Towards Interpretable Protein Structure Prediction with Sparse Autoencoders

Nithin Parsan
Reticular

David J. Yang*
University of Pennsylvania

John J. Yang†
Reticular

Abstract

Protein language models have revolutionized structure prediction, but their nonlinear nature obscures how sequence representations inform structure prediction. While sparse autoencoders (SAEs) offer a path to interpretability here by learning linear representations in high-dimensional space, their application has been limited to smaller protein language models unable to perform structure prediction. In this work, we make two key advances: (1) we scale SAEs to ESM2-3B, the base model for ESMFold, enabling mechanistic interpretability of protein structure prediction for the first time, and (2) we adapt Matryoshka SAEs for protein language models, which learn hierarchically organized features by forcing nested groups of latents to reconstruct inputs independently. We demonstrate that our Matryoshka SAEs achieve comparable or better performance than standard architectures. Through comprehensive evaluations, we show that SAEs trained on ESM2-3B significantly outperform those trained on smaller models for both biological concept discovery and contact map prediction. Finally, we present an initial case study demonstrating how our approach enables targeted steering of ESMFold predictions, increasing structure solvent accessibility while fixing the input sequence. To facilitate further investigation by the broader community, we open-source our code, dataset, pretrained models, and visualizer.

1 Introduction

Machine learning-based protein structure prediction models have achieved remarkable success by leveraging vast amounts of sequence data (Jumper et al., 2021; Lin et al., 2022), building on the success of previous sequence homology-based methods (Marks et al., 2011; Kamisetty et al., 2013). However, their nonlinear nature obscures how sequence information informs structure prediction.

Previous interpretability studies on transformer-based protein language models (PLMs) have demonstrated PLMs learn structural information despite only training on sequences. Rao et al. (2021) and Vig et al. (2021) showed that the attention map learns to predict residue-residue contacts. Zhang et al. (2024) show PLMs predict structure primarily by memorizing and retrieving patterns of coevolving residues. However, these works have yet to establish causal relationships between internal mechanisms and predictions.

Sparse autoencoders (SAEs) offer a promising direction for understanding language models by learning interpretable linear representations (Templeton et al., 2024; Gao et al., 2024). Prior works (Simon and Zou, 2024; Adams et al., 2025) train SAEs to uncover thousands of biologically interpretable features in ESM2 models, but these studies focused on smaller variants rather than ESM2-3B, the language model underlying ESMFold's structure prediction.

Our work advances protein model interpretability by scaling SAEs to ESM2-3B (the base model for ESMFold) and also adapting Matryoshka SAEs (Nabeshima, 2024; Bussmann

*Work conducted during an internship at Reticular

†Corresponding author. Contact `john@reticular.ai`

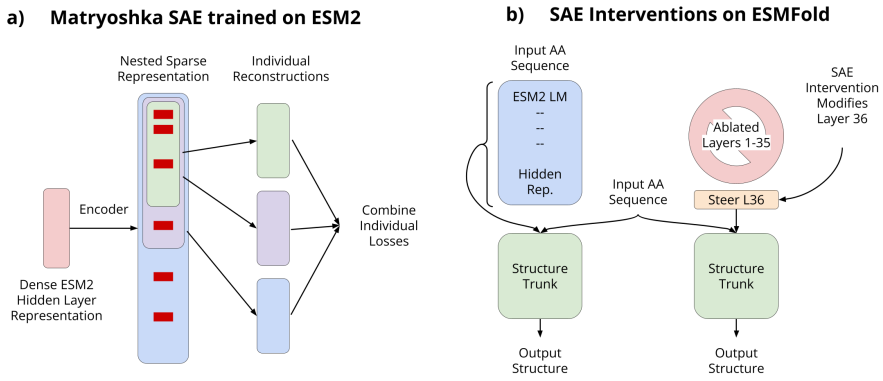


Figure 1: a) Matryoshka Sparse Autoencoder (SAE) architecture for training on ESM2 hidden layer representations, showing nested sparse feature organization. b) SAE intervention framework for ESMFold, comparing normal operation (left) where all ESM2 hidden representations flow to the structure trunk, versus intervention (right) where only a modified layer 36 representation is used while ablating all other layers.

et al., 2024) to learn hierarchically organized features that align with protein structure's multi-scale nature.

We structure the paper as follows: In Section 2, we present scaling SAEs to ESM2-3B and Matryoshka SAEs for hierarchical features. Section 3 shows Matryoshka SAEs match or exceed L1 and TopK in language modeling and structure prediction. Section 4 evaluates ESM2-3B's performance on biological concept discovery and contact prediction. Section 5 presents feature steering to control protein structure properties. Code, models, and a visualizer will be released upon publication.

2 Methods

2.1 Setup

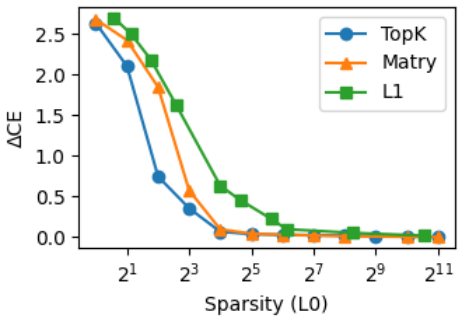
Problem Statement. Given token embeddings $x \in \mathbb{R}^d$ from a transformer-based PLM, our SAE encodes x into a sparse, higher-dimensional latent representation $z \in \mathbb{R}^n$ where $d << n$, and decodes it to reconstruct x by minimizing the L2 loss $\mathcal{L} = |x - \hat{x}|_2^2$. We enforce sparsity on z through methods detailed in Appendix I.1.

Models and Training. We train SAEs on layers 18 and 36 of ESM2 (Lin et al., 2022), focusing on the 3 billion parameter ESM2-3B model which provides representations for ESMFold. During training, embeddings are normalized to enable hyperparameter transfer between layers, with biases scaled during inference following Marks et al. (2024). Our training data consists of 10M sequences randomly selected from UniRef50, constituting 2.5 billion tokens. See Appendix I.3 for details.

