ProtSAE: Disentangling and Interpreting Protein Language Models via Semantically-Guided Sparse Autoencoders

Xiangyu Liu[1], **Haodi Lei**[1], **Yi Liu**[1], **Yang Liu**[1], **Wei Hu**[1,2,*]

[1] State Key Laboratory for Novel Software Technology, Nanjing University, China
[2] National Institute of Healthcare Data Science, Nanjing University, China
{xyl.nju, haodilei, yiliu07.nju, yliu20.nju}@gmail.com, whu@nju.edu.cn

Abstract

Sparse Autoencoder (SAE) has emerged as a powerful tool for mechanistic interpretability of large language models. Recent works apply SAE to protein language models (PLMs), aiming to extract and analyze biologically meaningful features from their latent spaces. However, SAE suffers from semantic entanglement, where individual neurons often mix multiple nonlinear concepts, making it difficult to reliably interpret or manipulate model behaviors. In this paper, we propose a semantically-guided SAE, called ProtSAE. Unlike existing SAE which requires annotation datasets to filter and interpret activations, we guide semantic disentanglement during training using both annotation datasets and domain knowledge to mitigate the effects of entangled attributes. We design interpretability experiments showing that ProtSAE learns more biologically relevant and interpretable hidden features compared to previous methods. Performance analyses further demonstrate that ProtSAE maintains high reconstruction fidelity while achieving better results in interpretable probing. We also show the potential of ProtSAE in steering PLMs for downstream generation tasks.

Introduction

In recent years, protein language models (PLMs) (Lin et al. 2023a; Nijkamp et al. 2023) have developed rapidly and been widely applied to downstream tasks including protein function prediction (Lin et al. 2024), structural modeling (Lin et al. 2023b), and protein design (Ferruz and Höcker 2022). However, the internal mechanisms of PLMs remain largely unknown (Garcia and Ansuini 2025).

For protein engineering, it is important to understand how latent features map to biological concepts, such as binding pockets, post-translational modifications, or fold families. Such analysis facilitates the identification of spurious correlations and biases, enhancing both performance and robustness of PLMs. Moreover, it allows for the extraction of latent relationships among protein characteristics, providing meaningful insights that can inform and support biological research (Zhang et al. 2024). Early works have attempted to analyze protein models, exploring the relationships between attention mechanisms and amino acids (Vig et al. 2021), as well as identifying neurons associated with certain biological concepts (Nori, Singireddy, and Have 2023a).

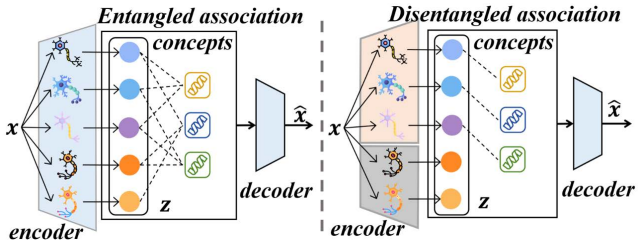


Figure 1: Illustration of SAE semantic entanglement (left): individual neurons conflate multiple biological concepts, and semantic disentanglement (right): each defined neuron maps to a single biological concept.

Recent studies (Simon and Zou 2024; Garcia and Ansuini 2025; Adams et al. 2025) have begun applying sparse autoencoder (SAE) to PLMs and observed the emergence of features associated with various biological concepts. SAE is an effective tool for understanding and explaining the internal representations of PLMs. Based on the assumption of linear feature superposition (Yun et al. 2021), it decomposes the hidden representations to extract sparse features. These features can be further analyzed for their correlations with specific concepts, enabling interpretability. Furthermore, the sparse features can be selectively activated to steer the generation along the directions of relevant concepts.

A typical SAE requires an annotation dataset after training to interpret the learned features (Simon and Zou 2024). This annotation contains concepts of interest, and post-hoc correlation analysis is often performed to establish the relationship between the features in SAE and these concepts. However, SAE suffers from the problem of semantic entanglement: *individual neurons often conflate multiple concepts* (Joshi et al. 2025). This entanglement results in ambiguous interpretations of the learned features. As illustrated in Figure 1, each neuron is likely to be simultaneously associated with multiple, semantically divergent concepts. Consequently, identifying the true meaning of the given feature and using it to steer the generation becomes challenging, undermining the interpretability of the model.

In this paper, we propose ProtSAE, which incorporates se-

*Corresponding author

mantic guidance into the SAE training to disentangle semantic features. First, we leverage the semantic annotations used for post-hoc interpretation of SAE features to constrain the relationship between defined activations and specific concepts during training. We also use forced activations and feature rescaling to ensure that the defined activations effectively participate in reconstruction with high fidelity. Second, considering the rich prior knowledge in the protein domain, where concepts are not mutually independent, we incorporate ELEmbeddings (Kulmanov et al. 2019) to model potential logical constraints among concepts, e.g., subsumption and conjunction. The constraints are integrated into the training process of ProtSAE to enhance the interpretability and semantic consistency of the learned features.

We construct interpretability experiments to demonstrate that ProtSAE effectively captures biologically meaningful features in PLMs, such as molecular functions, biological processes, and binding sites. Compared with features annotated from the typical SAE, the defined neurons in ProtSAE learn more accurate, disentangled representations that are more tightly aligned with protein structures. Furthermore, performance analyses show that ProtSAE preserves richer semantic information related to protein concepts. Under varying levels of sparsity, it consistently achieves stronger performance on protein function prediction (Kulmanov et al. 2024; Kulmanov and Hoehndorf 2022) while maintaining high reconstruction fidelity. Finally, through targeted activation steering across various biological concepts, we demonstrate that the semantic features learned by ProtSAE can effectively guide PLM outputs toward desired functional outcomes—validating both the quality of the learned representations and their potential for precise model control.

The main contributions of this paper are listed as follows:

- We propose ProtSAE, a novel semantically-guided SAE that can disentangle complex protein features, yielding features in PLMs more strongly aligned with biological concepts. See https://github.com/nju-websoft/ProtSAE.
- We introduce protein domain knowledge into ProtSAE training to learn the logical constraints among concepts, and apply forced activations and feature rescaling to ensure that defined activations effectively participate in reconstruction with high fidelity.
- We conduct extensive experiments and analyses. Interpretability experiments demonstrate that ProtSAE captures more interpretable features that are closely aligned with biological concepts. Detailed performance analyses show that ProtSAE consistently outperforms baselines across varying levels of sparsity while maintaining high reconstruction fidelity. Steering experiments reveal that ProtSAE enables effective interventions across diverse biological concepts.

