SELF-SUPERVISED LEARNING OF EPISTATIC MUTATION EFFECTS IN STRUCTURE-BASED MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Predicting the impact of multiple point mutations on a given protein sequence remains a central challenge in protein engineering, in large part due to the nonlinear effects of epistasis. Experimental data sets that measure multiple mutations are scarce, motivating methods that learn from structural and evolutionary data. We introduce *EvoEpi*, a self-supervised framework for scoring double mutations from protein sequences and backbone structures without using phenotype labels in the training objective. *EvoEpi* learns from joint amino-acid frequencies in multiple sequence alignments (MSAs), retaining dependencies between residue pairs that single-position supervision discards. More specifically, our structure-based model predicts changes in log-frequency ratios resulting from double mutations. Trained on the single-chain subset of the ProteinMPNN training set and their corresponding MSAs, *EvoEpi* achieves global/mean per-protein Spearman correlations of $0.50/0.50$, $0.42/0.55$, and $0.62/0.51$ on ThermoMutDB-D, Megascale-D, and PTMul-D, respectively, outperforming all evaluated self-supervised baselines and reaching supervised-level performance, with higher correlations than all evaluated supervised predictors on ThermoMutDB-D and PTMul-D.

1 INTRODUCTION

Protein engineering enables the development of new therapeutics (Gebauer & Skerra, 2020; Meghwanshi et al., 2020; Listov et al., 2024), more efficient industrial biocatalysts (Bell et al., 2021; Wu et al., 2021), and a deeper understanding of how genetic variation affects human health (Stein et al., 2019). Engineering a protein requires identifying mutations that improve desired properties, or phenotypes, while preserving other essential properties. Advances in structure prediction and sequence design (Abramson et al., 2024; Zhang et al., 2026; Dauparas et al., 2022) have expanded the possibilities for in silico protein design. However, improving the phenotypes that limit an existing protein's commercial use often requires combining multiple mutations whose joint effects are difficult to predict.

The phenomena underpinning this difficulty is *epistasis*: the effect of one mutation can depend on the presence of another, causing their combined effect to differ from the sum of their individual effects (Figure 1). Consequently, combining promising single substitutions does not necessarily improve performance beyond that of the individual variants. Identifying effective combinations can therefore require laborious experimental screening across a large combinatorial space. Double substitutions provides the simplest setting for studying this challenge.

For protein stability prediction, supervised frameworks designed to model epistasis (Ouyang-Zhang et al., 2023; Dieckhaus & Kuhlman, 2025; Li & Luo, 2025) are fine-tuned on changes in Gibbs free energy ($\Delta\Delta G$) measured for double mutants in high-throughput assays (Tsuboyama et al., 2023). However, these labeled datasets sample relatively few protein families and residue combinations, leaving much of the diversity represented in larger sequence and structural databases unexplored (Consortium, 2019; Varadi et al., 2022). Furthermore, their predictions for double substitutions often achieve correlations similar to, or lower than, those of additive baselines that sum the predicted effects of the constituent single substitutions. Together, limited experimental coverage and inconsistent gains over additive predictions motivate the development of self-supervised learning (SSL) objectives that use existing protein sequence databases to learn features relevant to epistasis without requiring experimental phenotype labels.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

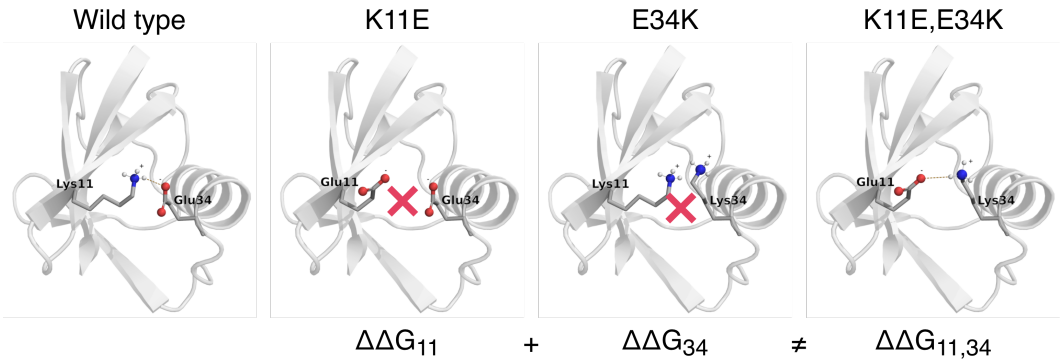


Figure 1: *Epistasis* describes nonlinear interactions between mutations, where the combined effect of multiple substitutions differs from the sum of their individual effects. Here, the thermodynamic stability change induced by the double substitution K11E/E34K differs from the sum of the effects of K11E and E34K individually. (PDB: 1UBQ)

Models trained on protein sequences and structures without stability labels (Meier et al., 2021; Hsu et al., 2022; Dauparas et al., 2022; Shuai et al., 2025) have shown an ability to learn features of protein energy landscapes (Dutton et al., 2024; Cagiada et al., 2025; Frellsen et al., 2026), enabling zero-shot stability prediction. Yet a protein backbone is compatible with many sequences, and the wild-type (WT) sequence is only one example. Objectives that reconstruct the WT sequence (Dauparas et al., 2022) or optimize its pseudo-likelihood conditioned on structure (Li et al., 2023; Shuai et al., 2025; Dieckhaus & Kuhlman, 2025) do not explicitly use the evolutionary variation captured by MSAs (Thompson et al., 1994; 1997). By capturing variation across homologous sequences, MSAs provide learning signals beyond those available from an individual sequence (Hopf et al., 2017; Rao et al., 2021; Riesselman et al., 2018; Abramson et al., 2024; Gong et al., 2024). In particular, dependencies between residues in these alignments offer a natural source of supervision for learning interactions relevant to epistasis.

We introduce *EvoEpi*, an SSL framework[1] that learns to score double substitutions from protein sequences and backbone structures using pairwise evolutionary statistics as supervision. Specifically, we train on joint amino-acid frequencies from MSAs to learn features relevant to protein fitness. Whereas objectives based on per-position MSA frequencies (Gong et al., 2024) summarize each position independently, joint frequencies preserve statistical dependencies between pairs of positions that may inform second-order mutational effects. We further introduce ranking-based training objectives that focus learning on energy differences associated with double substitutions rather than absolute protein energies. During training, we sample amino-acid replacements at randomly selected positions from an inverse-folding prior, exposing the scorer to varied sequence contexts on the same backbone. At inference, *EvoEpi* uses the original sequence and its corresponding backbone, together with embeddings of the amino-acid pairs before and after substitution, to score a proposed double substitution without requiring an MSA.

We evaluate *EvoEpi* on three curated benchmarks of experimentally measured $\Delta\Delta G$ for double substitutions: PTMul-D, derived from ProTherm (Nikam et al., 2021), Megascale-D (Tsuboyama et al., 2023), and ThermoMutDB-D (Xavier et al., 2021). On their single-chain subsets, *EvoEpi* achieves global/mean per-protein Spearman correlations of $0.62/0.51$, $0.42/0.55$, and $0.50/0.50$ on PTMul-D, Megascale-D, and ThermoMutDB-D, respectively. In mean per-protein Spearman, *EvoEpi* improves over the strongest evaluated self-supervised baseline by 0.15, 0.12, and 0.16 on the three benchmarks, respectively. It reaches supervised-level performance overall and, on PTMul-D and ThermoMutDB-D, exceeds the strongest evaluated supervised predictor by 0.03–0.06 in global Spearman and 0.07–0.10 in mean per-protein Spearman. These within-protein gains are particularly relevant to protein engineering, where the practical goal is often to rank candidate mutations for a fixed target protein.

[1]Here, we use SSL in the broad sense of model training without experimental phenotype labels.

