Variational Sparse Autoencoders for Disentangling Correlated Features in Language Models

Anonymous Authors
Paper under double-blind review

Abstract

Sparse autoencoders (SAEs) are one of the central tools in mechanistic interpretability for extracting monosemantic features from model activations via unsupervised dictionary learning. Yet in the presence of correlated or structured features, SAEs often exhibit failures such as feature splitting, absorption, and persistent polysemanticity. We introduce Variational SAEs (V-SAEs), which incorporate structured variational priors into the SAE framework to better reflect feature-level dependencies. This formulation yields a tractable, closed-form objective for correlation-aware disentanglement, enabling principled prior design from either synthetic latent geometry or empirical correlation in real model activations. We evaluate V-SAEs on both controlled toy models and LLM datasets, comparing isotropic, block-structured, and full-covariance priors. V-SAEs consistently outperform baselines in feature purity, sparsity, and alignment with ground truth directions, especially under high correlation.

1 Introduction

2 Related Work

Sparse autoencoders (SAEs) have emerged as a central tool in mechanistic interpretability, particularly for extracting modular and interpretable latent structure from language model activations [4, 5]. Recent research has improved SAE scalability [7] and atomicity of features [2], while various architectural innovations—such as Ladder networks [10, 11] and hierarchical VAEs [12]—have been explored in adjacent domains.

Our approach builds upon the variational autoencoder (VAE) framework [9], proposing structured priors to improve the disentanglement of correlated features. This continues a line of work on KL-regularized SAEs for disentanglement [1] and sparse dictionary learning for functional interpretability [3].

Complementary work by Zhou et al. [14] investigates how dictionary size affects feature granularity and stability, raising questions our experiments also address. Additionally, SAE training insights from energy-based models [13] and perceptual reconstruction losses [8, 6] inspire auxiliary losses used in our multi-resolutional extensions.

Our theoretical framing connects with ongoing discussion about the limits of the linear representation hypothesis (LRH), and our experiments attempt to clarify whether SAEs operate as faithful approximators or merely flexible learners of nonlinear structure.

Preprint. Under review.

3 Method: Variational Sparse Autoencoders

3.1 Mathematical Framework

To embed the correlation of features as a prior knowledge, we extend SAE with variational inference and refine its loss function to a variational version. Specifically, we first frame the encoder-decoder structure as a latent variable model (LVM). Then we interpret the feature correlations as distribution assumptions on the latent space. This allows us to derive the V-SAE objective from first principles using the evidence lower bound (ELBO) on the marginal likelihood.

1. Latent Variable Model (LVM) and Variational Inference

Let $x \in \mathbb{R}^d$ be the observed model activation (e.g., from a transformer MLP layer), and let $z \in \mathbb{R}^k$ be an **overcomplete** latent feature vector, where typically $k \geq d$, the input dimension. We posit a generative model:

$$p_\theta(x, z) = p_\theta(x|z)p(z) \tag{1}$$

where $p(z)$ is the prior over latent features, and $p_\theta(x|z)$ is the decoder likelihood. In the scope of Variational Autoencoders (VAEs), the prior can be a multivariate Gaussian while the posterior can be the Gaussian reconstruction.

The marginal likelihood is:

$$p_\theta(x) = \int p_\theta(x|z)p(z)dz \tag{2}$$

The goal is thus to maximize the marginal log-likelihood $\log p(x)$, which is typically intractable due to the integration over latents.

According to variational inference, we introduce a variational distribution to approximate the intractable posterior distribution, and learn the distribution parameters via the evidence lower bound (ELBO).