Related Work

Mechanistic interpretability and SAE. Mechanistic interpretability aims to understand how neural networks produce outputs based on the internal algorithms that they have learned (Olah et al. 2020). Previous works explore the computation subgraphs responsible for specific tasks (Shi et al. 2024; Dunefsky, Chlenski, and Nanda 2024; Wang et al. 2023), and analyze the behaviors within large language models (LLMs) (Makelov 2024; Miller, Chughtai, and Saunders 2024; Makelov et al. 2024). One prominent line of analysis focuses on identifying and studying sparse linear features within LLMs (Todd et al. 2024). Based on the assumption of linear feature superposition (Elhage et al. 2022), some works decompose language model activations and use them to intervene in the model's behavior (Yun et al. 2021; Tamkin, Taufeeque, and Goodman 2024). Recent scaling efforts demonstrate the viability of SAE across LLMs, from Claude 3 Sonnet (Paulo et al. 2024) to GPT-4 (Gao et al. 2024), with extensions to multi-modal LLMs as well (Pach et al. 2025). Several works propose architectural improvements to SAE to mitigate feature shrinkage and improve reconstruction fidelity (Wright and Sharkey 2024; Rajamanoharan et al. 2024). Others focus on developing comprehensive evaluation frameworks for SAE (Gallifant et al. 2025) and exploring generating more informative explanations for activated features with additional datasets or LLMs (Wu et al. 2025a,b).

Interpretability in PLMs. Early works in PLMs show that attention maps can capture structural and functional signals, including amino acid interactions (Vig et al. 2021), protein contacts (Rao et al. 2021), and functional sites like binding pockets and allosteric regions (Kannan, Hie, and Kim 2024; Dong et al. 2024). CB-pLM (Ismail et al. 2025) trains PLMs with a concept bottleneck layer for better understanding and controlling PLMs' generation. Recent studies explore how high-level conceptual knowledge is internally represented in the components of PLMs (Nori, Singireddy, and Have 2023b). SAE is used to decompose latent activations and reveal links between biological concepts and structural features (Simon and Zou 2024; Garcia and Ansuini 2025; Adams et al. 2025; Gujral et al. 2025). These studies also show that editing related activations can influence or steer protein sequence generation. However, features from SAE often entangle multiple concepts, making interpretation unclear. To address this, we introduce semantic guidance during the SAE training by linking specific activations to biological concepts, leading to more interpretable features.

Overview

SAE architectures. SAE is designed to learn sparse representations of high-dimensional inputs by encouraging only a small subset of neurons to be activated. It is widely used to extract interpretable and localized features, particularly in the context of mechanistic interpretability for deep models.

Given an input vector $\mathbf{x} \in \mathbb{R}^d$, the encoder maps it to the latent activations $\mathbf{z} \in \mathbb{R}^n$ using a linear transformation followed by the ReLU activation:

$$\mathbf{z} = \text{ReLU}(\mathbf{W}_{\text{enc}}(\mathbf{x} - \mathbf{b}_{\text{dec}}) + \mathbf{b}_{\text{enc}}), \tag{1}$$

where $\mathbf{W}_{\text{enc}} \in \mathbb{R}^{n \times d}$, $\mathbf{b}_{\text{enc}} \in \mathbb{R}^n$, and $\mathbf{b}_{\text{dec}} \in \mathbb{R}^d$ are learnable parameters. To enable sparsity in $\mathbf{z}$, n is typically set much larger than d ($n \gg d$). The decoder reconstructs the input via a linear transformation:

$$\hat{\mathbf{x}} = \mathbf{W}_{\text{dec}}\mathbf{z} + \mathbf{b}_{\text{dec}}, \tag{2}$$

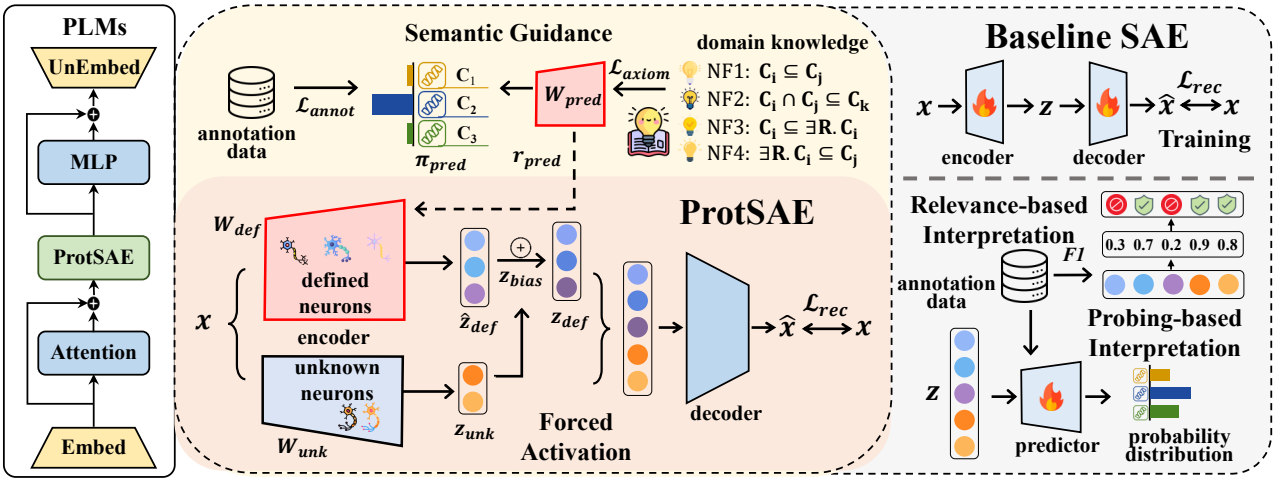


Figure 2: An overview of ProtSAE (left) and the baseline SAE (right). In the baseline SAE, annotation data is used post hoc to interpret learned features via relevance-based and probing-based methods. In contrast, ProtSAE incorporates semantic guidance during training by leveraging annotation data and protein domain knowledge to achieve semantic disentanglement. It also uses forced activations and feature rescaling to learn meaningful features while preserving reconstruction fidelity.

where $\mathbf{W}_{\text{dec}} \in \mathbb{R}^{d \times n}$. The training objective encourages accurate reconstruction and sparsity in $\mathbf{z}$:

$$\mathcal{L} = \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 + \lambda \|\mathbf{z}\|_1, \tag{3}$$

where $\|\mathbf{x} - \hat{\mathbf{x}}\|_2^2$ is the reconstruction MSE loss, $\|\mathbf{z}\|_1$ imposes an L1 penalty to encourage sparsity, and λ is a tunable hyperparameter to balance the two terms.