2 BACKGROUND

2.1 PROTEIN ENGINEERING

Protein engineering seeks a protein sequence $\mathbf{s}$ that maximizes a target phenotypic property $f(\mathbf{s})$, such as thermodynamic stability, binding affinity, fluorescence brightness, antibiotic resistance, expression, or activity. Because the sequence space is vast, practical workflows often search for beneficial local mutations around an initial sequence $\mathbf{s}^0$. A conventional approach is directed evolution (DE) (Arnold, 1998; Giver et al., 1998): starting from $\mathbf{s}^i$, DE constructs a library of nearby sequence variants and screens them experimentally to measure their fitness. The best-performing variant becomes $\mathbf{s}^{i+1}$, the starting sequence for the next round. This cycle continues until the target fitness is reached. Although multiplexed assays of variant effects (MAVE, Starita et al. (2017)), particularly deep mutational scanning (DMS, Fowler & Fields (2014)), have enabled parallel measurements of mutation effects, there remains strong demand for inexpensive *in silico* predictions of their functional consequences without requiring wet-lab experiments.

Protein Stability Assessment. Predicting changes in protein stability has been particularly successful, with model predictions showing high correlations with stability measured *in vitro*. Inverse-folding models have attracted interest because they offer a way to approximate stability changes using predicted log-likelihood ratios:

$$\Delta\Delta G \approx \log \frac{p_\theta(\text{Mutant sequence} \mid \text{WT structure})}{p_\theta(\text{WT sequence} \mid \text{WT structure})}, \tag{1}$$

where the conditional likelihood $p_\theta(\mathbf{s} \mid \mathbf{x})$ is evaluated directly using the inverse-folding model. This relationship can be motivated by a Boltzmann assumption (Frellsen et al., 2026). Inverse-folding methods based on Potts models (Shuai et al., 2025; Birnbaum & Keating, 2026) explicitly parameterize an energy landscape that serves as a proxy for free energy. Another advantage of inverse-folding models is that their training does not require stability labels. Nevertheless, inverse-folding models (Dauparas et al., 2022; Shuai et al., 2025; Birnbaum & Keating, 2026) remain less competitive than supervised stability predictors (Ouyang-Zhang et al., 2023; Dieckhaus & Kuhlman, 2025). This gap widens for higher-order mutations, reflecting the sparse representation of epistatic interactions in the paired wild-type (WT) sequence and structure datasets used to train self-supervised models. These limitations motivate richer self-supervised learning signals for modeling epistatic interactions that are sparse in paired WT sequence–structure data.

2.2 MULTIPLE SEQUENCE ALIGNMENTS

An alternative source of knowledge that biologists use besides from structure is evolution sequences. For a query protein, multiple sequence alignments (MSAs) capture genetic variation among homologous sequences retrieved from large databases, such as UniProt (Consortium, 2019), BFD (Steinegger et al., 2019), OMG (Cornman et al., 2024), and Logan (Chikhi et al., 2025). Despite these genetic variations, these sequences adopt the same global fold, making it a good candidate for data augmentation in training structure-based models. A sufficiently deep MSA captures evolutionary signals, structural constraints, and functional properties of a protein family (Thompson et al., 1994; 1997). MSAs have therefore long played an important role in computational biology (Morcos et al., 2011) and remain essential to many modern AI tools for *in silico* protein design. MSA is useful to resolve co-evolutionary information for structure (Benner & Gerloff, 1991; Balakrishnan et al., 2011). Recently, state-of-the-art structure prediction and design models (Abramson et al., 2024; Passaro et al., 2025; Zhang et al., 2026) use MSA-derived input features to improve prediction quality. However, compressing an MSA into a position-specific scoring matrix (PSSM) to construct a training loss (Gong et al., 2024) removes information about residue combinations within each row, discarding evidence of epistatic interactions. In this work, we use compact pairwise MSA statistics solely to construct the training loss while retaining these dependencies. This pairwise supervision helps address the sparsity of epistatic information in $(\mathbf{s}, \mathbf{x})$ datasets used for SSL.

3 METHODS

This section presents the training procedure and model architecture for scoring double mutations **without** stability labels, which require laborious wet-lab experiments to obtain. We instead use

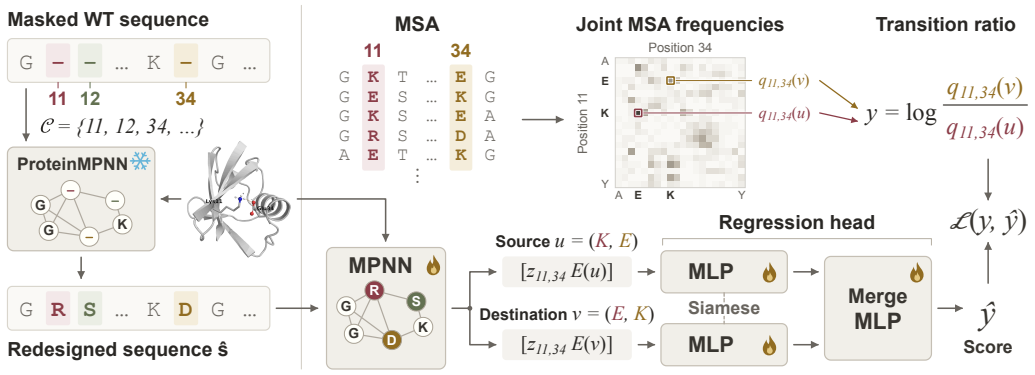


Figure 2: Overview of *EvoEpi*'s training pipeline.

structural, sequence, and evolutionary data, which are increasingly accessible with advances in co-folding models. Using these data, we train a model to predict log-frequency ratios between amino-acid pair states observed in MSAs.

Problem Setup. For a protein with L residues, let $\mathbf{s} = (s_1, \ldots, s_L)$ denote its sequence, with each residue s_i drawn from the alphabet of 20 canonical amino acids, $\mathcal{V} := \{A, C, \ldots, Y\}$. Let $\mathbf{x} \in \mathbb{R}^{L\times 4\times 3}$ contain the Cartesian coordinates of the backbone atoms: N, C_α, C, and O. We hold these coordinates fixed when scoring mutations, assuming that the backbone structure is conserved. For two positions $i < j$, let $\mathbf{u} = (u_i, u_j) \in \mathcal{V}^2$ and $\mathbf{v} = (v_i, v_j) \in \mathcal{V}^2$ denote the source and destination amino-acid pair states. Note that the source and destination sequences agree outside positions i and j. Our goal is to predict the energy change associated with the transition $\mathbf{u} \to \mathbf{v}$, which may involve a substitution at one or both positions. The source pair need not match the WT residues during training. At inference, we set $\mathbf{u} = (s_i, s_j)$ and let $\mathbf{v}$ denote the mutant pair.

During training, we also have access to an MSA $\mathcal{M}(\mathbf{s})$, whose sequences are assumed to share the same fold and function as the WT sequence $\mathbf{s}$. For each training example, we sample a subset of positions $\mathcal{C} \subset \{1, \ldots, L\}$ with $|\mathcal{C}| = K$. The same set $\mathcal{C}$ is used both to construct redesigned sequence contexts and to define the residue pairs supervised by the evolutionary objective.

3.1 Pairwise transition scoring model

Generative Input sequence augmentation. Prior works in vision (He et al., 2022; Azizi et al., 2023) and language (Yoo et al., 2021; Meng et al., 2022) have demonstrated that synthetic examples generated by pretrained generative models can augment labeled training data and improve the performance of downstream discriminative models, particularly in data-scarce regimes. Motivated by this idea, rather than relying solely on the WT sequence during training, we adopt a frozen generative prior p_ϕ (Dauparas et al., 2022) to synthesize a partially redesigned sequence for each training chain. Let $\mathbf{s}^{\setminus\mathcal{C}}$ denote the WT sequence with positions in $\mathcal{C}$ masked. We generate $\hat{\mathbf{s}}$ by sampling replacement amino acids at these positions in parallel, while leaving the remaining residues unchanged:

$$\hat{s}_i \sim \begin{cases} p_\phi(\hat{s}_i \mid \mathbf{s}^{\setminus\mathcal{C}}, \mathbf{x}), & \text{if } i \in \mathcal{C} \\ \delta(\hat{s}_i = s_i), & \text{otherwise,} \end{cases} \tag{2}$$

where $\delta(\cdot)$ is the delta function. This procedure enriches the training distribution by exposing the model to sequences beyond the WT sequences while retaining the same backbone structure used for supervision. Relying solely on the WT sequence provides only a single sequence for each structure, whereas independently sampling residues from a position-specific scoring matrix (PSSM) may produce implausible sequence combinations. In contrast, sampling from the generative prior produces alternative sequence contexts that better respect sequence–structure compatibility while increasing contextual diversity. See Appendix C.2 for comparison with the alternatives. Note that the sequence redesign is used only during training; at inference, the model receives the unmodified WT sequence.