We introduce a variational approximation $q_\phi(z|x)$ to the posterior. The evidence lower bound can be written as follows,

$$\log p_\theta(x) \geq \mathbb{E}_{q_\phi(z|x)}\big[\log p_\theta(x|z)\big] - KL\big(q_\phi(z|x)\big|\big|p(z)\big) =: \mathcal{L}_{ELBO}(x) \tag{3}$$

This is the canonical variational objective. In our setup, we parameterize:

- $p(z)$: using different structured priors to resemble different correlation assumptions of features (isotropic, group-structured, full covariance)
- $q_\phi(z|x)$: learned encoder distribution with parameter ϕ
- $p_\theta(x|z)$: learned decoder distribution with parameter θ (typically assumed as Gaussian with fixed variance)

Therefore, the first term corresponds to a reconstruction loss representing the likelihood under the decoder. The second term corresponds to a regularization loss, representing KL divergence between variational posterior and prior, where we can embed our prior knowledge over the latent space.

2. Distribution Assumptions

We adopt Gaussian distribution for all components. Specifically, we assume the following:

- Decoder likelihood: $p_\phi(x|z) = \mathcal{N}\big(x; W^{dec}z + b, \sigma_x^2 \boldsymbol{I}\big)$
- Prior over latents: $p(z) = \mathcal{N}\big(z; \mu_p, \Sigma_p\big)$
- Posterior: $q_\theta(z|x) = \mathcal{N}(z; \mu(x), \Sigma(x))$

This leads to the loss function of V-SAE:

$$\mathcal{L}(x) = \|x - \hat{x}\|_2^2 + \alpha \cdot KL\big(q(z|x)\big|\big|p(x)\big) \tag{4}$$

where α is a hyper-parameter for weighting.

3. Posterior Parameterizations: General to Special Cases

To embed the feature correlation, we consider three variants of posterior-prior pairings in increasing expressiveness:

Case A: Isotropic Posterior and Prior

$$q(z|x) = \mathcal{N}(z; \mu(x), \sigma^2 \boldsymbol{I}),\ p(z) = \mathcal{N}(z; \mathbf{0}, \boldsymbol{I}) \tag{5}$$

KL divergence simplifies to elementwise closed-form:

$$KL_{iso}(q\|p) = \sum_{i=1}^{k} \big[-\frac{1}{2}(1 + \log \sigma_i^2 - \mu_i^2 - \sigma_i^2)\big] \tag{6}$$

This is similar to the VAE baseline and can match the SAE loss with fixed variances.

Case B: Diagonal Posterior, Correlated Prior Assume:

$$q(z|x) = \mathcal{N}(z; \mu(x), \mathrm{diag}(\sigma_1^2, \ldots, \sigma_k^2)), \quad p(z) = \mathcal{N}(0, \Sigma_p) \tag{7}$$

Let the prior covariance be:

$$\Sigma_p = \rho J + (1 - \rho) I, \quad \text{with } J_{ij} = 1 \tag{8}$$

Then the inverse and determinant are:

$$\Sigma_p^{-1} = \frac{1}{1-\rho}\left(I - \frac{\rho}{1 + (k-1)\rho} J\right), \quad |\Sigma_p| = (1-\rho)^{k-1}(1 + (k-1)\rho) \tag{9}$$

The KL divergence has closed form:

$$\mathrm{KL}_{corr}(q\|p) = \frac{1}{2}\left[\log \frac{|\Sigma_p|}{\prod_{i=1}^{k} \sigma_i^2} - k + \mathrm{tr}(\Sigma_p^{-1}\Sigma_q) + \mu^\top \Sigma_p^{-1} \mu\right] \tag{10}$$

This allows for modeling correlated features via a constant off-diagonal structure. Empirically, we find this improves disentanglement when true latents are group-correlated (e.g., in toy models with setwise superposition). We refer to this as the *correlated prior V-SAE*.

Case C: Full Covariance Posterior and Prior Let:

$$q(z|x) = \mathcal{N}(\mu(x), \Sigma_q), \quad p(z) = \mathcal{N}(\mu_p, \Sigma_p) \tag{11}$$

Then the KL divergence is the full multivariate form:

$$\mathrm{KL}(q\|p) = \frac{1}{2}\left[\mathrm{tr}(\Sigma_p^{-1}\Sigma_q) + (\mu - \mu_p)^\top \Sigma_p^{-1}(\mu - \mu_p) - k + \log \frac{|\Sigma_p|}{|\Sigma_q|}\right] \tag{12}$$