Interpreting SAE activations. As shown in Figure 2, we follow prior works and use an annotation dataset that maps proteins to biological concepts to interpret the learned SAE features via two approaches: (1) *Relevance-based interpretation.* Following previous work (Garcia and Ansuini 2025), we compute the activation levels of each feature across annotated proteins. Based on the annotation, we calculate a relevance score (e.g., F1-score) between a feature and a target concept, and use the relevant concept to interpret the feature. Detailed formulations are described in the appendix. (2) *Probing-based interpretation.* We train linear probing classifiers on SAE activations using the annotation dataset (Simon and Zou 2024; Gurnee et al. 2023). It aims to detect the presence of specific concepts within the learned features through supervised training.

Method

Conventionally, the encoder of SAE serves two purposes: (1) *Determining which features should be active.* To disentangle semantics in activated features, each defined neuron should be selectively activated only by proteins associated with a specific concept, while remaining inactive for unrelated protein sequences. This constraint helps prevent entangled semantics within the same neuron. Based on this intuition, we introduce semantic guidance from the annotation data and domain knowledge to learn such semantically selective activations during training. (2) *Estimating the magnitude of active features to support faithful reconstruction.* Although the magnitude of each active feature should be determined by the reconstruction training, we must ensure that the feature directions associated with the predefined concepts effectively contribute to reconstruction. This guarantees that steering the encoder's activation yields consistent and interpretable effects on the model's behavior.

Guiding SAE with Annotation Data

We adopt the TopK-SAE as the backbone, where only the top-K neurons with the highest activations are used in reconstruction. We define the encoder as follows:

$$\mathbf{z} = \text{TopK}\big(\mathbf{W}_{\text{enc}}(\mathbf{x} - \mathbf{b}_{\text{dec}}) + \mathbf{b}_{\text{enc}}\big), \tag{4}$$

where $\mathbf{W}_{\text{enc}} \in \mathbb{R}^{n \times d}$, $\mathbf{b}_{\text{enc}} \in \mathbb{R}^n$, and $\mathbf{b}_{\text{dec}} \in \mathbb{R}^d$. $\text{TopK}(\cdot)$ retains only the top K largest activations, zeroing out the rest. Suppose that there are m defined concepts of interest. We aim for m corresponding activations $\mathbf{z}_{\text{def}}$ to accurately represent these concepts, while retaining the remaining $n - m$ activations $\mathbf{z}_{\text{unk}}$ to capture unknown semantic concepts. We partition the activation $\mathbf{z}$, weight matrix $\mathbf{W}_{\text{enc}}$, and $\mathbf{b}_{\text{enc}}$ into two components:

$$\mathbf{z} = \begin{bmatrix} \mathbf{z}_{\text{def}} \\ \mathbf{z}_{\text{unk}} \end{bmatrix}, \mathbf{W}_{\text{enc}} = \begin{bmatrix} \mathbf{W}_{\text{def}} \\ \mathbf{W}_{\text{unk}} \end{bmatrix}, \mathbf{b}_{\text{enc}} = \begin{bmatrix} \mathbf{b}_{\text{def}} \\ \mathbf{b}_{\text{unk}} \end{bmatrix}, \tag{5}$$

where $\mathbf{W}_{\text{def}} \in \mathbb{R}^{m \times d}$ and $\mathbf{b}_{\text{def}} \in \mathbb{R}^m$ corresponds to m defined activations $\mathbf{z}_{\text{def}}$ aligned with predefined concepts, $\mathbf{W}_{\text{unk}} \in \mathbb{R}^{(n-m) \times d}$ and $\mathbf{b}_{\text{unk}} \in \mathbb{R}^{n-m}$ capture the remaining activations $\mathbf{z}_{\text{unk}}$.

Semantic disentanglement. To guide the semantic disentanglement of specific activations, we introduce a concept predictor. It learns to estimate the presence of each predefined concept from the input. Let $\mathbf{W}_{\text{pred}} \in \mathbb{R}^{m \times d}$ denote its weight matrix, the prediction probability of defined activations is computed as follows:

$$\pi_{\text{pred}} = \sigma\big(\mathbf{W}_{\text{pred}}(\mathbf{x} - \mathbf{b}_{\text{dec}}) + \mathbf{b}_{\text{pred}}\big) \in (0, 1)^m, \tag{6}$$

where $\sigma(\cdot)$ is the sigmoid function and $\mathbf{b}_{\text{pred}} \in \mathbb{R}^m$ is the prediction bias. We train this predictor on the available annotation data using a binary cross-entropy loss:

$$\mathcal{L}_{\text{annot}} = \text{CrossEntropy}(\pi_{\text{pred}}, y), \tag{7}$$

where $y \in \{0,1\}^m$ is the binary annotation vector indicating which semantic concepts are present.

We assume that $\mathbf{W}_{\text{def}}$ (used for reconstruction) and $\mathbf{W}_{\text{pred}}$ (used for prediction) encode the same underlying semantic meanings. Thus, they should share the same projection directions. To achieve this, we treat $\mathbf{W}_{\text{def}}$ as a rescaled version of $\mathbf{W}_{\text{pred}}$, and tie their weights as follows:

$$\mathbf{W}_{\text{def}} = \mathbf{W}_{\text{pred}}^{\text{detach}} \cdot \exp(\mathbf{r}_{\text{pred}}), \tag{8}$$

where $\mathbf{r}_{\text{pred}} \in \mathbb{R}^m$ is a learnable scaling vector and $\cdot$ denotes row-wise multiplication, where each row i of $\mathbf{W}_{\text{pred}}^{\text{detach}}$ is scaled by $\exp(\mathbf{r}_{\text{pred}}[i])$. The `detach` indicates that gradients from the reconstruction loss are prevented from updating $\mathbf{W}_{\text{pred}}$. This formulation ensures that $\mathbf{W}_{\text{def}}$ retains the semantic directionality learned from supervision, while its magnitude can adapt to improve reconstruction. The exponential guarantees positivity and allows smooth multiplicative modulation.