Input features and architecture. Conditioned on the $(\mathbf{s}, \mathbf{x})$ pair, we adopt an MPNN encoder–decoder architecture as the model backbone. During training, the sequence context is the redesigned sequence $\hat{\mathbf{s}}$, while inference uses the WT sequence $\mathbf{s}$. The MPNN decoder produces residue features

$(\mathbf{h}_l)_{l=1}^{L}$, with $\mathbf{h}_l \in \mathbb{R}^{d_h}$. For each selected residue pair (i, j), we concatenate the endpoint representations to obtain $\mathbf{z}_{ij} = [\mathbf{h}_i, \mathbf{h}_j]$. A learnable embedding map $E : \mathcal{V}^2 \to \mathbb{R}^{d_e}$ assigns an embedding to each of the 20^2 amino-acid pair states. As illustrated in Figure 2, we separately concatenate $\mathbf{z}_{ij}$ with the source and destination pair embeddings, $E(\mathbf{u})$ and $E(\mathbf{v})$, and process the resulting vectors with a shared Siamese MLP. A final merging MLP combines the two contextualized pair representations to predict the transition score $f_{\theta,ij}(\mathbf{u}, \mathbf{v}; \hat{\mathbf{s}}, \mathbf{x})$ during training.

3.2 Supervising double mutation feasibility from the MSA

Constructing joint evolutionary distributions. Having defined the pairwise transition scorer, we construct its training targets from pairwise evolutionary statistics in the MSA. For positions $i < j$, let $\mathcal{M}_{ij}(\mathbf{s})$ denote the alignment rows with canonical amino acids at both positions, excluding entries affected by insertions or deletions. Formally, the empirical joint distribution over $\mathcal{V}^2$ is

$$q_{ij}(r, t; \mathbf{s}) = \frac{\sum_{\tilde{\mathbf{s}} \in \mathcal{M}_{ij}(\mathbf{s})} \delta(\tilde{s}_i - r)\delta(\tilde{s}_j - t)}{|\mathcal{M}_{ij}(\mathbf{s})|}, \tag{3}$$

where $r, t \in \mathcal{V}$ are amino-acid categories. Figure 2 illustrates the joint MSA frequencies corresponding to $q_{11,34}(r, t; \mathbf{s})$. Rather than supervising all $\binom{L}{2}$ residue pairs, we supervise the $\binom{K}{2}$ unordered pairs formed by the sampled positions in $\mathcal{C}$.

Predicting pairwise transition ratios. Following the relative-likelihood approach of Gong et al. (2024), we predict log-frequency ratios under the empirical MSA distribution rather than fitting the full pairwise distribution with a cross-entropy objective. For a transition $\mathbf{u} \to \mathbf{v}$, the evolutionary target is

$$y_{ij}(\mathbf{u} \to \mathbf{v}; \mathbf{s}) := \log \frac{q_{ij}(\mathbf{v}; \mathbf{s})}{q_{ij}(\mathbf{u}; \mathbf{s})}. \tag{4}$$

We supervise the transition scorer using

$$\mathcal{L}(\theta; \mathbf{s}, \mathbf{x}) = \sum_{\substack{i,j \in \mathcal{C} \\ i<j}} \mathbb{E}_{\mathbf{u},\mathbf{v}} \left[H_\delta \left(f_{\theta,ij}(\mathbf{u}, \mathbf{v}; \hat{\mathbf{s}}, \mathbf{x}) - y_{ij}(\mathbf{u} \to \mathbf{v}; \mathbf{s}) \right) \right], \tag{5}$$

where H_δ is the Huber loss with threshold δ, and the expectation is over sampled source and destination pair states. Figure 2 illustrates how the transition target is constructed from the joint MSA distribution. See Appendix A.2 for details.

Directly predicting the ratio avoids requiring the model to reconstruct and normalize the entire pairwise distribution when only relative preferences between sequence states are needed. This formulation is also well matched to protein-engineering phenotypes such as $\Delta\Delta G$ and ΔT_m, which quantify the change induced by a mutation relative to a reference sequence rather than an absolute property of the full sequence space. Conceptually, this parallels the discrete score-modeling view of SEDD (Lou et al., 2023), which directly parameterizes the probability ratios needed by the reverse process. Our setting differs in an important respect: rather than learning these ratios implicitly through a score-matching objective, we directly supervise them using empirical frequency ratios obtained from the MSA.

Sampling transition pairs. To improve training efficiency, we Monte Carlo sample transitions rather than enumerate all possible pair-state transitions. For each selected residue pair (i, j), we sample 64 transitions $(\mathbf{u}, \mathbf{v})$ uniformly with replacement. At position i, amino acids are sampled from categories with nonzero empirical marginal frequency, $\sum_{t \in \mathcal{V}} q_{ij}(r, t; \mathbf{s}) > 0$, and analogously at position j from categories satisfying $\sum_{r \in \mathcal{V}} q_{ij}(r, t; \mathbf{s}) > 0$. The sampled transitions are evaluated in a single batch, avoiding enumeration of all 20^4 possible $(\mathbf{u}, \mathbf{v})$ transitions.

Importantly, although each individual amino acid must be observed at its respective position, their joint pair state need not have been observed in the MSA. Thus, a sampled state $\mathbf{u} = (u_i, u_j)$ may satisfy $q_{ij}(\mathbf{u}; \mathbf{s}) = 0$ even when both u_i and u_j have positive marginal frequencies. This exposes the model to combinations whose plausibility cannot be inferred from single-position frequencies alone and therefore encourages learning pairwise dependencies. Although each training step supervises only a subset of residue pairs and transitions, the dense evolutionary information available across MSA sequences provides a rich supervision signal at substantially lower computational cost.

3.3 Reference calibration at inference time

Following the IDP-reference approach of Frellsen et al. (2026) for zero-shot stability prediction, we construct a global 20×20 reference frequency table R from residue pairs occurring within the same intrinsically disordered region (IDR), using annotations from MobiDB (Piovesan et al., 2021). We aggregate pair counts across these IDRs to estimate a background distribution over the 20^2 amino-acid pair states. For a WT pair $\mathbf{u}$ and mutant pair $\mathbf{v}$, we calibrate the predicted transition score by subtracting the corresponding background log-frequency ratio, yielding $f_{\theta,ij}(\mathbf{u}, \mathbf{v}; \mathbf{s}, \mathbf{x}) - \log \frac{R(\mathbf{v})}{R(\mathbf{u})}$. This correction accounts for background amino-acid pair preferences in disordered sequences. We refer the reader to Frellsen et al. (2026) for the theoretical motivation for IDP-based reference calibration.

4 Experiments

In this section, we evaluate *EvoEpi* through a series of experiments designed to answer three questions:

Q1. *Which evaluation metrics best reflect performance in practical protein engineering?*

Q2. *How accurately does EvoEpi score double mutations, and does joint scoring improve over an additive treatment of the two substitutions?*

Q3. *Does the learned representation generalize beyond thermodynamic stability to other protein phenotypes?*

Together, these experiments evaluate not only the predictive performance of *EvoEpi*, but also whether its pairwise evolutionary supervision captures information that is useful across proteins, mutations, and phenotypes.

4.1 Training Details

Training data and MSA collection. We derive our training set from the clustered ProteinMPNN dataset (Dauparas et al., 2022), retaining its 30% sequence-identity cluster splits and associated structures while treating individual protein chains as separate training examples. ProteinMPNN was originally trained on biological assemblies, whereas we restrict our study to individual chains as a proof of concept for MSA-based supervision. Extending our approach to multi-chain assemblies using paired MSAs is left for future work. After filtering, the training set contains 438,507 chain–structure entries corresponding to 97,070 unique amino-acid sequences. Because the same sequence can occur in multiple chains or structural conformations, a single sequence is associated with ~ 4.5 structural examples on average.