This is the most general setting, capturing arbitrary correlation structure between features. While more expressive, it requires computing and differentiating full covariance matrices, which increases computational cost.

$$q_\phi(z|x) = \mathcal{N}(z; \mu_\phi(x), diag(\sigma_i^2)), \tag{13}$$

$$p(z) = \mathcal{N}(z; \mathbf{0}, \Sigma_p), \text{with } \Sigma_p = \rho \boldsymbol{J} + (1 - \rho)\boldsymbol{I} \tag{14}$$

Closed-form KL using the Sherman-Morrison formula:

3.2 Theoretical Insights (Zoom-in from General to Special)

Further we zoom in from general posterior parameterizations to specific case, especially **Case C (full-covariance Gaussian)**.

TODO: Henry's derivation here, and more details leave in appendix

Revisiting **Case C: Full-Covariance Posterior** ($\boldsymbol{z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$), we now consider the general case where both the encoder posterior $q(\boldsymbol{z}|\boldsymbol{x})$ and the prior $p(\boldsymbol{z})$ are multivariate Gaussians with full covariance. Specifically:

Encoder: $q(\boldsymbol{z}|\boldsymbol{x}) = \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{x}), \boldsymbol{L}\boldsymbol{L}^\top)$

Prior: $p(\boldsymbol{z}) = \mathcal{N}(\mathbf{0}, \boldsymbol{I})$

The KL divergence for this case admits a closed-form:

$$KL\big(q(z|X)\big|\big|p(z)\big) = \frac{1}{2}\big(tr(\boldsymbol{\Sigma} + \boldsymbol{\mu}^T\boldsymbol{\mu} - \boldsymbol{k} - \log\det(\Sigma))\big) \tag{15}$$

where $\boldsymbol{\Sigma} = \boldsymbol{L}\boldsymbol{L}^\top$ and k is the latent dimensionality.

This parameterization permits learning structured posteriors that model entangled or correlated features, particularly useful in settings where linear SAEs suffer from feature splitting. However, it comes with greater sample complexity and sensitivity to the covariance structure encoded in the prior.

See Appendix A for full derivation and numerical stability details.

The further theoranalysis further improve theoretical faithfulness of V-SAE to variational inference, justifying usage of more complex priors over SAEs. It also allows testable alignment between assumed priors and latent structure — something that's hard to get from just tuning α in SAE loss.

3.3 Interpretability Rationale

A key motivation for variational sparse autoencoders (V-SAEs) lies in their ability to incorporate explicit structural assumptions via the prior distribution $p(\mathbf{z})$. These assumptions serve as inductive biases, shaping the geometry of the latent space and thereby influencing both reconstruction fidelity and the semantic disentanglement of features. We highlight the interpretability implications of several commonly used prior families.

- **Isotropic Gaussian Prior ($\mathcal{N}(0, I)$):** Enforces spherical symmetry and independence among latent dimensions. This is the default choice in many variational models due to its simplicity and closed-form KL divergence. However, this prior is oblivious to correlations or clustering structure among features and may therefore induce polysemantic or entangled activations when the underlying generative factors are dependent.
- **Structured Low-Rank Priors (e.g., Shared Correlation $\Sigma_p = \rho J + (1-\rho)I$):** Captures partial dependency by imposing soft global correlation among latent units. This prior is analytically tractable while allowing latent codes to co-activate along correlated directions. It is especially suitable for modeling subspaces where features arise in overlapping or combinatorial groupings, though it may underperform when true correlations are sparse or hierarchical rather than globally shared.
- **Full-Covariance Priors (Σ_p arbitrary):** Provide maximal flexibility to represent feature correlations of arbitrary form. These priors enable the posterior to adapt to the manifold geometry of latent factors, which is essential in tasks involving highly entangled or clustered features. However, this expressiveness introduces practical challenges: estimating Σ_p may require auxiliary learning stages or domain knowledge, and the high-dimensional KL divergence may destabilize training when the true latent structure is low-rank or sparse.