Forced activation. Using the encoder, we compute the semantic and unsupervised activations as

$$\mathbf{z}_{\text{unk}} = \text{TopK}\big(\mathbf{W}_{\text{unk}}(\mathbf{x} - \mathbf{b}_{\text{dec}}) + \mathbf{b}_{\text{unk}}\big), \tag{9}$$

$$\hat{\mathbf{z}}_{\text{def}} = \mathbf{W}_{\text{def}}(\mathbf{x} - \mathbf{b}_{\text{dec}}) + \mathbf{b}_{\text{def}}, \tag{10}$$

where $\hat{\mathbf{z}}_{\text{def}}$ denotes the pre-activation output for the defined concept neurons before any sparsity constraints. In practice, we observe that the reconstruction tends to rely more on the entangled, unsupervised activations $\mathbf{z}_{\text{unk}}$, which diminishes the contribution of concept-specific activations $\mathbf{z}_{\text{def}}$. To mitigate this issue, we introduce a semantic bias that encourages the activations associated with predicted concepts to contribute more strongly to reconstruction:

$$\mathbf{z}_{\text{bias}} = \mathbb{1}_{\pi_{\text{pred}}>0.5} \cdot \text{ReLU}\big(\text{mean}(\mathbf{z}_{\text{unk}}) - \hat{\mathbf{z}}_{\text{def}}\big), \tag{11}$$

$$\mathbf{z}_{\text{def}} = \hat{\mathbf{z}}_{\text{def}} + \mathbf{z}_{\text{bias}}. \tag{12}$$

Here, $\mathbb{1}_{\pi_{\text{pred}}>0.5} \in \{0,1\}^m$ denotes an indicator function, marking whether each semantic concept is predicted to be present (i.e., $\pi_{\text{pred}} > 0.5$) and $\cdot$ denotes element-wise multiplication. For such concepts, we enforce the corresponding activation to be no less than the average activation of $\mathbf{z}_{\text{unk}}$, preventing the model from ignoring semantically meaningful features during reconstruction. This bias effectively forces the features aligned with known semantics to participate in encoding, enhancing both interpretability and task relevance.

Guiding SAE with Domain Knowledge

In protein sequence modeling, biological concepts are often semantically interdependent. The relationship of some concepts can be defined with logical constraints including "is-a", "part-of", "regulates", and other relations. These axioms establish a stable, expert-curated structure over biological knowledge, dictating how concepts relate and compose. Therefore, aligning latent directions in SAE's hidden space with concept semantic relationships can enhance the detection and disentanglement of meaningful biological concepts.

To achieve this, we incorporate domain knowledge into SAE using ELEmbededings (Kulmanov et al. 2019). It is an ontology representation learning method, which represents each concept as a hypersphere in the embedding space, and encodes logical axioms as constraints on the positions and relationships between these regions. Given a concept c_i, the prediction probability of a protein p can be modeled using ELEmbeddings as

$$y_i' = \sigma\left(f_\eta(p)^\top \cdot \big(f_\eta(hF) + f_\eta(c_i)\big) + r_\eta(c_i)\right), \tag{13}$$

where $f_\eta(\cdot)$ is the projection function into the semantic embedding space, hF is the hasFunction relation, and $r_\eta(c_i) \in \mathbb{R}_{>0}$ is a learned radius bias. In the appendix, we prove that Eq. (6) is structurally equivalent to Eq. (13). Thus, from the perspective of ontology representation learning, the weight matrix $\mathbf{W}_{\text{pred}}$ learned on the LLM latent space can be interpreted as an ontology embedding with relational biases.

We adopt four normalized axiom forms supported by ELEmbeddings, each corresponding to a specific type of logical relation commonly found in ontologies:

- **NF1.** Subclass axioms of the form $c_i \sqsubseteq c_j$, indicating that concept c_i is a subclass of c_j.
- **NF2.** Conjunctive subclass axioms $c_i \sqcap c_j \sqsubseteq c_k$, stating that the intersection of concepts c_i and c_j is a subclass of c_k.
- **NF3.** Existential inclusion axioms of the form $c_i \sqsubseteq \exists R.c_j$, meaning that instances of c_i are related via relation R to some instance of c_j.
- **NF4.** Existential restriction axioms $\exists R.c_i \sqsubseteq c_j$, expressing that any entity related to an instance of c_i via relation R must belong to concept c_j.

These normalized forms serve as the basis for encoding ontological constraints as geometric relations within the embedding space. The training loss is defined as

$$\mathcal{L}_{\text{axiom}} = \mathcal{L}_{\text{NF1}} + \mathcal{L}_{\text{NF2}} + \mathcal{L}_{\text{NF3}} + \mathcal{L}_{\text{NF4}}. \tag{14}$$

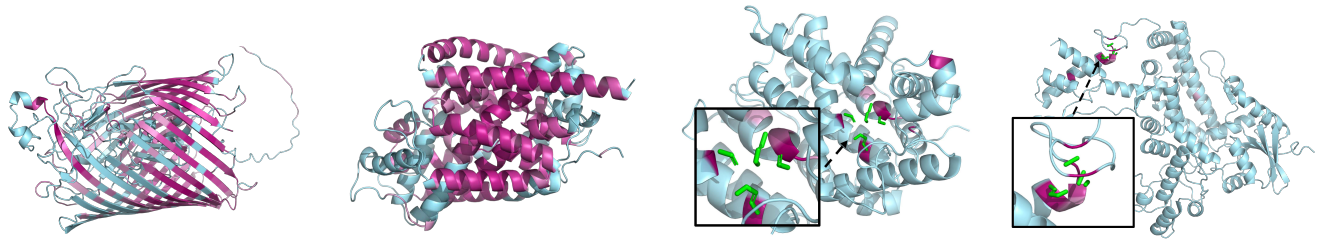
where L_1 to L_4 represent the training losses of the four axioms under ProtSAE. Appendix presents the detailed formulation of NF1 to NF4 and the derivation of the corresponding training loss.

Training Strategy

The overall training objective combines the reconstruction loss, the supervised prediction loss, and a semantic regularization term guided by domain knowledge:

$$\mathcal{L} = \underbrace{\|\hat{\mathbf{x}} - \mathbf{x}\|_2^2}_{\mathcal{L}_{\text{rec}}} + \lambda_{\text{annot}}\,\mathcal{L}_{\text{annot}} + \lambda_{\text{axiom}}\,\mathcal{L}_{\text{axiom}}, \tag{15}$$

where $\hat{\mathbf{x}}$ is the reconstructed input defined in Eq. (2). Here, $\mathcal{L}_{\text{rec}}$ encourages faithful reconstruction of the input, $\mathcal{L}_{\text{annot}}$ is the cross-entropy loss in Eq. (7), $\mathcal{L}_{\text{axiom}}$ regularizes semantic alignment based on domain knowledge in Eq. (14). We use λ_{annot} and λ_{axiom} to weight these losses. Appendix describes the detailed computation process.