2.2 Matryoshka SAEs

Proteins exhibit inherent hierarchical organization across scales, from local amino acid patterns to molecular assemblies. We employ Matryoshka Sparse Autoencoders (SAEs), which learn nested hierarchical representations through embedded features of increasing dimensionality (Fig. 1a) (Nabeshima, 2024; Bussmann et al., 2024).

Our implementation follows Bussmann et al. (2024), Marks et al. (2024), and Gao et al. (2024) in dividing the latent dictionary into nested groups where earlier groups capture abstract features while later groups encode granular details. For mathematical details of the encoding, decoding, and loss computation, see Appendix D.



(a) CE loss reconstruction across sparsity levels for TopK, Matryoshka (Matry), and L1 regularized autoencoders. Matryoshka requires similar or less active latents to achieve good reconstruction.

Comparison	RMSD (Å)
Exp vs. ESMFold (baseline)	3.1 ± 2.5
Exp vs. ESMFold (full ablation)	15.1 ± 5.9
Exp vs. ESMFold (only layer 36)	2.9 ± 2.1
Exp vs. SAE (layer 36)	3.2 ± 2.6
ESMFold vs. SAE (both L36)	3.1 ± 4.4

(b) Backbone RMSD (Å) comparing experimental structures (Exp), ESMFold predictions, and SAE reconstructions. Keeping layer 36 or using SAE preserves accuracy, while full ablation degrades performance.

3 Evaluations on Downstream Loss

We evaluate our SAEs on language modeling and structure prediction for ESM2-3B and ESMFold respectively to assess how well they preserve performance on downstream tasks.

3.1 Language Modeling

Setup. We report the average difference in cross-entropy loss ΔCE between the logits from the original and SAE reconstructed PLM. We evaluate on a heldout test set of 10k sequences randomly sampled from Uniref50 on layer 18 of ESM2.

Results. Figure 2a shows the impact of different autoencoder architectures on downstream language modeling performance across sparsity levels. For architecture details, see Appendix I.2. At low sparsity ($L_0 < 10$), all approaches - TopK, Matryoshka, and L1 regularization - show similar degradation (ΔCE $\approx$ 2.5–3.0). As sparsity increases, TopK and Matryoshka SAEs maintain better performance compared to L1 regularization, with ΔCE approaching 0 for sparsity levels above 100.

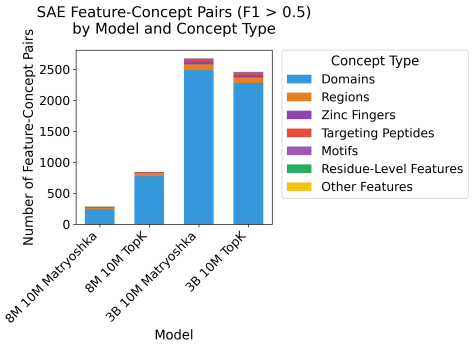
3.2 Structure Prediction

Setup. By scaling SAEs to ESM2-3B, we can now evaluate how well our SAE representations preserve ESMFold's structure prediction capabilities. We focus on reconstructing ESM2's hidden representations that feed into ESMFold. Since ESMFold uses representations from all layers but our SAE reconstructs only one layer, we ablate all other layers to isolate reconstruction effects. This ablation maintains performance on the CASP14 test set (see Fig. 2b).

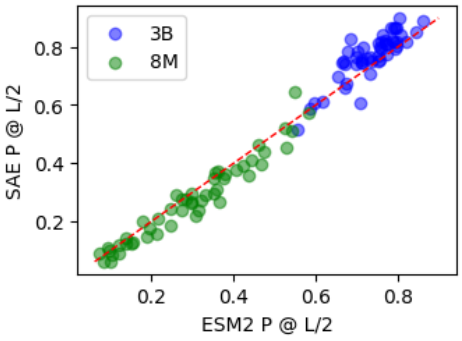
Data. We evaluate structure prediction on the CASP14 dataset, a diverse challenging benchmark held out from ESMFold during training. After filtering for computational constraints and prediction quality (see Appendix F), we analyze 17 protein targets.

Results. Our experiments demonstrate effective preservation of structural information. In Fig. 2b, we see that comparing experimental structures to ESMFold predictions shows an RMSD of 3.1 ± 2.5 Å without ablation. While full ablation of ESM2 embeddings significantly degrades performance (RMSD 15.1 ± 5.9 Å), both keeping only layer 36 (RMSD 2.9 ± 2.1 Å) and using our SAE reconstruction (RMSD 3.2 ± 2.6 Å) maintain comparable performance. The similarity between ESMFold and SAE predictions (RMSD 3.1 ± 4.4 Å) confirms preservation of structural information. We also see that in Fig. 7, with only 8 to 32 active latents, SAEs can reasonably recover structure prediction performance.

These results show our SAE effectively compresses model representations while maintaining both sequence-level and structural prediction capabilities across sparsity levels.



(a) Number of high-performing feature-concept pairs (F1 $>$ 0.5) across model scales and architectures, broken down by concept type.



(b) Correlation plot on long-range contact accuracy measured by Precision at L/2 (P @ L/2) between ESM2 and SAE reconstructions on 8M and 3B subject model sizes.

Figure 3: Analysis of feature-concept relationships and long-range contact accuracy.

4 Further Evaluations

4.1 Swiss-Prot Concept Discovery

We evaluate feature interpretability through alignment with Swiss-Prot annotations following Simon and Zou (2024)'s methodology (details in Appendix E.1), analyzing 30,871,402 amino acid tokens across 476 biological concepts. Features capturing concepts (F1 ¿ 0.5) are identified using domain-level recall with post-hoc [0,1] activation normalization across architectural variants.

Comparing ESM2-3B versus ESM2-8M models (dictionary size 20,480, sparsity k=100), 3B variants substantially outperform their 8M counterparts. The 3B Matryoshka and TopK models identify 233 concepts (48.9%) with F1 $>$ 0.5, generating 2,677 and 2,461 high-quality feature-concept pairs respectively. In contrast, 8M Matryoshka and TopK variants capture only 72 (15.1%) and 95 (20.0%) concepts, with 287 and 844 pairs respectively. Performance gains are particularly pronounced in protein domains (76% versus 19.5-28.1% coverage). Head-to-head comparisons show 3B models achieve higher F1 scores on over 400 concepts (mean improvement 0.25), while architectural differences within each scale remain minimal (Figure 6, details in Appendix E.2).