From a variational perspective, each prior class defines a different family $\mathcal{Q}$ of admissible approximate posteriors $q(\mathbf{z}|\mathbf{x})$. More expressive families (e.g., non-diagonal posteriors) yield tighter evidence lower bounds (ELBO) in theory, but only improve interpretability and generalization when the prior structure aligns with the true posterior $p(\mathbf{z}|\mathbf{x})$. Hence, improvements observed in downstream metrics—such as feature sparsity, alignment with ground-truth factors, or clustering entropy—stem not merely from increased flexibility, but from principled structural matching.

In this context, V-SAEs offer a theoretical upgrade over conventional SAEs: the KL term enforces probabilistic consistency between encoder outputs and prior assumptions, beyond the crude L1 regularization in traditional SAE loss. Furthermore, this framework supports testable hypotheses about the latent space—for instance, evaluating how well a prior assumption matches empirical feature covariance—enabling diagnostics and architectural decisions grounded in statistical geometry rather than heuristics alone.

4 Experiments

Experiment TODOs:

1. **Toy Model Results**: Finalize training runs on setwise correlation and Cholesky-Gaussian cases using full-covariance V-SAE. Generate latent heatmaps, disentanglement comparisons, and overlap plots.
2. **KL Divergence Tracking**: Visualize KL term dynamics across priors (isotropic, mixture, multivariate) during training. Include theoretical alignment to derived forms.
3. **TinyStories Activation Experiments**: Train baseline SAE and full-covariance V-SAE on 1L-GELU TinyStories. Compare using SAEBench metrics (sparsity, feature overlap, clustering entropy, reconstruction).
4. **SAEBench Benchmarking**: Use standard metrics from `https://www.neuronpedia.org/sae-bench/info#unsupervised-metrics-core` to compare models. Format as table.
5. **Visualizations**: Create placeholder plots: latent correlation heatmaps, reconstruction loss trajectories, example reconstructions, TopK decoder overlap.
6. **Stretch**: Repeat key experiments with larger dataset (e.g., Pythia-70M or gelu-6L layer) if time allows.

We evaluate Variational Sparse Autoencoders (V-SAEs) across two experimental domains: (1) controlled toy models with synthetic latent correlation structure, and (2) real-world LLM representations using TinyStories activations. Our goal is to assess whether correlation-aware priors in V-SAEs improve feature disentanglement, reduce overlap, and preserve reconstruction accuracy under sparse constraints. **Code Repository:** All code for training and evaluation is available at `github.com/JadeLilyx/v-sae_mech-interp-spar2025`.

4.1 Setup and Metrics

We compare standard SAEs (with TopK sparsity) against three V-SAE variants:

- **V-SAE with isotropic Gaussian prior** (baseline VAE-like),
- **V-SAE with Gaussian mixture prior** for setwise disentanglement,
- **V-SAE with full covariance prior**, enabling correlation-sensitive inference.

All models share a sparse decoder and Gaussian latent posterior with reparameterization. The loss includes a squared reconstruction error and a KL divergence regularization term, derived analytically. For all experiments, we use both custom metrics and SAEBench [7]:

- **Reconstruction error** (MSE),
- **KL divergence** (closed-form),
- **Decoder overlap matrix** (cosine similarity between feature vectors),
- **Feature activation sparsity** (TopK thresholding),
- **Latent entropy** (Shannon entropy across tokens),
- **SAEBench unsupervised metrics:** L1 coherence, identity similarity, compositionality.

4.2 Toy Model Experiments

1) Variational + TopK; 2) See if TopK can be modeled by some distribution; 3) Since all Gaussian assumptions only matches the first 2 modes, if we can theoretically assume a Laplacian graph for covariance matrix and embed that as prior.

Toy Model Experiments ($TMS_VSAE.ipynb$):

1. **Implementation:**
   - Toy Model 1: Groupwise correlated features.
   - Toy Model 2: Full-covariance correlated latent sampling.
   - Trained baseline SAE, JumpReLU SAE, and V-SAEs with isotropic + structured priors.
   - Visualized decoder weight heatmaps and cosine overlaps.