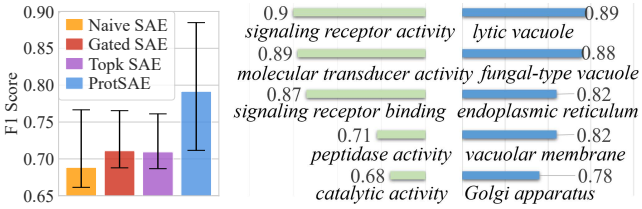
(a) The feature related to *iron ion binding* activated on the TonB-dependent receptor.

(b) The feature related to *sodium ion transport* activated on transmembrane segments.

(c) Interpreting Mn^{2+} ion binding sites with an activated feature.

(d) Interpreting Zn^{2+} ion binding sites with an activated feature.

Figure 3: Interpretability visualization shows that ProtSAE reveals semantic alignment between learned features and protein structures, including functional regions and ion binding sites. We use red intensity to indicate feature activation strength, and green sticks to mark ground truth binding sites.



(a) F1-score (mean, max, min) of top-10 activations. All results are reported as averages over 15 concepts.

(b) Top-5 activated concepts related to *receptor ligand activity* (left) and *storage vacuole* (right)

Figure 4: Comparison of relevance-based interpretation

Experiments and Results

Experiment Setup

Dataset and baselines. We construct the annotation data from protein function prediction datasets (Kulmanov et al. 2024). The protein function prediction datasets contains 77,647 proteins extracted from UniProtKB/Swiss-Prot. The annotated concepts can be categorized into three sub-ontologies: molecular function (MFO), biological process (BPO), and cellular component (CCO). Furthermore, we use the ion binding sites dataset (Yuan et al. 2022) as another annotation dataset, which covers four biologically relevant ion types: Zn^{2+}, Ca^{2+}, Mg^{2+}, and Mn^{2+}. We compare ProtSAE with widely adopted SAE baselines, including Naive SAE, Gated SAE (Rajamanoharan et al. 2024), and TopK SAE (Gao et al. 2024), in terms of both interpretability experiments and performance analyses. In the probing-based interpretation experiments, we further compare linear probing on the PLM hidden representations and the dictionary learning method SpLiCE (Bhalla et al. 2024). Detailed datasets and baseline settings are included in the appendix.

Evaluation metrics. For relevance-based interpretation, we use F1-score to evaluate the relevance between neurons and biological concepts. To evaluate the probing-based interpretation, we employ standard metrics from the protein function prediction benchmark (Kulmanov et al. 2024) including AUPR, AUC, maximum protein-centric F-measure ($F_{\max}$), and minimum semantic distance ($S_{\min}$). We use Loss Recovered (Rajamanoharan et al. 2024) to assess the reconstruction fidelity on varying sparsity. For intervention, we evaluate structural similarity using Template Modeling score (TM-score) (Zhang and Skolnick 2004) and Root Mean Square Distance (RMSD) (Betancourt and Skolnick 2001). Details are described in the appendix.

Implementation. We train all SAE on the internal activations of ESM2-15B, with $5e^{-4}$ learning rate, 12,800 batch size, and 25,000 steps. For ProtSAE and TopK SAE, the number of active neurons K is varied in $\{50, 100, 500, 1000\}$. Gated SAE is tuned with L1 coefficients in $\{1.5e^{-4}, 2e^{-4}, 3e^{-4}, 4e^{-4}, 5e^{-4}\}$, and Naive SAE in $\{8e^{-5}, 6e^{-5}, 2e^{-4}, 3e^{-4}, 4e^{-4}\}$. All SAE activation width is set to 40,000 for BPO, 30,000 for MFO and CCO, and 10,000 for the ion binding-site dataset. λ_{annot} and λ_{axiom} are fixed at 1. Experiments are run on four NVIDIA A800 GPUs.

Interpretability Experiments

Interpretability visualization. Figure 3 visualizes the features learned by ProtSAE that are aligned with biological concepts and demonstrates their utility in interpreting and exploring the semantics of specific protein structural elements. Warmer colors (e.g., red) indicate stronger activation of the feature at that amino acid. The positions of the true binding sites are marked by green sticks. In Figure 3a, we examine activations associated with the concept *iron ion binding* on protein P06971. We observe strong activation in regions corresponding to the *TonB-dependent receptor* structure, which is known to be tightly associated with the recognition and transport of Fe^{3+} ions. In Figure 3b, the feature related to *sodium ion transport* is highly activated on the transmembrane segments of protein O67854, suggesting that certain α-helical transmembrane regions may play a crucial role in sodium ion transport. Furthermore, Figures 3c and 3d show that features related to *metal ion binding sites* can highlight binding sites in proteins.

Relevance-based interpretation evaluation. In this experiment, we evaluate whether ProtSAE can effectively identify features related to specific concepts. Following the previous

Method	$F_{\max}\uparrow$	$S_{\min}\downarrow$	AUPR↑	AUC↑
SpLiCE	.417	23.4	.360	.329
Naive SAE	.421	23.3	.340	.511
Gated SAE	.441	22.7	.368	.533
TopK SAE	.444	22.7	.379	.565
Linear Probe	.537	**20.9**	**.522**	.751
ProtSAE	**.579**	**20.9**	.487	**.797**

Table 1: Average performance across three datasets on probing-based interpretation

work (Simon and Zou 2024), we extract relevant sequences from UniProtKB,[1] and construct a validation set of 5,000 sequences for each of 15 GO terms along with an equal number of unrelated sequences used as negative examples. We compute the activation of each feature with respect to a given concept and report the top-10 features ranked by F1-score. A higher F1-score indicates a stronger correlation between the feature and the concept.

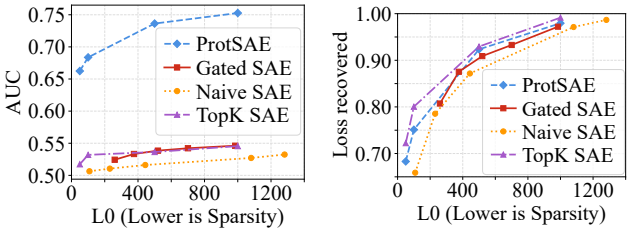
ProtSAE identifies features that are more semantically aligned with the target concepts. Figure 4a shows the average results of 15 concepts. Compared to the baselines, ProtSAE demonstrates significantly stronger relevance in both the mean and maximum activation levels. This suggests that the incorporation of semantic guidance during training enables ProtSAE to effectively disentangle semantic signals, and learns more accurate concept-related features. We further present additional highly activated features that show strong relevance to the validation set. As shown in Figure 4b, these features exhibit close functional or structural relationships with the target concept. For example, in the case of *storage vacuole*, the model activates structurally similar vacuole types such as *lytic vacuole* and *fungal-type vacuole*, as well as membrane-associated components like *vacuolar membrane* and *Golgi apparatus*.