4.2 Contact Map Prediction

Background. Following Zhang et al. (2024), we evaluate our SAEs' ability to capture coevolutionary statistics through contact map prediction using the Categorical Jacobian. Details in Appendix G. This provides an unsupervised test of whether our compressed representations preserve the structural information encoded in the original model.

Results. We assess contact prediction accuracy using the precision at L/2 metric (P @ L/2), which measures the fraction of correctly predicted contacts among the top L/2 predicted long-range contacts, where L is the protein sequence length. Figure 3b shows the correlation between contact prediction accuracy of ESM2 and our SAE reconstructions across different model scales. The 3B model demonstrates consistently higher precision compared to the 8M model, with reconstructions closely tracking the original ESM2 predictions. This suggests that larger language models capture more robust coevolutionary signals that can be effectively compressed by our approach.

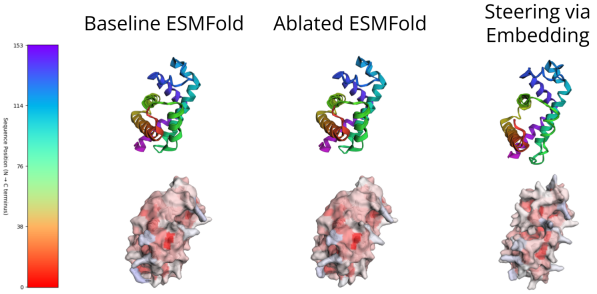
5 Case Study: SAE Feature Steering on ESMFold

5.1 SAEs on ESMFold Help Identify Structural Features

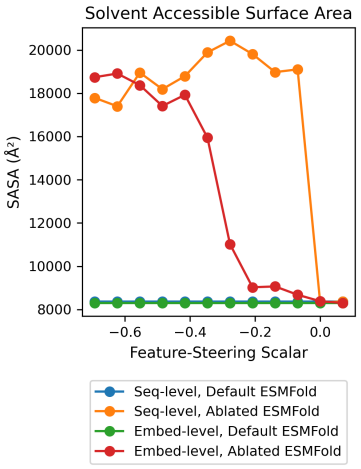
Building on previous work by Simon and Zou (2024), we investigated whether targeted feature manipulation of ESM2-3B could induce coordinated changes in both sequence composition and predicted structure through ESMFold's pipeline. Using our Matryoshka SAE trained on ESM2-3B, we identified a feature strongly correlated with residue hydrophobicity. Following the feature steering methodology of Templeton et al. (2024) and detailed in Appendix Section H, we steer this feature (α = -0.275) and observe significant changes in predicted protein surface accessibility of myoglobin (PDB ID: 1MBN) while maintaining structural integrity (Figure 4a).

Steering increased total Solvent Accessible Surface Area (SASA) by 31.5% (from 8,369.5 to 11,009.3 Å^2) with minimal structural disruption (RMSD 2.76 Å) in the expected range of Figure 2b. Feature intervention affected both sequence-level predictions via steering toward more hydrophilic residues and direct structural predictions of ESMFold (Figure 9). Critically, steering the hidden representation at layer 36 alone was sufficient to induce significant structural changes consistent with increased hydrophilicity, even when providing ESMFold with the correct input sequence (Figure 4b). Further details found in Appendix Section H.2 and feature validation in Appendix Section H.3.

6 Discussion



(a) Structural visualization of feature steering effect on myoglobin with α= -0.275 on selected feature. Bottom row surface representation is colored by computed SASA, with blue as low, white as medium, and red as higher SASA.



(b) Solvent accessible surface area (SASA) changes under different steering conditions.

Figure 4: Feature steering and SASA analysis.

This work advances PLM interpretability by scaling SAEs to ESM2-3B, extending recent work using SAEs to interpret PLMs to the structure prediction task (Simon and Zou (2024); Adams et al. (2025)). Through a combination of increasing subject model size, leveraging the Matryoshka architecture, and targeted interventions on ESMFold's structure predictions, we present the first work applying mechanistic interpretability to protein structure prediction. A key limitation is that our focus is exclusively on how PLM-derived sequence representations inform structure prediction. Looking ahead, we aim to establish formal connections between SAE sequence representations and equivariant representations of geometric data. Additional limitations and future directions can be found in Appendix B and C, respectively.

7 Appendix

A Related Work

Recent work has focused on mechanistically interpreting protein language models (PLMs) to understand how they achieve remarkable success in protein modeling and design. Early interpretability studies examined PLM internals, with Vig et al. (2021) and Rao et al. (2021) discovering that attention patterns encode structural relationships between amino acids. Later work extended these findings, showing attention could identify functionally important regions like allosteric sites (Kannan et al. (2024); Dong et al. (2024)).

Building on this attention-based analysis, Zhang et al. (2024) provided key mechanistic insights by showing that PLMs primarily learn by storing and looking up coevolutionary patterns preserved through evolution, rather than learning fundamental protein physics. By analyzing what sequence features influence contact predictions through a "categorical Jacobian" method, they demonstrated that PLMs memorize statistics of co-evolving residues analogous to classic evolutionary models.

Sparse autoencoders (SAEs) have emerged as a powerful new tool for interpreting these models. Simon and Zou (2024) developed InterPLM, extracting thousands of interpretable features from ESM-2 that correspond to known biological concepts like binding sites, structural motifs, and functional domains. Their work revealed that most biological concepts exist in superposition within the model's neurons, as SAEs vastly outperformed individual neurons at capturing these concepts. Concurrently, Adams et al. (2025) explored different SAE training approaches and evaluated their biological relevance through novel downstream tasks. In this work, we scale SAE training and evaluation to ESM2-3B and introduce Matryoshka SAEs (Bussmann et al. (2024); Nabeshima (2024)) with hierarchical encoding to steer ESMFold's structure prediction capabilities for the first time.