- Compared sparsity, reconstruction loss, KL, and latent dimension usage.

2. **Results:**
   - Structured V-SAEs achieve lower reconstruction loss + better disentanglement (qualitative & quantitative).
   - KL divergence tracks correlation strength effectively.
   - Heatmaps show reduced feature overlap compared to baseline.

To isolate the effect of correlation, we construct two synthetic benchmarks inspired by the Toy Models of Superposition framework. We reproduce and extend toy model setups proposed in [5] and [**?**]:

To illustrate the difficulty of disentangling correlated features, consider the simple setup extended from [5]: A 'bottleneck' encoder reconstructs a high dimensional signal with a non-linear 2-dimensional layer, i.e. the 'bottleneck'. A sparse signal is simulated by drawing each element of the signal from $U(0, p)$. The sparsity of the signal is defined as $S \equiv 1 - p$ and higher sparsity leads to better feature representation due to reduced interference in the signal. The 'bottleneck' encoder is defined as follows:

$$h = Wx\,, \tag{16}$$

$$f(h) = W^T h + b \tag{17}$$

where $x \in \mathbb{R}^{d_s}$ and $W \in \mathbb{R}^{d_f}$ with $d_s > d_f$. For the purpose of visualization, we will choose the bottleneck dimension $d_f = 2$. The encoder is then trained with the MSE loss. Structured correlations can be constructed by correlating pairs of dimensions in the signal, which can be anti-correlated or correlated.

Anti-correlated $p(x_i, x_{i+1}|x_i \neq 0) = 0$

Correlated $p(x_i, x_{i+1}|x_i \neq 0) = 1$

VSAE Setup

$$h_{\text{cent}} = h - b_{\text{dec}}\,, \tag{18}$$

$$\mu = \text{ReLU}(W^{\mu}_{\text{enc}} h + b^{\mu}_{enc})\,, \tag{19}$$

$$\sigma^2 = \text{ReLU}(W^{\sigma}_{\text{enc}} h + b^{\sigma}_{enc})\,, \tag{20}$$

$$z = \mu + \epsilon\sqrt{\sigma^2} A\,, \tag{21}$$

$$h_{\text{recon}} = W_{\text{dec}} z + b_{\text{dec}} \tag{22}$$

with A parameterizing the scale of the learned variances.

- **Toy Model 1: Setwise Correlation.** Feature groups sampled from overlapping Gaussians with $\rho \in [0.0, 0.8]$.
- **Toy Model 2: General Correlated Latents.** Latent vectors from multivariate Gaussian with Cholesky-decomposed covariance matrix.

4.3 Toy Models: Controlled Correlation Benchmarks

To isolate the effect of correlation, we construct two synthetic benchmarks inspired by the Toy Models of Superposition framework:

Toy Model 1: Setwise Correlation. Latents are grouped into clusters of correlated variables (within-group correlation coefficient ρ), then projected to a linear observation space.

Toy Model 2: Full-Covariance Gaussians. Latents are sampled from a multivariate Gaussian with a dense Cholesky-decomposed covariance matrix to introduce general structured correlations.

Compared Models:

- **SAE Baseline:** TopK sparse decoder + L1 penalty.
- **JumpReLU SAE:** Adaptive thresholding sparsity method.

- **V-SAE Isotropic:** Variational encoder with $\mathcal{N}(0, I)$ prior.
- **V-SAE Structured:** Full-covariance Gaussian prior, analytically derived.

Metrics: Reconstruction loss, decoder weight overlap, KL divergence, sparsity level (L0), and cosine similarity between recovered features and ground-truth latents.

Findings: V-SAEs with structured priors achieve:

- Lower decoder overlap (indicating better disentanglement),
- Higher entropy in active features (less collapsed features),
- Comparable or better reconstruction error at fixed sparsity.

`TODO:` Run new sweeps across:

- Varying dictionary width $n \in 256, 512, 1024$,
- KL weight $\alpha \in 0.01, 0.1, 1.0$,
- Prior type (isotropic, mixture, full covariance).