Probing-based interpretation evaluation. To better evaluate the capability of probing-based interpretation, we conduct protein function prediction across datasets from three ontologies. The averaged results are summarized in Table 1. For a fair comparison, all SAE-based methods are evaluated under the same sparsity level. Notably, ProtSAE consistently outperforms all SAE baselines and the dictionary learning method SpLiCE across all evaluation metrics, and achieves comparable performance to linear probing on the hidden representations of PLMs. It suggests that ProtSAE, with semantic guidance, encourages the model to attend more effectively to the biological concepts during training. As a result, it mitigates the semantic loss that may occur during the SAE training, thereby achieving performance comparable to direct linear probing on PLMs.

Performance Analyses

Performance across different sparsity. Figure 5 illustrates the effect of sparsity on model performance. We assess the reconstruction fidelity using the metrics of *Loss Recovered*, which measures the proportion of the original PLM loss that

[1]https://www.uniprot.org/



(a) AUC under different sparsity (b) Loss Recovered under different sparsity

Figure 5: Performance comparison under different sparsity on the BPO dataset

can be recovered using SAE. The left subfigure shows the AUC performance under varying levels of sparsity. ProtSAE consistently outperforms all baselines, indicating its ability to preserve semantics relevant to predefined concepts even under high sparsity, thereby achieving superior predictive performance. The right subfigure shows the trend of reconstruction fidelity as sparsity increases. Compared to other SAE variants, ProtSAE maintains comparable reconstruction quality, demonstrating its effectiveness in decomposing semantic concepts while faithfully preserving the original latent representations from PLMs. Detailed results on all datasets are provided in the appendix.

Ablation study. We conduct an ablation study of ProtSAE by removing key components and retraining the model under various sparsity levels. Figure 6 depicts the ablation results. We discuss the effect of each component below:

- Without `detach`. In this variant, we no longer detach $\mathbf{W}_{\text{pred}}$ when constructing $\mathbf{W}_{\text{def}}$, allowing the gradients from $\mathcal{L}_{\text{rec}}$ to update $\mathbf{W}_{\text{pred}}$ directly. So, the defined activations are now updated not only by the supervised data, but also by the reconstruction objective. While this slightly improves reconstruction fidelity, it leads to a dramatic decrease in AUC. This suggests that allowing $\mathcal{L}_{\text{rec}}$ to influence $\mathbf{W}_{\text{pred}}$ introduces entangled or ambiguous semantics into the defined activations, thereby degrading precision and interpretability.
- Removing $\mathcal{L}_{\text{axiom}}$. When the axiom learning component based on ELEmbeddings is removed, the training of $\mathbf{W}_{\text{pred}}$ relies solely on the supervised data. This leads to a clear degradation in both AUC and reconstruction fidelity, highlighting the importance of modeling complex concept relationships through axioms to capture the intricate semantic structure of protein functions.
- Without $\mathbf{z}_{\text{bias}}$. In the right subfigure of Figure 6, we report the proportion of defined activations predicted as active that are indeed used during decoding. With $\mathbf{z}_{\text{bias}}$, nearly all predicted activations participate in reconstruction, indicating that $\mathbf{z}_{\text{def}}$ holds strong potential for steering. Removing $\mathbf{z}_{\text{bias}}$ reduces this proportion, weakening the alignment between prediction and actual activation.
- Without $\mathbf{r}_{\text{pred}}$. By setting the scaling parameter $\mathbf{r}_{\text{pred}}$ in Eq. (8) to zero, $\mathbf{W}_{\text{def}}$ and $\mathbf{W}_{\text{pred}}$ become identical. This

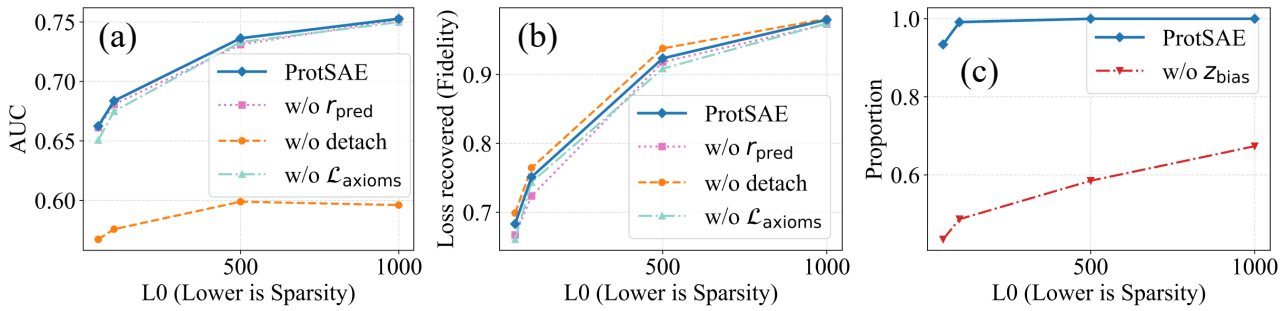


Figure 6: Ablation results on (a) AUC, (b) Loss Recovered, and (c) reconstruction proportion of predicted activations w.r.t. L_0

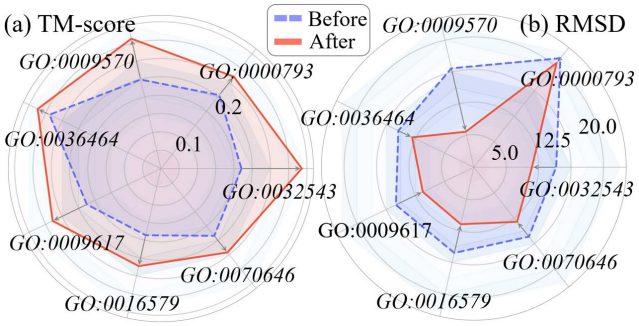


Figure 7: Effect of TM-score (left) and RMSD (right) before and after intervention

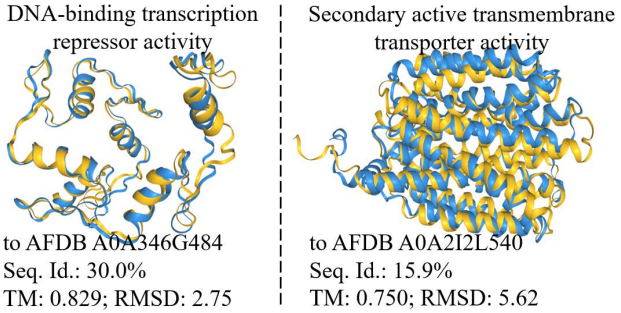


Figure 8: Intervention case study

modification causes drops in both AUC and reconstruction fidelity. The removal of $\mathbf{r}_{\text{pred}}$ hinders the model's ability to learn feature magnitudes.