We retrieve homologous sequences from UniRef90 (Suzek et al., 2007) using MMseqs2 (Steinegger & Söding, 2017) to construct MSAs, with a maximum MSA depth of 10,000 sequences. After removing A3M insertions, we precompute residue-pair counts for each alignment. For each pair of positions, joint amino-acid frequencies are computed using only alignment rows containing canonical residues at both positions. Rows containing a gap or noncanonical residue at either position are excluded. Consequently, the joint frequencies for a sequence of length L yield $\binom{L}{2}$ tables with a shape of 20×20. See Appendix A for more details.

4.2 Results

Benchmarks and baselines. Our evaluation covers $\Delta\Delta G$ prediction on ThermoMutDB (Xavier et al., 2021), Megascale (Tsuboyama et al., 2023) test split, and PTMul-D, a double-mutation subset of ProThermDB (Nikam et al., 2021; Ouyang-Zhang et al., 2023). All benchmarks are further restricted to variants with single-chain source structures. The suffix -D denotes double-mutation subsets, while -S denotes single-mutation subsets. Our primary evaluation metrics are Spearman correlation (ρ) and area under the receiver operating characteristic curve at stabilization threshold t, denoted AUC_t. Since $\Delta\Delta G$ sign conventions differ across benchmarks, we orient the experimental measurements so that larger values consistently indicate greater stabilization. A mutation is considered stabilizing at threshold t when its oriented stability change exceeds t kcal/mol. See Appendix B for details on dataset filtering, prediction coverage, score orientation, and metric computation.

Table 1: **Double-mutant $\Delta\Delta G$ prediction on single-chain source structures.** Each cell reports global / mean per-protein performance, and n and P denote the numbers of mutations and proteins. **Bold** indicates the best overall value; underlining indicates the best self-supervised value. *Test data leakage. †Reduced prediction coverage. AUC_t identifies variants whose stabilization-oriented effect exceeds t kcal/mol.

	ThermoMutDB-D $n = 504,\ P = 73$			Megascale-D $n = 18{,}574,\ P = 20$			PTMul-D $n = 408,\ P = 52$		
Method	ρ	AUC_0	$\mathrm{AUC}_{0.5}$	ρ	AUC_0	$\mathrm{AUC}_{0.5}$	ρ	AUC_0	$\mathrm{AUC}_{0.5}$
Supervised Methods									
Mutate Everything*	.35/.37	.68/.70	.65/.58	–	–	–	.38/.20	.68/.54	.66/.46
SPURS	.38/.40	.69/.77	.67/**.70**	**.57**/.62	**.80**/.85	**.80/.86**	.50/.29	.76/.67	.75/.67
ThermoMPNN-D	.42/.30	.68/.73	.67/.61	.55/**.62**	.80/**.86**	.78/.84	.58/.44	.79/.70	**.79**/.69
PottsMPNN-FT	.44/.40	.68/.74	.67/.67	.50/.59	.66/.84	.61/.81	.59/.37	.77/.63	.76/.66
Self-supervised Methods									
MSA frequency	.05/.25	.50/.57	.48/.51	.08/.11	.52/.55	.53/.52	.08/.12	.52/.56	.49/.52
EvoRank†	.37/.13	.63/.55	.63/.58	.20/.21	.66/.55	.66/.42	.47/.27	.69/.60	.70/.59
ProteinMPNN	.40/.25	.67/.64	.65/.58	.23/.27	.51/.64	.42/.54	.58/.29	.77/.61	.75/.62
ProteinMPNN+ref	.41/.34	.69/.66	.66/.58	.42/.43	.61/.72	.56/.61	.60/.36	.79/.62	.76/.63
PottsMPNN	.41/.28	.66/.63	.65/.58	.29/.36	.50/.63	.43/.58	.56/.31	.75/.60	.75/.62
Caliby	.39/.23	.66/.62	.64/.59	.19/.30	.51/.65	.45/.57	.53/.23	.73/.60	.72/.63
EvoEpi (additive)	.46/.43	.71/.76	.68/.67	.40/.44	.64/.71	.61/.57	.60/.38	.80/.71	.77/.70
EvoEpi	**.50/.50**	**.72/.78**	**.70**/.68	.42/.55	.67/.82	.66/.79	**.62/.51**	**.81/.78**	.78/**.75**

Supervised baselines include Mutate Everything (ME) (Ouyang-Zhang et al., 2023), SPURS (Li & Luo, 2025), ThermoMPNN-D (Dieckhaus & Kuhlman, 2025), and PottsMPNN-FT (Birnbaum & Keating, 2026). These methods are trained or fine-tuned using labeled stability measurements from Megascale (Tsuboyama et al., 2023). We omit ME's Megascale-D and Megascale-S results due to training–test data leakage.

Self-supervised baselines include empirical MSA frequencies, EvoRank (Gong et al., 2024), ProteinMPNN (Dauparas et al., 2022), PottsMPNN (Birnbaum & Keating, 2026), and Caliby (Shuai et al., 2025). These models have comparable model sizes to *EvoEpi*. ProteinMPNN+ref applies MobiDB20 reference calibration following Frellsen et al. (2026). For models with a native energy-based mutation score, namely PottsMPNN and Caliby, we use the corresponding energy difference rather than the log-likelihood gap in Eq. 1.

To evaluate whether direct joint scoring captures information beyond independently scoring the two substitutions, we additionally construct an additive variant of *EvoEpi* using the same learned model. Let $g_{ij}(v_i, v_j \mid \mathbf{s}, \mathbf{x})$ denote the predicted mutation score after reference calibration, for replacing the WT pair (s_i, s_j) with (v_i, v_j). For a double mutation with $v_i \neq s_i$ and $v_j \neq s_j$, we define the additive score as

$$g_{ij}^{\mathrm{add}}(v_i, v_j \mid \mathbf{s}, \mathbf{x}) = g_{ij}(v_i, s_j \mid \mathbf{s}, \mathbf{x}) + g_{ij}(s_i, v_j \mid \mathbf{s}, \mathbf{x}). \tag{6}$$

Each term scores one substitution while keeping the other site at its WT residue. Comparing the additive and direct scores provides a rigorous test of whether *EvoEpi* captures epistatic effects between the two mutations.

Ranking mutations within proteins. We first address Q1. Protein engineering seeks beneficial variants around a fixed WT sequence, as described in Section 2.1. Accordingly, the relevant prediction task is often to rank candidate mutations **within** a target protein. A global metric across an entire benchmark can obscure this objective as it reflects both within-protein ranking and systematic differences between proteins. It can also give disproportionate weight to proteins with larger mutation panels. We therefore report both global metrics and the unweighted mean of metrics computed separately for each eligible protein. A protein is included in the per-protein evaluation only if it contains at least three scored mutations. We denote mean per-protein metrics using a hat (e.g. $\hat{\rho}$) and report them alongside their global counterparts. These per-protein metrics more directly measure a model's ability to prioritize mutations within individual protein-engineering targets.

Table 2: **Single-mutant $\Delta\Delta G$ prediction on single-chain source structures.** Metric definitions and notation follow Table 1.