`FIGURE:` *Decoder Overlap Matrix (heatmaps) for different priors.*
`TABLE:` *Reconstruction error vs KL entropy tradeoff.*

4.4 Real-World Datasets Experiments

4.5 Empirical Correlation Estimation for Prior Design

In contrast to toy models with known latent dependencies, real-world data (e.g., TinyStories or C4-Code) lacks explicit ground-truth correlation structures among features. To design variational priors that meaningfully capture inter-feature dependencies, we introduce an empirical correlation estimation pipeline.

Methodology: We analyze activations produced by a pretrained SAE or encoder model over a representative token set. For each latent feature, we compute:

- **Pearson correlation coefficients** and **cosine similarities** between activation vectors.
- **Mutual information** between activation pairs to capture non-linear dependence.
- **SVD/PCA eigenspectrum** of the activation matrix to detect low-rank latent clusters or superposition.

These statistics are visualized as heatmaps and optionally clustered to reveal block structure. We use the estimated correlation or covariance matrices to construct structured priors in downstream V-SAE and Crosscoder models.

Motivation: This step ensures that structured priors reflect actual latent geometry, avoiding prior-posterior mismatch and guiding models toward disentangled representations.

Implementation: It supports both token-wise and feature-wise aggregation, and works out of the box with the C4-Code-20K subset for fast prototyping.

4.6 VSAEs vs SAEs in Small Language Models

1) Analogically how we define feature correlation (try different scopes of correlation definitions to see which the model behaves more sensitive to). 2) Variational + TopK first

To investigate our VSAEs, we perform comparative analysis of feature learning approaches on transformer model activations. We employ a sparse autoencoder (SAE) framework to extract interpretable features from the activations of a single-layer GELU transformer trained on the c4-code dataset from Neel Nanda. Our work follows the recent line of research on mechanistic interpretability methods based on dictionary learning.

The code can be leveraged to analyze larger language models as well, should time permit.

4.6.1 Methods

Model Architecture and Dataset. We extract activations from the post-MLP hook point of a single-layer GELU transformer model (`gelu-1l`) trained on c4-code. We utilize the `NeelNanda/c4-code-tokenized-2b` dataset for training our sparse autoencoders. The activations are collected at layer 0, using the hook point `blocks.0.mlp.hook_post`, which represents the fully processed information after non-linear transformations within the neural network layer.

For feature extraction, we implement two primary autoencoder variants:

1. A standard L1-regularized sparse autoencoder following methodology from Sam Marks' Dictionary Learning GitHub
2. A variational sparse autoencoder (V-SAE) with an isotropic prior distribution that encourages structured organization in the latent space, implemented through our `VSAEIsoTrainer` class and the `VSAEIsoGaussian` dictionary architecture

Both models are trained with varied dictionary sizes, derived as multiples ($4\times$, $8\times$, and $16\times$) of the model's hidden layer dimension, resulting in dictionary sizes of 8192, 16,384, and 32,768.

Training Procedure. The activation data is collected from the target model with a context length of 128 tokens, approximately 3,000 contexts, a refresh batch size of 32, and an output batch size of 1,024. The standard SAE is trained with a learning rate of 1×10^{-3}, an L1 penalty coefficient of 1×10^{-1}, 1,000 warmup steps, 1,000 sparsity warmup steps, and a total of 10,000 training steps. For the V-SAE, we use a learning rate of 3×10^{-4}, a KL divergence coefficient of 5.0, a warmup and sparsity warmup fraction of 0.05 of total steps, decay starting at 0.8 of total steps, and a total of 10,000 training steps.

For the V-SAE, we use a learning rate of 5×10^{-5}, a KL divergence coefficient of 5.0, a warmup fraction and sparsity warmup fraction of 0.05 of total steps, decay starting at 0.8 of total steps, a total of 20,000 training steps, and enable the April update mode.

All training is performed using bfloat16 precision on a single NVIDIA RTX 3080 GPU to balance computational efficiency and numerical stability. For both variants, we normalize activations during training, which has been shown to improve hyperparameter transfer between different layers and models.