Steering Experiment

Concept intervention. We conduct a steering experiment across various biological concepts to evaluate whether ProtSAE can effectively steer PLM's generation based on the learned concept-specific features. For each of seven selected concept-related sequences, we mask 50% of the tokens and compare the reconstructions generated before and after intervention. Following previous works (Zongying et al. 2024; Lv et al. 2024; Liu et al. 2025), we use TM-score and RMSD to measure structural similarity between the generated sequences and the natural proteins with the target concept, in order to assess whether the intervention can guide the model's generation aligned with the desired concept. We use pLDDT to evaluate the structure stability.

As shown in Figure 7, TM-scores significantly increase while RMSD decreases after intervention, indicating improved structural alignment with the target concepts. Furthermore, the appendix includes detailed results and highlights significant improvements in the pLDDT scores of the generated proteins. These results suggest that ProtSAE successfully stores concept-aligned representations in its learned dictionary, and activating these features during generation enables the PLM to produce structurally stable proteins that better reflect the semantics of the desired concept.

Case study. Figure 8 visualizes proteins generated by ProtSAE after intervention (in blue). We identify their most similar natural counterparts (in yellow) with Foldseek (van Kempen et al. 2022). After intervention, ProtSAE can generate proteins with high structural similarity to natural counterparts with relevant concepts, remaining low sequence identity. This demonstrates that ProtSAE effectively captures concept-specific structural features and can successfully steer PLM's generation. For example, we intervene on the concept of "*DNA-binding transcription repressor activity*" and generate a protein structurally similar to the natural protein "A0A346G484" (TM-score: 0.829, RMSD:2.75), while maintaining sequence novelty (Seq. ID: 30.0%). "A0A346G484" contains a putative zinc-finger domain and is annotated with the desired concept. We also generate proteins with high structural similarity to natural proteins exhibiting transmembrane transporter activity.

Conclusion

We propose ProtSAE, a semantically-guided SAE to tackle semantic entanglement in SAE training and improve interpretability of PLMs. We introduce domain knowledge into ProtSAE to constraint the relationship among concepts, and apply forced activations and feature rescaling to ensure that the learned features effectively contribute to the reconstruction while maintaining high reconstruction fidelity. Interpretability experiments show that ProtSAE consistently captures features more aligned with protein structures and functions. Performance analyses and steering experiments show the superiority of ProtSAE against existing SAE baselines.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 62272219).

References

Adams, E.; Bai, L.; Lee, M.; Yu, Y.; and AlQuraishi, M. 2025. From Mechanistic Interpretability to Mechanistic Biology: Training, Evaluating, and Interpreting Sparse Autoencoders on Protein Language Models. *bioRxiv*, 2025–02.

Betancourt, M. R.; and Skolnick, J. 2001. Universal similarity measure for comparing protein structures. *Biopolymers: Original Research on Biomolecules*, 59(5): 305–309.

Bhalla, U.; Oesterling, A.; Srinivas, S.; Calmon, F.; and Lakkaraju, H. 2024. Interpreting clip with sparse linear concept embeddings (splice). *NeurIPS*, 37: 84298–84328.

Dong, T.; Kan, C.; Devkota, K.; and Singh, R. 2024. AlloAllo: Data-efficient prediction of allosteric sites. *bioRxiv*, 2024–09.

Dunefsky, J.; Chlenski, P.; and Nanda, N. 2024. Transcoders find interpretable LLM feature circuits. In *NeurIPS*.

Elhage, N.; Hume, T.; Olsson, C.; Schiefer, N.; Henighan, T.; Kravec, S.; Hatfield-Dodds, Z.; Lasenby, R.; Drain, D.; Chen, C.; et al. 2022. Toy models of superposition. *arXiv*.

Ferruz, N.; and Höcker, B. 2022. Controllable protein design with language models. *Nature Machine Intelligence*, 4(6): 521–532.

Gallifant, J.; Chen, S.; Sasse, K.; Aerts, H.; Hartvigsen, T.; and Bitterman, D. S. 2025. Sparse autoencoder features for classifications and transferability. *arXiv*.

Gao, L.; la Tour, T. D.; Tillman, H.; Goh, G.; Troll, R.; Radford, A.; Sutskever, I.; Leike, J.; and Wu, J. 2024. Scaling and evaluating sparse autoencoders. *arXiv*.

Garcia, E. N. V.; and Ansuini, A. 2025. Interpreting and Steering Protein Language Models through Sparse Autoencoders. *arXiv*.

Gujral, O.; Bafna, M.; Alm, E.; and Berger, B. 2025. Sparse autoencoders uncover biologically interpretable features in protein language model representations. *Proceedings of the National Academy of Sciences*, 122(34): e2506316122.

Gurnee, W.; Nanda, N.; Pauly, M.; Harvey, K.; Troitskii, D.; and Bertsimas, D. 2023. Finding Neurons in a Haystack: Case Studies with Sparse Probing. *Transactions on Machine Learning Research*.

Ismail, A. A.; Oikarinen, T.; Wang, A.; Adebayo, J.; Stanton, S. D.; Bravo, H. C.; Cho, K.; and Frey, N. C. 2025. Concept Bottleneck Language Models For Protein Design. In *ICLR*.

Joshi, S.; Dittadi, A.; Lachapelle, S.; and Sridhar, D. 2025. Identifiable Steering via Sparse Autoencoding of Multi-Concept Shifts. *arXiv*.

Kannan, G. R.; Hie, B. L.; and Kim, P. S. 2024. Single-Sequence, Structure Free Allosteric Residue Prediction with Protein Language Models. *bioRxiv*, 2024–10.

Kulmanov, M.; Guzmán-Vega, F. J.; Duek Roggli, P.; Lane, L.; Arold, S. T.; and Hoehndorf, R. 2024. Protein function prediction as approximate semantic entailment. *Nature Machine Intelligence*, 6(2): 220–228.

Kulmanov, M.; and Hoehndorf, R. 2022. DeepGOZero: Improving protein function prediction from sequence and zero-shot learning based on ontology axioms. *Bioinformatics*, 38(Supplement_1): i238–i245.

Kulmanov, M.; Liu-Wei, W.; Yan, Y.; and Hoehndorf, R. 2019. EL Embeddings: Geometric construction of models for the description logic EL++. In *IJCAI*, 6103–6109.