	ThermoMutDB-S $n = 4{,}235,\ P = 189$			Megascale-S $n = 28{,}172,\ P = 28$		
Method	ρ	AUC_0	$\mathrm{AUC}_{0.5}$	ρ	AUC_0	$\mathrm{AUC}_{0.5}$
Supervised Methods						
Mutate Everything*	.48/.37	.69/.72	.64/.69	–	–	–
SPURS	**.60/.47**	**.79/.80**	**.76/.76**	**.74/.80**	**.83/.87**	**.82/.84**
ThermoMPNN-D	.56/.37	.75/.75	.74/.75	.68/.74	.80/.83	.79/.81
PottsMPNN-FT	.59/.42	.78/.77	.76/.75	.70/.76	.82/.85	.81/.84
Self-supervised Methods						
MSA frequency	.20/.30	.55/.63	.55/.61	.20/.31	.55/.57	.48/.50
EvoRank†	.43/.23	.69/.65	.67/.67	.39/.41	.67/.68	.62/.62
ProteinMPNN	.41/.32	.68/.73	.66/.69	.53/.58	.73/.76	.66/.68
ProteinMPNN+ref	.45/.36	.71/.74	.68/.71	.58/.64	.75/.78	.70/.72
PottsMPNN	.47/.32	.71/.71	.69/.69	.57/.62	.73/.77	.66/.68
Caliby	.43/.31	.70/.71	.67/.67	.52/.58	.72/.76	.65/.68
EvoEpi (single)	.54/.42	.74/.76	.69/.71	.60/.66	.75/.79	.73/.75

Double-mutant stability. We next address Q2. Table 1 reports $\Delta\Delta G$ prediction performance for double mutations. *EvoEpi* performs strongly under both global and per-protein evaluation. On ThermoMutDB-D and PTMul-D, it achieves the highest ρ and $\hat{\rho}$ among all evaluated methods, including supervised predictors. On Megascale-D, *EvoEpi* outperforms all self-supervised baselines in both metrics while remaining competitive with supervised methods trained on the Megascale training set.

The comparison with *EvoEpi* (add) further highlights the benefit of modeling the two mutations jointly. Direct scoring improves global ρ on all three benchmarks, while the gains are substantially larger for $\hat{\rho}$. Specifically, $\hat{\rho}$ improves by 16%, 25%, and 34% on ThermoMutDB-D, Megascale-D, and PTMul-D, respectively.

The AUC results show a similar overall trend for identifying stabilizing double mutations. *EvoEpi* achieves the strongest AUC_0 on ThermoMutDB-D and PTMul-D under both global and per-protein evaluation. Megascale-D is substantially more imbalanced, with only 3.6% and 0.96% of variants classified as stabilizing at thresholds of 0 and 0.5 kcal/mol, respectively. Even in this setting, *EvoEpi* consistently outperforms the self-supervised baselines and remains competitive with PottsMPNN-FT. Overall, the results indicate that joint pairwise scoring is effective both for global double-mutant prediction and for prioritizing mutations within individual proteins.

Single-mutant stability. We additionally evaluate our model's ability to predict single mutational effect using ThermoMutDB-S and the Megascale-S test split. Although *EvoEpi* is designed around pairwise mutation scoring, single-site transitions are naturally included during training since the sampled source and destination pair states may differ at only one endpoint. This allows the same model to learn from both single- and double-mutation transitions within a unified pairwise formulation.

At inference, however, scoring a single mutation requires choosing a second residue to form the pairwise input. For a mutation at position i from the WT residue s_i to residue r, we thereby define $\mathcal{N}_i$ as the set of the 16 nearest eligible residues to position i and average the mutation score across these local pairs:

$$g_i(r \mid \mathbf{s}, \mathbf{x}) = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} g_{ij}(r, s_j \mid \mathbf{s}, \mathbf{x}). \tag{7}$$

For notational simplicity, Eq. 7 assumes $i < j$. In practice, the pair indices and corresponding residue states are reordered consistently before scoring when $j < i$.

Table 2 shows that *EvoEpi* performs strongly on single-mutant stability prediction under both global and mean per-protein evaluation. It consistently outperforms the self-supervised baselines and substantially narrows the gap to supervised predictors. These results demonstrate that the pairwise

Table 3: **Double-mutant global Spearman correlations across non-stability phenotypes.** Results for *EvoEpi* are reported without reference calibration. The Mean column reports the unweighted average of Spearman correlations across the six assays.

Method / dataset	IRED	FLAb	ppluGFP2	amacGFP	His3	KRas	Mean
Phenotype	Activity	Expression	Fluorescence	Fluorescence	Organismal fitness	Expression	–
n	3,200	2,120	15,992	11,260	1,475	22,946	–
Mutate Everything	**.24**	.02	.04	.13	.19	.27	.15
SPURS	.23	.14	.13	.16	.25	.29	.20
ThermoMPNN-D	.20	.10	.03	.15	.24	.31	.17
PottsMPNN-FT	.22	.30	.08	.16	**.26**	.36	.23
MSA frequency	.18	.14	.04	.03	.15	.17	.12
EvoRank	.15	.29	.17	.17	.14	.32	.21
ProteinMPNN	.14	.39	.14	.14	.18	.36	.22
ProteinMPNN+ref	.17	.26	.17	.16	.22	.25	.20
PottsMPNN	.17	.24	.12	.15	.23	**.45**	.23
Caliby	.14	.36	.08	.12	.13	.36	.20
EvoEpi	.18	**.44**	**.18**	**.18**	.15	.40	.25

scoring formulation can effectively model single-mutation effects while maintaining the same training objective as for double mutations.

Additional phenotypes. To address Q3, we evaluate the same model on six double-mutant assays spanning non-stability phenotypes. As each assay measures variants of a single protein, we report only global ρ and do not separately consider mean per-protein metrics. We also omit reference calibration, which is designed specifically for $\Delta\Delta G$ prediction and does not naturally apply to these other phenotypes. IRED measures enzymatic activity, while FLAb measures protein expression. The remaining four assays are drawn from ProteinGym (Notin et al., 2023): ppluGFP2 and amacGFP measure fluorescence, His3 measures an activity-related organismal-fitness phenotype, and KRas measures expression. As shown in Table 3, *EvoEpi* achieves a mean Spearman correlation of .25, compared with .23 for the strongest baseline. It also achieves the highest correlation on FLAb expression, ppluGFP2 fluorescence, and amacGFP fluorescence. These results support the generalization of MSA-supervised mutation scoring beyond thermodynamic stability.

5 Conclusion

We introduced *EvoEpi*, a self-supervised framework for learning mutation effects from joint amino-acid pair frequencies in MSAs. Across three double-mutant $\Delta\Delta G$ benchmarks, *EvoEpi* demonstrates strong performance under both global and mean per-protein evaluation, consistently outperforming self-supervised baselines and exceeding the evaluated supervised predictors on ThermoMutDB-D and PTMul-D. Its strong mean per-protein performance further highlights its practical value for ranking candidate mutations within individual proteins. Direct joint scoring also improves over the additive scoring variant, particularly for within-protein ranking. *EvoEpi* remains competitive on single mutations and across additional phenotypes including activity, expression, fluorescence, and organismal fitness, demonstrating generalization beyond thermodynamic stability.

That said, the current framework is limited to single-chain proteins, fixed backbone structures, and primarily single- and double-mutation settings, reflecting the current design of our MSA-supervised training pipeline. Future work includes extending MSA supervision to protein assemblies using paired MSAs, higher-order mutations, and settings involving structural rearrangements. More broadly, our results suggest that evolutionary statistics can provide scalable supervision for mutation-effect prediction without phenotype-specific labels, offering a path toward more general protein-engineering models.

AI USE STATEMENT

Generative AI tools were used to improve the readability, grammar, and presentation of the manuscript and to refactor portions of the research code after the initial implementation. They were not used to formulate the research hypotheses, design the methodology or experiments, or develop the initial implementation. All AI-assisted manuscript edits and code changes were reviewed and validated by the authors, who take full responsibility for the content of the paper and the reported results.

ETHICS STATEMENT

This work uses publicly available protein sequence, structure, evolutionary, and mutation-effect data and does not involve human subjects or personally identifiable information. Protein engineering methods may have potential dual-use applications. The present work focuses on computational prediction of mutation effects and does not design or experimentally validate harmful biological agents. We encourage responsible use of the proposed method in accordance with applicable research and bio-safety standards.

REFERENCES

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, 2024.

Frances H Arnold. Design by directed evolution. *Accounts of chemical research*, 31(3):125–131, 1998.

Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J Fleet. Synthetic data from diffusion models improves imagenet classification. *arXiv preprint arXiv:2304.08466*, 2023.

Sivaraman Balakrishnan, Hetunandan Kamisetty, Jaime G Carbonell, Su-In Lee, and Christopher James Langmead. Learning generative models for protein fold families. *Proteins: Structure, Function, and Bioinformatics*, 79(4):1061–1078, 2011.