B Additional Limitations

Below we outline additional limitations of our work. First, as mentioned earlier, although we take an important step toward mechanistic interpretability of protein structure prediction, our focus is exclusively on how PLM-derived sequence representations inform structure prediction. The interpretability of ESMFold's structure prediction head is left for future investigation. Second, our interventions on ESMFold were restricted to single-feature manipulations on embeddings from individual layers rather than exploring combinations of features or cross-layer interactions. Third, our analysis is limited to the 8M and 3B models of ESM2; evaluating a broader range of model sizes could reveal whether SAE performance eventually plateaus. Fourth, ESMFold is no longer state-of-the-art compared to newer diffusion-based protein structure prediction methods Abramson et al. (2024); Wohlwend et al. (2024); Watson et al. (2023).

Fifth, recent work suggests that SAEs can learn different features even when trained on the same data, implying that our results may be sensitive to both initialization and the specific UniRef50 sample used (Paulo and Belrose (2025)). Sixth, our structural steering experiments focused primarily on surface accessibility, leaving other potential structural properties unexplored, and the observed hydrophobicity effects might be achievable through simpler approaches such as steering smaller PLMs or using supervised linear probes. Seventh, for Matryoshka SAEs, we do not report how the number of groups and group size choices impact results. Finally, although our ablation studies demonstrate preservation of structural information, we have not fully characterized how interactions among SAE features might affect ESMFold's folding mechanism. We anticipate that future work leveraging our open-source models developed with substantial computational resources will address these limitations and yield further insights for the community.

C Future Work

Connections to Geometric Representations. This work, while taking a significant step towards interpreting protein structure prediction, focuses primarily on learning linear representations of sequence information. Future work could establish formal connections between SAE sequence representations and equivariant representations of geometric data (Thomas et al., 2018; Geiger and Smidt, 2022; Lee et al., 2023), particularly investigating how the linear overcomplete basis learned by SAEs map to irreducible representations of geometric transformations in the context of protein structure prediction.

Theoretical Analysis of Matryoshka SAEs. Additionally, theoretical analysis drawing from ordered autoencoding (Rippel et al., 2014; Xu et al., 2021) and information bottleneck methods (Tishby et al., 2000) applied to protein structure prediction could improve our understanding of multi-scale feature learning in biological sequences. Such analysis may provide deeper insights into how hierarchical feature representations emerge and interact within the context of protein language models.

D Matryoshka SAE Implementation Details

The Matryoshka SAE architecture builds upon previous ones through nested groups of increasing size. This progressive constraint naturally encourages the emergence of a feature hierarchy - earlier groups must capture high-level, abstract features to achieve reasonable reconstruction with limited capacity, while later groups can encode more granular details. The key innovation lies in their group-wise decoding process, where each group must reconstruct the input using only its allocated subset of latents.

When processing a protein token embedding $x \in \mathbb{R}^d$ from a PLM, the encoding process follows as:

$$z = \text{BatchTopK}(W_{\text{enc}}x + b_{\text{enc}}) \tag{1}$$

where $W_{\text{enc}} \in \mathbb{R}^{n \times d}$ is the encoder weight matrix, $b_{\text{enc}} \in \mathbb{R}^n$ is the encoder bias, and BatchTopK enforces sparsity by keeping only the $B * K$ highest activations in each batch. This enforces average sparsity while allowing the number of active latents per token to vary.

The decoding process is:

$$\hat{x}^{(m)} = W_{\text{dec}}^{(m)} z_{[1:m]} + b_{\text{dec}} \tag{2}$$

Here, $z_{[1:m]}$ represents the first m components of the latent vector, $W_{\text{dec}}^{(m)} \in \mathbb{R}^{d \times m}$ is the decoder weight matrix restricted to its first m columns, and $b_{\text{dec}} \in \mathbb{R}^d$ is the decoder bias. Each group m must reconstruct the input using only its allocated subset of latents. The total loss combines reconstruction errors across all group sizes M:

$$\mathcal{L}(x) = \sum_{m \in M} \|x - \hat{x}^{(m)}\|_2^2 + \alpha \mathcal{L}_{\text{aux}} \tag{3}$$

where α weights any auxiliary regularization terms. We use the auxiliary loss term from Marks et al. (2024) and Gao et al. (2024) to reduce dead latents.

E Swiss-Prot Evaluation

E.1 Background of Swiss-Prot Evaluation

Swiss-Prot is the gold standard manually curated section of the UniProt Knowledge Base, containing expert-annotated protein sequences with detailed information about their structure, function, modifications, and key biological regions (Poux et al. (2017); Consortium (2024)). Through extensive experimental validation and literature review, Swiss-Prot provides high-quality annotations that span from individual catalytic sites to complete functional domains. Following Simon and Zou (2024), we evaluate whether learned features align

with these known biological concepts using a modified F1 score that handles the mismatch between precise feature activations and broader protein annotations present in Swiss-Prot. This approach calculates precision at the amino acid level but recall at the domain level, enabling systematic comparison between learned features and known biological concepts.

E.2 Performance of SAE Models on F1 Concept Evaluation

Model Scale vs Architecture: At 3B scale, architectural differences have minimal impact, with both Matryoshka and TopK identifying 233 concepts and showing only small variations in high-quality feature-concept pairs (2,677 vs 2,461). At 8M scale, architectural differences are more pronounced though still modest, with TopK outperforming Matryoshka (95 vs 72 concepts). However, the scale effect dominates these architectural differences - both 3B variants substantially outperform both 8M variants across all concept types.