Lin, B.; Luo, X.; Liu, Y.; and Jin, X. 2024. A comprehensive review and comparison of existing computational methods for protein function prediction. *Briefings in Bioinformatics*, 25(4): 289.

Lin, Z.; Akin, H.; Rao, R.; Hie, B.; Zhu, Z.; Lu, W.; Smetanin, N.; Verkuil, R.; Kabeli, O.; Shmueli, Y.; et al. 2023a. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637): 1123–1130.

Lin, Z.; Akin, H.; Rao, R.; Hie, B.; Zhu, Z.; Lu, W.; Smetanin, N.; Verkuil, R.; Kabeli, O.; Shmueli, Y.; et al. 2023b. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637): 1123–1130.

Liu, X.; Liu, Y.; Chen, S.; and Hu, W. 2025. Controllable Protein Sequence Generation with LLM Preference Optimization. In *AAAI*.

Lv, L.; Lin, Z.; Li, H.; Liu, Y.; Cui, J.; Chen, C. Y.-C.; Yuan, L.; and Tian, Y. 2024. ProLLaMA: A protein large language model for multi-task protein language processing. *arXiv preprint arXiv:2402.16445*.

Makelov, A. 2024. Sparse autoencoders match supervised features for model steering on the ioi task. In *ICML 2024 Workshop on Mechanistic Interpretability*.

Makelov, A.; Lange, G.; Geiger, A.; and Nanda, N. 2024. Is This the Subspace You Are Looking for? An Interpretability Illusion for Subspace Activation Patching. In *ICLR*.

Miller, J.; Chughtai, B.; and Saunders, W. 2024. Transformer Circuit Faithfulness Metrics are not Robust. In *COLM*.

Nijkamp, E.; Ruffolo, J. A.; Weinstein, E. N.; Naik, N.; and Madani, A. 2023. ProGen2: Exploring the boundaries of protein language models. *Cell Systems*, 14(11): 968–978.

Nori, D.; Singireddy, S.; and Have, M. T. 2023a. Identification of Knowledge Neurons in Protein Language Models. *arXiv preprint arXiv:2312.10770*.

Nori, D.; Singireddy, S.; and Have, M. T. 2023b. Identification of Knowledge Neurons in Protein Language Models. *arXiv preprint arXiv:2312.10770*.

Olah, C.; Cammarata, N.; Schubert, L.; Goh, G.; Petrov, M.; and Carter, S. 2020. Zoom in: An introduction to circuits. *Distill*, 5(3): e00024–001.

Pach, M.; Karthik, S.; Bouniot, Q.; Belongie, S.; and Akata, Z. 2025. Sparse Autoencoders Learn Monosemantic Features in Vision-Language Models. *arXiv preprint arXiv:2504.02821*.

Paulo, G.; Mallen, A.; Juang, C.; and Belrose, N. 2024. Automatically interpreting millions of features in large language models. *arXiv preprint arXiv:2410.13928*.

Rajamanoharan, S.; Conmy, A.; Smith, L.; Lieberum, T.; Varma, V.; Kramar, J.; Shah, R.; and Nanda, N. 2024. Improving sparse decomposition of language model activations with gated sparse autoencoders. In *NeurIPS*.

Rao, R.; Meier, J.; Sercu, T.; Ovchinnikov, S.; and Rives, A. 2021. Transformer protein language models are unsupervised structure learners. In *ICLR*.

Shi, C.; Beltran Velez, N.; Nazaret, A.; Zheng, C.; Garriga-Alonso, A.; Jesson, A.; Makar, M.; and Blei, D. 2024. Hypothesis testing the circuit hypothesis in LLMs. In *NeurIPS*.

Simon, E.; and Zou, J. 2024. InterPLM: Discovering Interpretable Features in Protein Language Models via Sparse Autoencoders. *bioRxiv*, 2024–11.

Tamkin, A.; Taufeeque, M.; and Goodman, N. 2024. Codebook Features: Sparse and Discrete Interpretability for Neural Networks. In *ICML*, 47535–47563.

Todd, E.; Li, M.; Sharma, A. S.; Mueller, A.; Wallace, B. C.; and Bau, D. 2024. Function Vectors in Large Language Models. In *ICLR*.

van Kempen, M.; Kim, S. S.; Tumescheit, C.; Mirdita, M.; Gilchrist, C. L.; Söding, J.; and Steinegger, M. 2022. Foldseek: Fast and accurate protein structure search. *bioRxiv*, 2022–02.

Vig, J.; Madani, A.; Varshney, L. R.; Xiong, C.; Rajani, N.; et al. 2021. BERTology Meets Biology: Interpreting Attention in Protein Language Models. In *ICLR*.

Wang, K. R.; Variengien, A.; Conmy, A.; Shlegeris, B.; and Steinhardt, J. 2023. Interpretability in the Wild: A Circuit for Indirect Object Identification in GPT-2 Small. In *ICLR*.

Wright, B.; and Sharkey, L. 2024. Addressing feature suppression in saes. In *AI Alignment Forum*, 16.

Wu, X.; Yu, W.; Zhai, X.; and Liu, N. 2025a. Self-regularization with latent space explanations for controllable LLM-based classification. *arXiv*.

Wu, X.; Yuan, J.; Yao, W.; Zhai, X.; and Liu, N. 2025b. Interpreting and steering LLMs with mutual information-based explanations on sparse autoencoders. *arXiv*.

Yuan, Q.; Chen, S.; Wang, Y.; Zhao, H.; and Yang, Y. 2022. Alignment-free metal ion-binding site prediction from protein sequence through pretrained language model and multi-task learning. *Briefings in Bioinformatics*, 23(6): bbac444.

Yun, Z.; Chen, Y.; Olshausen, B. A.; and LeCun, Y. 2021. Transformer visualization via dictionary learning: Contextualized embedding as a linear superposition of transformer factors. *NAACL-HLT*, 1.

Zhang, Y.; and Skolnick, J. 2004. Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics*, 57(4): 702–710.

Zhang, Z.; Wayment-Steele, H. K.; Brixi, G.; Wang, H.; Kern, D.; and Ovchinnikov, S. 2024. Protein language models learn evolutionary statistics of interacting sequence motifs. *Proceedings of the National Academy of Sciences*, 121(45): e2406285121.

Zongying, L.; Hao, L.; Liuzhenghao, L.; Bin, L.; Junwu, Z.; Yu-Chian, C. C.; Li, Y.; and Yonghong, T. 2024. TaxDiff: Taxonomic-Guided Diffusion Model for Protein Sequence Generation. *arXiv preprint arXiv:2402.17156*.