Elizabeth L Bell, William Finnigan, Scott P France, Anthony P Green, Martin A Hayes, Lorna J Hepworth, Sarah L Lovelock, Haruka Niikura, Sílvia Osuna, Elvira Romero, et al. Biocatalysis. *Nature Reviews Methods Primers*, 1(1):46, 2021.

Steven A Benner and Dietlinde Gerloff. Patterns of divergence in homologous proteins as indicators of secondary and tertiary structure: a prediction of the structure of the catalytic domain of protein kinases. *Advances in Enzyme Regulation*, 31:121–181, 1991.

Foster Birnbaum and Amy E Keating. Beyond native sequence recovery: Improved modeling of the sequence-energy landscape of protein structures. *Proceedings of the National Academy of Sciences*, 123(27):e2535494123, 2026.

Matteo Cagiada, Sergey Ovchinnikov, and Kresten Lindorff-Larsen. Predicting absolute protein folding stability using generative models. *Protein Science*, 34(1):e5233, 2025.

Rayan Chikhi, Téo Lemane, Raphaël Loll-Krippleber, Mercè Montoliu-Nerin, Brice Raffestin, Antonio Pedro Camargo, Carson J Miller, Mateus Bernabe Fiamenghi, Daniel Paiva Agustinho, Sina Majidian, et al. Logan: planetary-scale genome assembly surveys life's diversity. *Biorxiv*, pp. 2024–07, 2025.

UniProt Consortium. Uniprot: a worldwide hub of protein knowledge. *Nucleic acids research*, 47 (D1):D506–D515, 2019.

Andre Cornman, Jacob West-Roberts, Antonio Pedro Camargo, Simon Roux, Martin Beracochea, Milot Mirdita, Sergey Ovchinnikov, and Yunha Hwang. The omg dataset: An open metagenomic corpus for mixed-modality genomic language modeling. *bioRxiv*, pp. 2024–08, 2024.

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.

Henry Dieckhaus and Brian Kuhlman. Protein stability models fail to capture epistatic interactions of double point mutations. *Protein Science*, 34(1):e70003, 2025.

Oliver Dutton, Sandro Bottaro, Michele Invernizzi, Istvan Redl, Albert Chung, Falk Hoffmann, Louie Henderson, Stefano Ruschetta, Fabio Airoldi, Benjamin MJ Owens, et al. Improving inverse folding models at protein stability prediction without additional training or data. *bioRxiv*, pp. 2024–06, 2024.

Douglas M Fowler and Stanley Fields. Deep mutational scanning: a new style of protein science. *Nature methods*, 11(8):801–807, 2014.

Jes Frellsen, Maher Kassem, Tone Bengtsen, Lars Olsen, Kresten Lindorff-Larsen, Jesper Ferkinghoff-Borg, and Wouter Boomsma. Zero-shot protein stability prediction by inverse folding models: a free energy interpretation. *Advances in Neural Information Processing Systems*, 38:84641–84667, 2026.

Michaela Gebauer and Arne Skerra. Engineered protein scaffolds as next-generation therapeutics. *Annual review of pharmacology and toxicology*, 60(1):391–415, 2020.

Lori Giver, Anne Gershenson, Per-Ola Freskgard, and Frances H Arnold. Directed evolution of a thermostable esterase. *Proceedings of the National Academy of Sciences*, 95(22):12809–12813, 1998.

Chengyue Gong, Adam Klivans, James Madigan Loy, Tianlong Chen, Daniel Jesus Diaz, et al. Evolution-inspired loss functions for protein representation learning. In *Forty-first International Conference on Machine Learning*, 2024.

Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and Xiaojuan Qi. Is synthetic data from generative models ready for image recognition? *arXiv preprint arXiv:2210.07574*, 2022.

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.

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*, pp. 8946–8970. PMLR, 2022.

Alex J Li, Mindren Lu, Israel Desta, Vikram Sundar, Gevorg Grigoryan, and Amy E Keating. Neural network-derived potts models for structure-based protein design using backbone atomic coordinates and tertiary motifs. *Protein Science*, 32(2):e4554, 2023.

Ziang Li and Yunan Luo. Generalizable and scalable protein stability prediction with rewired protein generative models. *Nature Communications*, 17(1):891, 2025.

Dina Listov, Casper A Goverde, Bruno E Correia, and Sarel Jacob Fleishman. Opportunities and challenges in design and optimization of protein function. *Nature Reviews Molecular Cell Biology*, 25(8):639–653, 2024.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion modeling by estimating the ratios of the data distribution. *arXiv preprint arXiv:2310.16834*, 2023.

Gautam Kumar Meghwanshi, Navpreet Kaur, Swati Verma, Narendra Kumar Dabi, Abhishek Vashishtha, PD Charan, Praveen Purohit, HS Bhandari, N Bhojak, and Rajender Kumar. Enzymes for pharmaceutical and therapeutic applications. *Biotechnology and applied biochemistry*, 67(4):586–601, 2020.

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.

Yu Meng, Jiaxin Huang, Yu Zhang, and Jiawei Han. Generating training data with language models: Towards zero-shot language understanding. *Advances in Neural Information Processing Systems*, 35:462–477, 2022.

Faruck Morcos, Andrea Pagnani, Bryan Lunt, Arianna Bertolino, Debora S Marks, Chris Sander, Riccardo Zecchina, José N Onuchic, Terence Hwa, and Martin Weigt. Direct-coupling analysis of residue coevolution captures native contacts across many protein families. *Proceedings of the National Academy of Sciences*, 108(49):E1293–E1301, 2011.

Rahul Nikam, Abhishek 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.

Pascal Notin, Aaron Kollasch, Daniel Ritter, Lood Van Niekerk, Steffanie Paul, Han Spinner, Nathan Rollins, Ada Shaw, Rose Orenbuch, Ruben Weitzman, et al. Proteingym: Large-scale benchmarks for protein fitness prediction and design. *Advances in neural information processing systems*, 36: 64331–64379, 2023.

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: 76229–76247, 2023.

Saro Passaro, Gabriele Corso, Jeremy Wohlwend, Mateo Reveiz, Stephan Thaler, Vignesh Ram Somnath, Noah Getz, Tally Portnoi, Julien Roy, Hannes Stark, et al. Boltz-2: Towards accurate and efficient binding affinity prediction. *BioRxiv*, 2025.

Damiano Piovesan, Marco Necci, Nahuel Escobedo, Alexander Miguel Monzon, András Hatos, Ivan Mičetić, Federica Quaglia, Lisanna Paladin, Pathmanaban Ramasamy, Zsuzsanna Dosztányi, et al. Mobidb: intrinsically disordered proteins in 2021. *Nucleic acids research*, 49(D1):D361–D367, 2021.

Roshan M Rao, Jason Liu, Robert Verkuil, Joshua Meier, John Canny, Pieter Abbeel, Tom Sercu, and Alexander Rives. Msa transformer. In *International conference on machine learning*, pp. 8844–8856. PMLR, 2021.

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.

Richard W Shuai, Tianyu Lu, Subhang Bhatti, Petr Kouba, and Po-Ssu Huang. Ensemble-conditioned protein sequence design with caliby. *bioRxiv*, 2025.

Lea M Starita, Nadav Ahituv, Maitreya J Dunham, Jacob O Kitzman, Frederick P Roth, Georg Seelig, Jay Shendure, and Douglas M Fowler. Variant interpretation: functional assays to the rescue. *The American Journal of Human Genetics*, 101(3):315–325, 2017.

Amelie Stein, Douglas M Fowler, Rasmus Hartmann-Petersen, and Kresten Lindorff-Larsen. Biophysical and mechanistic models for disease-causing protein variants. *Trends in biochemical sciences*, 44(7):575–588, 2019.

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.

Martin Steinegger, Milot Mirdita, and Johannes Söding. Protein-level assembly increases protein sequence recovery from metagenomic samples manyfold. *Nature methods*, 16(7):603–606, 2019.

Baris E Suzek, Hongzhan Huang, Peter McGarvey, Raja Mazumder, and Cathy H Wu. Uniref: comprehensive and non-redundant uniprot reference clusters. *Bioinformatics*, 23(10):1282–1288, 2007.

Julie D Thompson, Desmond G Higgins, and Toby J Gibson. Clustal w: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic acids research*, 22(22):4673–4680, 1994.