Category	Total Concepts	Concepts with F1 > 0.5 (%)			
		3B MatK	3B TopK	8M MatK	8M TopK
Domains	256	193 (75.4%)	195 (76.2%)	50 (19.5%)	72 (28.1%)
Regions	55	19 (34.5%)	20 (36.4%)	10 (18.2%)	13 (23.6%)
Motifs	37	6 (16.2%)	5 (13.5%)	2 (5.4%)	3 (8.1%)
Zinc Fingers	15	5 (33.3%)	5 (33.3%)	2 (13.3%)	2 (13.3%)
Other Features	10	5 (50.0%)	3 (30.0%)	4 (40.0%)	1 (10.0%)
Targeting Peptides	5	5 (100.0%)	5 (100.0%)	4 (80.0%)	4 (80.0%)
Total	476	233 (48.9%)	233 (48.9%)	72 (15.1%)	95 (20.0%)

Table 1: Concept coverage comparison across model scales and architectures. We report the number (percentage) of concepts with F1 scores exceeding 0.5 for each model variant, broken down by concept category.

Figure 5: Distribution of highest F1 scores achieved for each concept across models.

The boxplot distribution reveals a clear separation between model scales (5. The 3B models show broader distributions with medians around 0.4 and numerous concepts achieving scores above 0.6. In contrast, 8M models show compressed distributions with medians below 0.1, indicating substantially weaker concept capture across the board.

The comparison heatmaps demonstrate relative performance between model variants by comparing their best-performing features for each concept. The left heatmap shows the number of concepts where each row model achieves a higher F1 score than the column model, while the right heatmap shows the average F1 score difference (Figure 6). The 3B models

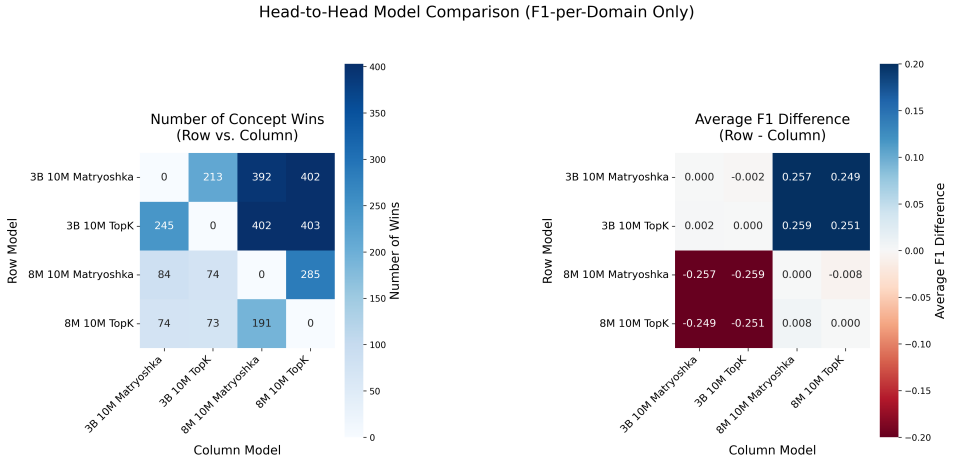


Figure 6: Left: Number of concepts where the row model achieves a higher maximum F1 score for a given concept than the column model. Right: Average score difference of the highest F1 scores for a given concept between models (row minus column) across all concepts. Each cell compares the best-performing feature for each concept between model pairs. Darker blue indicates stronger performance advantage for the row model.

outperform 8M variants on approximately 400 concepts, with an average F1 improvement of 0.25. Within each scale, architectural variants show minimal differences (average F1 difference < 0.01), suggesting model scale rather than architecture drives concept capture ability.

F Structure Prediction

F.1 CASP14 Dataset

For computational efficiency, we filtered the CASP14 dataset to exclude sequences longer than 700 residues to enable single-GPU evaluation. We further filtered out targets where ESMFold predictions showed RMSD $>$ 10 Å compared to the experimental structure, resulting in 17 of the original 34 targets

Data. We downloaded the CASP14 targets dataset from the CASP14 website here.

After filtering, we benchmark on these targets: [T1024, T1025, T1026, T1029, T1030, T1032, T1046s1, T1046s2, T1050, T1054, T1056, T1067, T1073, T1074, T1079, T1082, T1090].

F.2 Sparsity Sweep

We report the difference in RMSD on the above CASP14 dataset between ESMFold and SAE reconstructions across sparsity levels for Matryoshka SAEs. In Figure 7, we see that with only 2^3 to 2^5 active latents, SAEs can reasonably reconstruct structure prediction performance. These results raise questions on how much information is required per token for structure prediction.

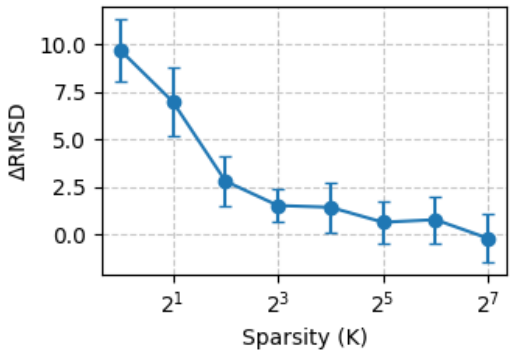


Figure 7: CASP14 ΔRMSD Results Across Sparsity Levels K for Matryoshka SAE Architecture

G Contact Map Prediction

Motivation. Zhang et al. (2024) introduced the Categorical Jacobian calculation to extract coevolutionary signals from protein language models in an unsupervised manner via masked language modeling. Given that SAEs also learn through unsupervised training, we adapt this approach for contact map prediction to evaluate whether SAEs preserve coevolutionary patterns and structural information learned by ESM2.

Data. We randomly sampled 50 proteins from the evaluation dataset in Zhang et al. (2024) due to computational constraints. As shown in Fig. 8, our results replicate the authors' findings, demonstrating superior contact map accuracy across nearly the entire sample. Based on these consistent results, we believe our conclusions would generalize to the complete dataset.

Hyperparameters. In Fig 3b, our SAEs were trained on layer 36 with TopK architecture with expansion factors 16x for ESM2-8M (dict size 10240) and 8x for ESM2-3B (dict size 20480). We did not rerun for the Matryoshka architecture given TopK's similar language modeling reconstruction performance and how computationally expensive it is.