Evaluation Framework. We evaluate our models using a comprehensive set of quantitative metrics that had already been implemented in our `evaluation.py` module:

1. **MSE loss**: The mean squared error between the original activations and their reconstructions, measuring reconstruction fidelity
2. **L1 loss**: The average L1 norm of feature activations, quantifying the overall sparsity of the representation
3. **L0 norm**: The average number of features active above threshold per token, providing a more intuitive measure of sparsity
4. **Percentage of neurons alive**: The fraction of features that activate on at least one token in a sample of random tokens, indicating the utilization of the learned dictionary
5. **Fraction of variance explained(!!!)**: The proportion of the total variance in the original activations captured by the reconstruction, measuring information preservation
6. **Relative reconstruction bias(!!!)**: A metric measuring potential systematic biases in the reconstruction
7. **Cross-entropy difference**: The change in cross-entropy loss when using reconstructed activations instead of original ones for next-token prediction
8. **Percentage of CE loss recovered**: The percentage of the model's predictive capability retained when using reconstructed activations, relative to zero-ablation baseline

These metrics provide a comprehensive assessment of both reconstruction quality and feature utility across our different autoencoder variants.

!!!We had to have a batch cutoff the reconstruction error to avoid out of memory errors, so this may not be accurate!!!

4.6.2 Feature Visualization and Interpretation

For qualitative analysis and interpretation of the learned features, we adapt the sae_vis framework through our `AutoEncoderAdapter` class. This adaptation enables us to generate interactive HTML visualizations that include activation histograms, feature correlation tables, and token-level attribution analysis.

This visualization framework allows for direct inspection of whether features correspond to coherent, distinct concepts within the model's representation space, providing crucial qualitative evidence for the monosemanticity of learned features.

4.6.3 Organized Latent Space Analysis

A key focus of our work is examining how the structure of the latent space affects feature quality and interpretability. The V-SAE's isotropic prior is designed to encourage a more organized latent space.

This structured approach stands in contrast to the standard SAE, which relies solely on L1 regularization to induce sparsity without explicitly organizing the latent space. We hypothesize that this organization will manifest as higher scores in the SAEBench evaluation framework, as well as in qualitative assessments of feature interpretability.

In future work, we aim to develop a specific methodology for measuring the quality of feature organization, focusing on how features span the activation space and how they relate to each other. This approach will build on ideas from feature splitting and absorption studies, examining how features transition from one concept to another throughout the latent space.

4.6.4 Benchmark Comparisons

While our initial implementation compares the standard SAE to the V-SAE model, the evaluation framework is built to support additional SAE variants. The codebase includes implementations of TopK SAEs with k-winners-take-all sparsity, Batch TopK SAEs with batch-level constraints, Matryoshka Batch TopK SAEs with nested hierarchical sparsity, Jump ReLU SAEs using jump ReLU activations, Gated SAEs with separate gating and magnitude networks, and Gated Anneal and P-Anneal variants that incorporate annealed training procedures.

For each model variant, we compute a comprehensive set of evaluation metrics using the SAEBench framework to standardize comparisons across architectures, focusing on feature quality and interpretability, training efficiency and stability, reconstruction performance, feature monosemanticity, and downstream task performance.

The outcomes of these comparisons will help establish whether the structured latent space approach of V-SAEs offers significant advantages over existing methods in terms of feature interpretability, while maintaining or improving reconstruction performance.

4.7 Comparison to KL-Regularized SAE

We include a KL-penalized SAE baseline [1] with fixed decoder and Gaussian prior. This allows us to isolate the benefit of full variational inference vs static KL constraints.

`TODO:` Implement KL-SAE variant and compare to V-SAE and TopK on both datasets.

`FIGURE:` *Loss curve comparison (KL-SAE vs V-SAE vs SAE).*
`TABLE:` *Disentanglement metrics comparison.*

Figure 1: TODO: Toy Model KL Curve KL divergence vs reconstruction loss on Toy Model 1 for SAE, V-SAE (iso), and V-SAE (structured).