Julie D Thompson, Toby J Gibson, Frédéric Plewniak, François Jeanmougin, and Desmond G Higgins. The clustal_x windows interface: flexible strategies for multiple sequence alignment aided by quality analysis tools. *Nucleic acids research*, 25(24):4876–4882, 1997.

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.

Mihaly Varadi, Stephen Anyango, Mandar Deshpande, Sreenath Nair, Cindy Natassia, Galabina Yordanova, David Yuan, Oana Stroe, Gemma Wood, Agata Laydon, et al. Alphafold protein structure database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. *Nucleic acids research*, 50(D1):D439–D444, 2022.

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.

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.

Kang Min Yoo, Dongju Park, Jaewook Kang, Sang-Woo Lee, and Woomyoung Park. Gpt3mix: Leveraging large-scale language models for text augmentation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pp. 2225–2239, 2021.

Yuxuan Zhang, Chengyue Gong, Jinyuan Sun, Jiaqi Guan, Milong Ren, Song Xue, Hanyu Zhang, Wenzhi Ma, Zhenyu Liu, Xinshi Chen, et al. Protenix-v2: Broadening the reach of structure prediction and biomolecular design. *bioRxiv*, pp. 2026–04, 2026.

CONTENTS

1 Introduction **1**

2 Background **3**

2.1 Protein Engineering 3

2.2 Multiple Sequence Alignments 3

3 Methods **3**

3.1 Pairwise transition scoring model 4

3.2 Supervising double mutation feasibility from the MSA 5

3.3 Reference calibration at inference time 6

4 Experiments **6**

4.1 Training Details 6

4.2 Results 6

5 Conclusion **9**

A Training Details **15**

A.1 Training pipeline 15

A.2 Transition regression 15

B Appendix B: Experiment Details **15**

C Appendix C: Additional Experiments **16**

C.1 Results including multi-chain sources 16

C.2 Ablation Studies 18

A Training Details

A.1 Training pipeline

We implement the training procedure in Section 3.1 using three MPNN encoder layers and three MPNN decoder layers, hidden dimension 128, and a 48-neighbor residue graph. The MPNN encoder and decoder are initialized from pretrained ProteinMPNN weights (Dauparas et al., 2022), while the pair-state embeddings and regression head are initialized separately. Unlike ProteinMPNN, we do not add Gaussian noise to the backbone coordinates during training. Since our objective is to regress mutation transition scores rather than generate sequences, we train the scoring model on unperturbed input structures. The MLPs in the regression head use GELU activations.

We train our model for 20 epochs using AdamW with a learning rate of 3×10^{-4} and weight decay 0.01, using dynamic batches with a budget of $10,000$ padded residues. During training, we maintain an exponential moving average (EMA) of the model parameters with decay 0.999 and use the EMA weights for all reported evaluations. We set $K = 16$. The model is trained on a single NVIDIA A100 80GB GPU for 6 hours.

A.2 Transition regression

For each valid selected pair, we independently sample starting and destination states $A, B$with replacement, repeating this draw 64 times in the current configuration. The states may coincide or differ at only one endpoint. The target is the clipped log-frequency ratio

$$y_{ij}(A, B) = \mathrm{clip}_{[-10,10]}\left[\log \frac{q_{ij}(B) + \epsilon}{q_{ij}(A) + \epsilon}\right], \tag{8}$$

where $\epsilon = 10^{-8}$. For residual r, we use the Huber penalty

$$H_\delta(r) = \begin{cases} \frac{1}{2}r^2, & |r| \le \delta, \\ \delta\big(|r| - \frac{1}{2}\delta\big), & |r| > \delta, \end{cases} \qquad \delta = 1.5. \tag{9}$$

B Appendix B: Experiment Details

We orient experimental effects toward stabilization using $g = \Delta\Delta G$ for ThermoMutDB and $g = -\Delta\Delta G$ for Megascale and PTMul. Alongside Spearman correlation, we report AUC_t at $t = 0$ and 0.5 kcal/mol, with $g > t$ positive and $g \le t$ negative. For Megascale-D, the positive fractions are 3.6% and 0.96%, respectively.

Each metric is reported globally over all scored measurements and as an unweighted mean over eligible proteins. Per-protein Spearman requires at least three measurements and nonconstant targets and predictions; AUROC requires at least three measurements and both classes. Constant predictions receive AUROC 0.5.

Prediction coverage. Table headers give full-cohort counts. Missing predictions are omitted separately for each method, without imputation or a cross-model intersection. On single-chain sources, EvoRank covers $n/P = 481/63$ for ThermoMutDB-D, $18{,}260/19$ for Megascale-D, $406/50$ for PTMul-D, $3{,}909/143$ for ThermoMutDB-S, and $20{,}567/20$ for Megascale-S. On all source-chain classes, its coverage increases to $631/100$ for ThermoMutDB-D, $527/79$ for PTMul-D, and $5{,}028/248$ for ThermoMutDB-S. ME's Megascale results are omitted due to training–test data leakage.

Baselines and scoring recipes. For double mutations, EvoRank sums individual scores, while MSA frequency uses joint pair counts; for single mutations, MSA frequency uses single-column counts. PottsMPNN and Caliby use their native energy differences. ME and SPURS outputs are oriented toward stabilization and treated as ranking scores.

C Appendix C: Additional Experiments

C.1 Results including multi-chain sources

Tables 4 and 5 report double- and single-mutant stability results, respectively, including both single-chain and multi-chain source structures. Megascale results coincide with the main tables because its source structures are all single-chain.

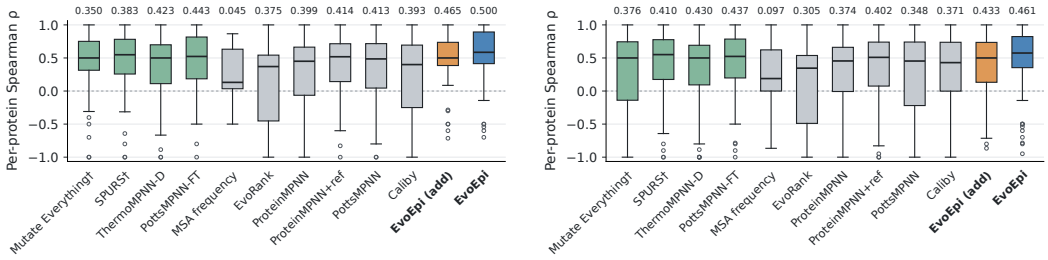
Table 4: **Double-mutant $\Delta\Delta G$ prediction on all source-chain classes.** Metric definitions and notation follow Table 1. Megascale all-chain and single-chain cohorts coincide.

	ThermoMutDB-D $n = 700,\ P = 121$			Megascale-D $n = 18{,}574,\ P = 20$			PTMul-D $n = 536,\ P = 83$		
Method	ρ	AUC_0	$\mathrm{AUC}_{0.5}$	ρ	AUC_0	$\mathrm{AUC}_{0.5}$	ρ	AUC_0	$\mathrm{AUC}_{0.5}$
Supervised Methods									
Mutate Everything*	.38/.29	.68/.66	.66/.62	–	–	–	.37/.23	.68/.59	.67/.52
SPURS	.41/.37	.70/.73	.69/**.71**	**.57**/.62	**.80**/.85	**.80/.86**	.48/.25	.75/.63	.74/.62
ThermoMPNN-D	.43/.30	.70/.70	**.70**/.67	.55/**.62**	.80/**.86**	.78/.84	.57/**.37**	**.78/.67**	**.77**/.63
PottsMPNN-FT	.44/.40	.68/.73	.68/.71	.50/.59	.66/.84	.61/.81	.55/.34	.75/.62	.73/**.66**
Self-supervised Methods									
MSA frequency	.10/.22	.53/.59	.52/.57	.08/.11	.52/.55	.53/.52	.11/.17	.54/.53	.52/.51
EvoRank†	.31/.12	.60/.53	.60/.56	.20/.21	.66/.55	.66/.42	.42/.28	.67/.62	.67/.61
ProteinMPNN	.37/.28	.67/.68	.65/.65	.23/.27	.51/.64	.42/.54	.53/.26	.75/.60	.73/.61
ProteinMPNN+ref	.40/.35	.70/.70	.67/.65	.42/.43	.61/.72	.56/.61	.56/.30	.77/.61	.74/.60
PottsMPNN	.35/.27	.65/.62	.64/.58	.29/.36	.50/.63	.43/.58	.51/.24	.72/.54	.71/.54
Caliby	.37/.30	.67/.66	.66/.64	.19/.30	.51/.65	.45/.57	.49/.21	.71/.57	.69/.57
EvoEpi (additive)	.43/.37	.71/.75	.68/.65	.40/.44	.64/.71	.61/.57	.56/.28	.77/.62	.74/.60
EvoEpi	**.46/.43**	**.72/.75**	.69/.69	.42/.55	.67/.82	.66/.79	**.58**/.35	.78/.66	.75/.64