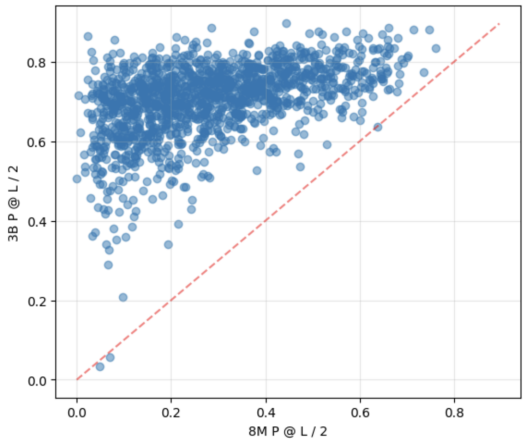


Figure 8: Contact Map Accuracy for Cat. Jacobian on ESM2 8M vs 3B, N=1412/1431

H ESMFold Steering Case Study

H.1 Feature Steering Background

Following the method described in Templeton et al. (2024), each target feature is represented by its corresponding decoder vector ($d_{(i)} = W_{dec}[i, :]$), which is normalized to unit norm during training, and an intervention is performed by adding a scaled version of this vector to the hidden state ($h_l \leftarrow h_l + \alpha \cdot d_{(i)}$) to amplify or suppress that feature's contribution. In our approach, we apply a maximum activation scaling factor during training and subsequently scale the encoder and decoder biases at inference. This means our effective steering coefficient, reported in all figures and results, is given by $\alpha' = \text{norm\_factor} \cdot \alpha$, ensuring consistent steering effects across different model configurations.

H.2 Details of Feature Steering Case Study

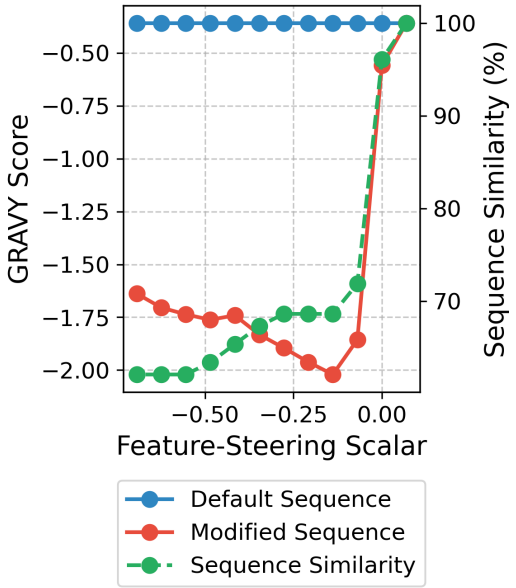


Figure 9: Plot of sequence GRAVY scores overlaid with sequence similarity with increased feature steering

Our intervention framework employed two complementary approaches: (1) examining sequence-level changes through standard ESMFold predictions, and (2) directly intervening on hidden representations while maintaining the true input sequence in an ablated version that isolated intervention effects. This design allowed us to dissect how structural information flows through the model. The ablated model showed minimal performance differences (Figure 2b).

The feature (f/2) was identified in the first group of a Matryoshka SAE trained on hidden layer 36 (dictionary size 10,240, group fractions [0.002, 0.0156, 0.125, 0.8574], sparsity k=20), showing strong correlation with residue hydrophobicity. We selected the Matryoshka SAE architecture as its hierarchical encoding tends to capture broader concepts in earlier groups, which we hypothesized would enable more robust steering across diverse proteins. We applied the steering methodology from Templeton et al. (2024), adding the decoder vector with coefficient α = -0.275. SASA was computed using FreeSASA Mitternacht (2016) with default parameters. The observed RMSD of 2.76 Å aligns with typical deviations for ESMFold with layer 36 LM head ablation, while ablating other ESM2 layer representations resulted in RMSD of 0.191 compared to regular ESMFold (Figure 2b).

As shown in Figure 9, steering progressively decreased sequence GRAVY scores, indicating a shift toward hydrophilic residues. When modified sequences were processed through default ESMFold, we observed even larger SASA increases, consistent with replacing hydrophobic core residues with hydrophilic ones. This effect manifested through two distinct mechanisms: direct biasing of the structure module toward more exposed conformations (even with correct sequence information), and shifting sequence predictions toward more hydrophilic residues when allowed to influence the sequence module. Notably, structural changes persisted even when preserving the original sequence, suggesting ESMFold's predictions are influenced by hidden representations corresponding to the modified sequence despite having access to correct sequence information.

H.3 Validation of Case Study Feature Steering Directionality and Specificity

Scalar	Modified SASA	SASA Change	Modified GRAVY	GRAVY Change
0.069	8369.55	0.00%	-0.3588	0.00%
0.208	8351.58	-0.21%	-0.0595	+83.42%
0.346	8568.43	+2.38%	0.6569	+283.08%
0.554	8616.49	+2.95%	1.3046	+463.60%
0.693	9719.28	+16.13%	1.5255	+525.17%

Table 2: Positive steering control results showing changes in SASA and GRAVY scores relative to baseline values.

Feature	Metric	Scalar Values		
		-0.069	-0.346	-0.693
Feature 5 (Group 1)	δSASA	0.00%	+0.38%	-0.54%
	δGRAVY	0.00%	+40.40%	+52.34%
Feature 39 (Group 2)	δSASA	0.00%	0.00%	0.00%
Feature 1381 (Group 3)	δSASA	0.00%	0.00%	0.00%
Feature 7921 (Group 4)	δSASA	0.00%	0.00%	0.00%
	δGRAVY	0.00%	0.00%	0.00%
Original Experiment	δSASA	+128.44%	+137.12%	+112.89%
	δGRAVY	-417.34%	-446.21%	-356.69%

Table 3: Comparison of random feature controls with original experiment, showing percentage changes relative to unsteered (scalar=0) values.

We performed positive steering control experiments to validate the directional behavior of our feature, applying scalar values from 0.069 to 0.693. As predicted, steering the feature positively resulted in a systematic increase in hydrophobicity (GRAVY score) while maintaining structural stability at moderate steering strengths. The modified sequence GRAVY scores increased from -0.36 to 1.53, reflecting enhanced hydrophobic content, while SASA values remained stable ($\leq 3\%$ change) until the highest scalar values where minor structural perturbations emerged.