Figure 2: TODO: Decoder Heatmap Decoder weight heatmaps. V-SAE (structured) shows clean, separated latent groups compared to baseline SAE.

Figure 3: TODO: Real Model Training Curve Training loss, KL term, and sparsity curve for V-SAE trained on GELU-1L TinyStories MLP output.

Figure 4: TODO: SAE-Vis Screenshot Top-token activations for selected V-SAE features using SAE-Vis on GELU-1L.

Table 1: TODO: Quantitative Comparison Table Comparison of V-SAE and SAE on reconstruction loss, sparsity, and feature overlap.

Model	Recon Loss ↓	L0 Sparsity ↓	Feature Overlap ↓
SAE (TopK)	-	-	-
V-SAE (Iso)	-	-	-
V-SAE (Structured)	**-**	**-**	**-**

5 Discussion

6 Conclusion

References

[1] Yilun Bao, Vincent Fortuin, Stephan Mandt, and Christopher J. Maddison. Revisiting end-to-end sparse autoencoder training: A short finetune is all you need. *arXiv preprint arXiv:2503.17272*, 2024. URL https://arxiv.org/abs/2503.17272.

[2] Jonas Bussmann, Roger Grosse, Tom Bricken, and Catherine Olsson. Atomic latents in meta-saes: Identifiability in sparse decomposition. *arXiv preprint arXiv:2403.00709*, 2024. URL https://arxiv.org/abs/2403.00709.

[3] Henrik Carlsson, Klas Ekvall, and Fredrik D. Johansson. Identifying functionally important features with end-to-end sparse dictionary learning. *arXiv preprint arXiv:2405.12241*, 2024. URL https://arxiv.org/abs/2405.12241.

[4] Jack Cunningham, Neel Nanda, Lukas Lieberum, et al. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2306.05379*, 2023. URL https://arxiv.org/abs/2306.05379.

[5] Nelson Elhage, Neel Nanda, Catherine Olsson, et al. Toy models of superposition. *Transformer Circuits Thread*, 2022. URL https://transformer-circuits.pub/2022/toy_model/index.html.

[6] Dario Fuoli, Luc Van Gool, and Radu Timofte. Fourier space losses for efficient perceptual image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2360–2369. IEEE, 2021.

[7] Tianyu Gao, Yuxiao Li, Neel Nanda, and Nelson Elhage. Scaling sparse autoencoders: Challenges and insights. *arXiv preprint arXiv:2404.06624*, 2024. URL https://arxiv.org/abs/2404.06624.

[8] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision (ECCV)*, pages 694–711. Springer, 2016.

[9] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. URL https://arxiv.org/abs/1312.6114.

[10] Antti Rasmus, Harri Valpola, Mikko Honkala, Mathias Berglund, and Tapani Raiko. Semi-supervised learning with ladder networks. *arXiv preprint arXiv:1507.02672*, 2015. URL https://arxiv.org/abs/1507.02672.

[11] Casper Kaae Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby, and Ole Winther. Ladder variational autoencoders. *arXiv preprint arXiv:1602.02282*, 2016. URL https://arxiv.org/abs/1602.02282.

[12] Arash Vahdat and Jan Kautz. NVAE: A deep hierarchical variational autoencoder. *arXiv preprint arXiv:2007.03898*, 2020. URL https://arxiv.org/abs/2007.03898.

[13] Harri Valpola. From neural pca to deep unsupervised learning. *arXiv preprint arXiv:1411.7783*, 2014. URL https://arxiv.org/abs/1411.7783.

[14] Hao Zhou, Henrik Carlsson, and Fredrik D. Johansson. Sparse dictionary size controls feature granularity and stability. *arXiv preprint arXiv:2410.07656*, 2024. URL https://arxiv.org/abs/2410.07656.

Acknowledgements

We thank the SPAR 2025 program for guidance.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: **[TODO]**

   Justification: **[TODO]**

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: **[TODO]**

   Justification: **[TODO]**

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: **[TODO]**

    Justification: **[TODO]**

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: **[TODO]**

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.