Table 5: **Single-mutant $\Delta\Delta G$ prediction on all source-chain classes.**

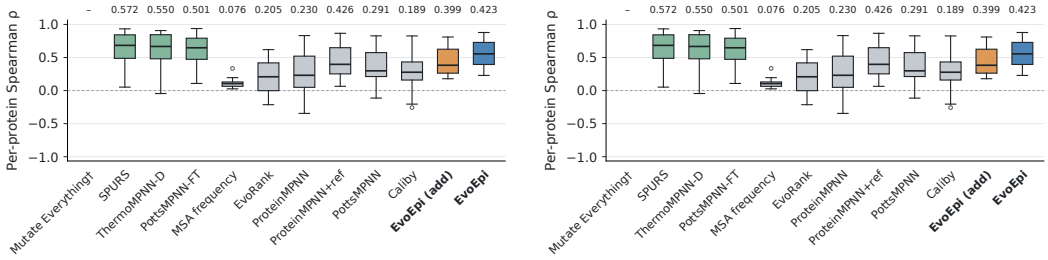
	ThermoMutDB-S $n = 5{,}647,\ P = 326$			Megascale-S $n = 28{,}172,\ P = 28$		
Method	ρ	AUC_0	$\mathrm{AUC}_{0.5}$	ρ	AUC_0	$\mathrm{AUC}_{0.5}$
Supervised Methods						
Mutate Everything*	.48/.37	.69/.72	.64/.69	–	–	–
SPURS	**.60/.47**	**.79/.80**	**.76/.76**	**.74/.80**	**.83/.87**	**.82/.84**
ThermoMPNN-D	.55/.37	.76/.74	.75/.75	.68/.74	.80/.83	.79/.81
PottsMPNN-FT	.57/.42	.78/.77	.76/.75	.70/.76	.82/.85	.81/.84
Self-supervised Methods						
MSA frequency	.22/.25	.56/.61	.55/.58	.20/.31	.55/.57	.48/.50
EvoRank†	.42/.26	.69/.66	.67/.67	.39/.41	.67/.68	.62/.62
ProteinMPNN	.42/.34	.69/.73	.68/.70	.53/.58	.73/.76	.66/.68
ProteinMPNN+ref	.46/.36	.72/.74	.69/.72	.58/.64	.75/.78	.70/.72
PottsMPNN	.46/.33	.71/.72	.69/.70	.57/.62	.73/.77	.66/.68
Caliby	.43/.31	.70/.72	.69/.69	.52/.58	.72/.76	.65/.68
EvoEpi (single)	.52/.40	.74/.75	.70/.71	.60/.66	.75/.79	.73/.75

Per-protein correlation distributions. Figures 3–5 show boxplots of per-protein Spearman correlations for ThermoMutDB-D, Megascale-D, and PTMul-D, respectively. Each figure presents single-chain sources on the left and all source-chain classes on the right; numbers above the boxes report global correlations.



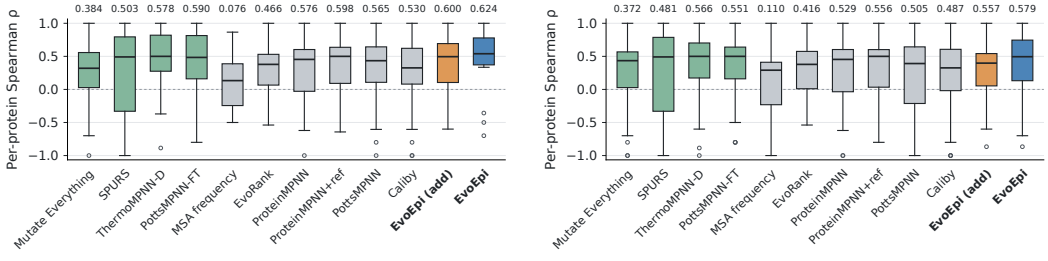
(a) Single-chain sources. (b) All source-chain classes.

Figure 3: **ThermoMutDB-D: double-mutant per-protein Spearman correlations.** Boxes show the median and interquartile range across eligible proteins; whiskers extend to the most extreme values within 1.5 interquartile ranges, and open circles denote outliers. Numbers above boxes are *global* correlations. Green denotes supervised baselines, gray other baselines, orange *EvoEpi* (additive), and blue direct *EvoEpi*. A † denotes test data leakage here, denoted by an * in the tables.



(a) Single-chain sources. (b) All source-chain classes.

Figure 4: **Megascale-D: double-mutant per-protein Spearman correlations.** Plot conventions and method order follow Figure 3. The single-chain and all-chain cohorts are identical.



(a) Single-chain sources. (b) All source-chain classes.

Figure 5: **PTMul-D: double-mutant per-protein Spearman correlations.** Plot conventions and method order follow Figure 3.

C.2 Ablation Studies

Training sequence context. Table 6 compares three choices of sequence context for the first training stage on the single-chain double-mutant benchmarks. *GT* uses native sequences without redesign; *Prior* resamples selected residues using the frozen ProteinMPNN prior; and *MSA profiles* samples residues independently from position-specific MSA frequencies. These alternatives examine the role of sequence augmentation while retaining the backbone structure. Note that training targets are derived from MSA pair frequencies regardless of the sequence context choice.

Reference calibration. For each context, *Without* uses the model's transition score directly, whereas *With* subtracts the IDR-reference log-frequency ratio at inference. This comparison measures the contribution of reference calibration within each checkpoint, separately from the choice of training context. For Prior, the mean per-protein Spearman correlations are .30, .29, and .29 without calibration and .50, .55, and .51 with calibration on ThermoMutDB-D, Megascale-D, and PTMul-D, respectively. The direction and magnitude of the change depend on the context and benchmark, as shown by the mixed changes for MSA profiles.

Table 6: **Training context and reference calibration on single-chain double-mutant $\Delta\Delta G$ benchmarks.** Cells report global / mean per-protein metrics, with definitions as in Table 1.

		ThermoMutDB-D $n = 504,\ P = 73$			Megascale-D $n = 18{,}574,\ P = 20$			PTMul-D $n = 408,\ P = 52$		
Context	Ref.	ρ	AUC_0	$\mathrm{AUC}_{0.5}$	ρ	AUC_0	$\mathrm{AUC}_{0.5}$	ρ	AUC_0	$\mathrm{AUC}_{0.5}$
GT	Without	.02/.20	.45/.61	.41/.53	−.09/.22	.33/.55	.26/.39	.08/.10	.45/.48	.41/.49
GT	With	.03/.28	.47/.67	.42/.57	.17/.47	.48/.72	.43/.63	.11/.17	.48/.52	.43/.53
Prior	Without	.47/.30	.69/.67	.68/.59	.08/.29	.47/.63	.40/.50	.59/.29	.77/.66	.77/.63
Prior	With	**.50/.50**	**.72/.78**	**.70/.68**	.42/**.55**	.67/**.82**	.66/**.79**	**.62/.51**	**.81/.78**	**.78/.75**
MSA profiles	Without	.31/.22	.61/.58	.58/.53	.11/.20	.39/.57	.33/.46	.37/.30	.62/.58	.58/.58
MSA profiles	With	.28/.25	.63/.68	.59/.57	**.49**/.55	**.69**/.81	**.70**/.78	.34/.29	.65/.66	.59/.65