Additionally, to assess the specificity of identified feature, we performed negative steering experiments on randomly selected features from each Matryoshka group (features 5, 39, 1381, and 7921). Unlike our target feature, random features showed minimal response to steering, with three features showing no changes ($\leq 0.1\%$ variation) across all metrics and one feature (Feature 5) showing only minor variations incomparable to the dramatic shifts observed in our main experiment.

I Implementation Details

I.1 SAE Implementation Details

For each token of a protein sequence, a transformer encoder-based PLM outputs an embedding x in each layer ℓ. To learn meaningful representations, sparsity constraints are imposed on the latent vector z. This sparsity can be achieved either through L1 regularization terms in the loss function or through specialized activation functions that directly restrict the number of non-zero elements. In practice, the number of active (non-zero) elements K satisfies $K \ll d \ll n$, preventing degenerate solutions such as the identity map.

Training and Normalization. During training, we normalize our embeddings to have average unit L2 norm, enabling hyperparameter transfer between layers. This normalization is reversed during inference by scaling the biases according to the procedure detailed in Marks et al. (2024). This approach ensures consistent performance across different layers while maintaining the model's representational capacity.

I.2 Code

For all SAE implementations, we use Marks et al. (2024). For our L1 regularized SAEs, we follow Conerly et al. (2024). For TopK, we follow Gao et al. (2024). In Fig 2a, we standardize hyperparameters with an expansion factor of 8x (dict size 20480) and batch size 2048 and use best learning rate.

I.3 Data Curation

Following the approach outlined in the InterPLM paper, we first process the UniRef50 FASTA file by iterating over each protein sequence. We remove any sequence that exceeds 1022 tokens, as ESM-2 cannot process longer inputs. From the remaining sequences, we randomly select a predetermined number of proteins to create our working dataset. This dataset is then divided into shards of 1000 sequences each, facilitating efficient handling during training.

I.4 Optimizing the Dataset and Training Procedure

After generating embeddings with ESM-2, the activations are stored as tensors representing entire shards. We then use Litdata's optimize function to iterate through each tensor and split it into fixed-size batches. The optimized dataset achieves speedup by reducing streaming I/O overhead to AWS S3 during training. Additionally, the optimized dataset uses Litdata's StreamingDataset and StreamingDataloader where we shuffle the batches during training, while also leveraging PyTorch Lightning for multi-GPU training support.

References

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, August 2021. ISSN 1476-4687. doi: 10.1038/s41586-021-03819-2. URL https://doi.org/10.1038/s41586-021-03819-2.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic level protein structure with a language model. *bioRxiv*, 2022. doi: 10.1101/2022.07.20.500902. URL https://www.biorxiv.org/content/early/2022/12/21/2022.07.20.500902.

Debora S. Marks, Lucy J. Colwell, Robert Sheridan, Thomas A. Hopf, Andrea Pagnani, Riccardo Zecchina, and Chris Sander. Protein 3d structure computed from evolutionary sequence variation. *PLOS ONE*, 6(12):1–20, 12 2011. doi: 10.1371/journal.pone.0028766. URL https://doi.org/10.1371/journal.pone.0028766.

Hetunandan Kamisetty, Sergey Ovchinnikov, and David Baker. Assessing the utility of coevolution-based residue–residue contact predictions in a sequence- and structure-rich era. *Proceedings of the National Academy of Sciences*, 110(39):15674–15679, 2013. doi: 10.1073/pnas.1314045110. URL https://www.pnas.org/doi/abs/10.1073/pnas.1314045110.

Roshan Rao, Joshua Meier, Tom Sercu, Sergey Ovchinnikov, and Alexander Rives. Transformer protein language models are unsupervised structure learners. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=fylclEqgvgd.

Jesse Vig, Ali Madani, Lav R. Varshney, Caiming Xiong, Richard Socher, and Nazneen Fatema Rajani. Bertology meets biology: Interpreting attention in protein language models, 2021. URL https://arxiv.org/abs/2006.15222.

Zhidian Zhang, Hannah K. Wayment-Steele, Garyk Brixi, Haobo Wang, Dorothee Kern, and Sergey Ovchinnikov. Protein language models learn evolutionary statistics of interacting sequence motifs. *Proceedings of the National Academy of Sciences*, 121(45):e2406285121, 2024. doi: 10.1073/pnas.2406285121. URL https://www.pnas.org/doi/abs/10.1073/pnas.2406285121.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L Turner, Callum McDougall, Monte MacDiarmid, C. Daniel Freeman, Theodore R. Sumers, Edward Rees, Joshua Batson, Adam Jermyn, Shan Carter, Chris Olah, and Tom Henighan. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*, 2024. URL https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html.

Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders, 2024. URL https://arxiv.org/abs/2406.04093.

Elana Simon and James Zou. Interplm: Discovering interpretable features in protein language models via sparse autoencoders. *bioRxiv*, 2024. doi: 10.1101/2024.11.14.623630. URL https://www.biorxiv.org/content/early/2025/01/28/2024.11.14.623630.

Etowah Adams, Liam Bai, Minji Lee, Yiyang Yu, and Mohammed AlQuraishi. From mechanistic interpretability to mechanistic biology: Training, evaluating, and interpreting sparse autoencoders on protein language models. *bioRxiv*, 2025. doi: 10.1101/2025.02.06.636901. URL https://www.biorxiv.org/content/early/2025/02/08/2025.02.06.636901.

Noa Nabeshima. Matryoshka sparse autoencoders, December 2024. URL https://www.lesswrong.com/posts/zbebxYCqsryPALh8C/matryoshka-sparse-autoencoders. AI Alignment Forum, Accessed: Feb 10, 2025.

Bart Bussmann, Patrick Leask, and Neel Nanda. Learning multi-level features with matryoshka saes, December 2024. URL https://www.lesswrong.com/posts/rKM9b6B2LqwSB5ToN/learning-multi-level-features-with-matryoshka-saes#What_are_Matryoshka_SAEs__. AI Alignment Forum, Accessed: Feb 10, 2025.

