648	131
S4346	3,988	281
S571	571	37
Domainome	536,164	522

Table S4: Statistics of datasets used for model training and evaluation. The number of unique proteins and total number of mutants across all proteins in each dataset.