Samuel Marks, Adam Karvonen, and Aaron Mueller. dictionary_learning package. https://github.com/saprmarks/dictionary_learning, 2024.

Gokul R. Kannan, Brian L. Hie, and Peter S. Kim. Single-sequence, structure free allosteric residue prediction with protein language models. *bioRxiv*, 2024. doi: 10.1101/2024.10.03.616547. URL https://www.biorxiv.org/content/early/2024/10/03/2024.10.03.616547.

Tianze Dong, Christopher Kan, Kapil Devkota, and Rohit Singh. Allo-allo: Data-efficient prediction of allosteric sites. *bioRxiv*, 2024. doi: 10.1101/2024.09.28.615583. URL https://www.biorxiv.org/content/early/2024/09/30/2024.09.28.615583.

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J. Ballard, Joshua Bambrick, Sebastian W. Bodenstein, David A. Evans, Chia-Chun Hung, Michael O'Neill, David Reiman, Kathryn Tunyasuvunakool, Zachary Wu, Akvilė Žemgulytė, Eirini Arvaniti, Charles Beattie, Ottavia Bertolli, Alex Bridgland, Alexey Cherepanov, Miles Congreve, Alexander I. Cowen-Rivers, Andrew Cowie, Michael Figurnov, Fabian B. Fuchs, Hannah Gladman, Rishub Jain, Yousuf A. Khan, Caroline M. R. Low, Kuba Perlin, Anna Potapenko, Pascal Savy, Sukhdeep Singh, Adrian Stecula, Ashok Thillaisundaram, Catherine Tong, Sergei Yakneen, Ellen D. Zhong, Michal Zielinski, Augustin Žídek, Victor Bapst, Pushmeet Kohli, Max Jaderberg, Demis Hassabis, and John M. Jumper. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, June 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-07487-w. URL https://doi.org/10.1038/s41586-024-07487-w.

Jeremy Wohlwend, Gabriele Corso, Saro Passaro, Mateo Reveiz, Ken Leidal, Wojtek Swiderski, Tally Portnoi, Itamar Chinn, Jacob Silterra, Tommi Jaakkola, and Regina Barzilay. Boltz-1: Democratizing biomolecular interaction modeling. *bioRxiv*, 2024. doi: 10.1101/2024.11.19.624167. URL https://www.biorxiv.org/content/early/2024/12/27/2024.11.19.624167.

Joseph L. Watson, David Juergens, Nathaniel R. Bennett, Brian L. Trippe, Jason Yim, Helen E. Eisenach, Woody Ahern, Andrew J. Borst, Robert J. Ragotte, Lukas F. Milles, Basile I. M. Wicky, Nikita Hanikel, Samuel J. Pellock, Alexis Courbet, William Sheffler, Jue Wang, Preetham Venkatesh, Isaac Sappington, Susana Vázquez Torres, Anna Lauko, Valentin De Bortoli, Emile Mathieu, Sergey Ovchinnikov, Regina Barzilay, Tommi S. Jaakkola, Frank DiMaio, Minkyung Baek, and David Baker. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976):1089–1100, August 2023. ISSN 1476-4687. doi: 10.1038/s41586-023-06415-8. URL https://doi.org/10.1038/s41586-023-06415-8.

Gonçalo Paulo and Nora Belrose. Sparse autoencoders trained on the same data learn different features, 2025. URL https://arxiv.org/abs/2501.16615.

Nathaniel Thomas, Tess E. Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation- and translation-equivariant neural networks for 3d point clouds. *CoRR*, abs/1802.08219, 2018. URL http://arxiv.org/abs/1802.08219.

Mario Geiger and Tess Smidt. e3nn: Euclidean neural networks, 2022. URL https://arxiv.org/abs/2207.09453.

Jae Hyeon Lee, Payman Yadollahpour, Andrew Watkins, Nathan C. Frey, Andrew Leaver-Fay, Stephen Ra, Kyunghyun Cho, Vladimir Gligorijević, Aviv Regev, and Richard Bonneau. Equifold: Protein structure prediction with a novel coarse-grained structure representation. *bioRxiv*, 2023. doi: 10.1101/2022.10.07.511322. URL https://www.biorxiv.org/content/early/2023/01/02/2022.10.07.511322.

Oren Rippel, Michael A. Gelbart, and Ryan P. Adams. Learning ordered representations with nested dropout, 2014. URL https://arxiv.org/abs/1402.0915.

Yilun Xu, Yang Song, Sahaj Garg, Linyuan Gong, Rui Shu, Aditya Grover, and Stefano Ermon. Anytime sampling for autoregressive models via ordered autoencoding. *CoRR*, abs/2102.11495, 2021. URL https://arxiv.org/abs/2102.11495.

Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method, 2000. URL https://arxiv.org/abs/physics/0004057.

Sylvain Poux, Cecilia N Arighi, Michele Magrane, Alex Bateman, Chih-Hsuan Wei, Zhiyong Lu, Emmanuel Boutet, Hema Bye-A-Jee, Maria Livia Famiglietti, Bernd Roechert, and The UniProt Consortium. On expert curation and scalability: Uniprotkb/swiss-prot as a case study. *Bioinformatics*, 33(21):3454–3460, 07 2017. ISSN 1367-4803. doi: 10.1093/bioinformatics/btx439. URL https://doi.org/10.1093/bioinformatics/btx439.

The UniProt Consortium. Uniprot: the universal protein knowledgebase in 2025. *Nucleic Acids Research*, 53(D1):D609–D617, 11 2024. ISSN 1362-4962. doi: 10.1093/nar/gkae1010. URL https://doi.org/10.1093/nar/gkae1010.

Simon Mitternacht. Freesasa: An open source c library for solvent accessible surface area calculations. *F1000Research*, 5:189, February 2016. ISSN 2046-1402. doi: 10.12688/f1000research.7931.1. URL http://dx.doi.org/10.12688/f1000research.7931.1.

Tom Conerly, Adly Templeton, Trenton Bricken, Jonathan Marcus, and Tom Henighan. April 2024 update on transformer circuits. *Transformer Circuits*, 2024. URL https://transformer-circuits.pub/2024/april-update/index.html.