SEPARATING INTERPRETABILITY AND RECONSTRUCTION OBJECTIVES IN SPARSE AUTOENCODERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Sparse Autoencoders (SAEs) are effective tools for mechanistic interpretability by decomposing neural representations into sparse and interpretable features. However, a single sparse representation in traditional SAEs is responsible for both reconstruction fidelity and interpretability, which are not guaranteed to align. To separate these objectives, we propose **DualSAE**, a modular framework that augments standard SAEs with a plug-in residual branch. A standard SAE serves as the *Sparse Branch*, learning a sparse code that contains the dominant semantics of the input. The added *Residual Branch* models the non-semantic information omitted by the sparse branch. We ensure that the contributions of the two branches towards the overall reconstruction have minimal overlap through a decorrelation loss. We also adopt a two-stage training strategy that first stabilizes sparse representation learning before coupling with the residual branch. Across diverse encoder–SAE configurations on two vision datasets and two language corpora, DualSAE achieves mean relative improvements of approximately 29% in reconstruction fidelity and 34% in interpretability (through the monosemanticity score). Moreover, with more interpretable representations, DualSAE yields consistent performance gains in the downstream applications of topic modeling, hypothesis generation, and LLM steering. Qualitative analysis further suggests that the Sparse Branch indeed preserves dominant semantic content, whereas the Residual Branch captures non-semantic information to improve reconstruction fidelity.

1 INTRODUCTION

Sparse autoencoders (SAEs) decompose high-dimensional neural activations into sparse, interpretable latent features while preserving enough information for reconstruction (Bricken et al., 2023; Huben et al., 2024; Gao et al., 2025; Templeton et al., 2024). Ideally, each latent captures a coherent and interpretable concept (Olshausen & Field, 1996; Lee et al., 2006; Mairal et al., 2009; Makhzani & Frey, 2014). However, in standard SAEs (Figure 1a), the same sparse code is responsible for both

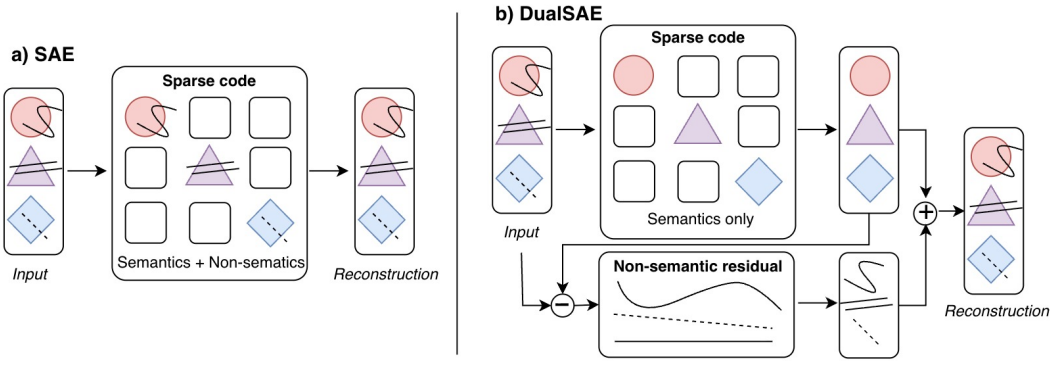

Figure 1: **Standard SAE and DualSAE comparison**. (a) A standard SAE encodes both dominant semantic concepts and non-semantic information into a single sparse code, entangling interpretability with reconstruction. (b) Our DualSAE disentangles these objectives by adding a non-semantic residual branch. This design separates interpretability from the overall reconstruction objective.

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

interpretability and reconstruction (Gao et al., 2025; Bussmann et al., 2024; Rajamanoharan et al., 2024b; Cui et al., 2026). These two objectives are not guaranteed to align: reconstruction favors any useful information, including non-semantic information that may not correspond to interpretable concepts. As a result, using a reconstruction objective to learn interpretable features can force latents to mix multiple factors, creating a trade-off between interpretability and reconstruction fidelity. We argue that this tension stems from requiring one sparse representation to fulfill two distinct roles.

To address the above problem, we propose **DualSAE**, a modular framework that separates semantic information from non-semantic residual information through a dual-branch architecture (Figure 1b). Architecturally, DualSAE extends a standard SAE by adding a residual branch. The *Sparse Branch*, modeled as a standard SAE, encodes the input into a sparse code and is responsible for capturing the dominant semantic information. In contrast, the *Residual Branch* is not designed for interpretability. Instead, its role is to model the remaining non-semantic information omitted by the Sparse Branch that is still useful for accurate reconstruction. This decomposition separates the representational responsibilities of the two branches: the Sparse Branch focuses on learning sparse, semantically coherent features, while the Residual Branch captures non-semantic residual information required for faithful reconstruction.

Empirically, by plugging the residual branch into standard SAEs, the DualSAE significantly improves both interpretability by 34% and reconstruction fidelity by 29% on various encoder–SAE configurations as shown by experiments on two vision datasets (ImageNet-1K (Russakovsky et al., 2015) and OpenImagesV7 (Kuznetsova et al., 2020)) and two language corpora (OpenWebText (Gokaslan et al., 2019) and GitHub code (CodeParrot, 2022)). The more interpretable representation from DualSAE also improves performance on two vision (topic modeling and hypothesis generation) and one language (LLM steering) downstream applications, with qualitative results supporting enhanced interpretability. In summary, our main contributions are:

1. **DualSAE**, a modular framework that augments standard SAEs with a plug-in residual branch to separate interpretability from the overall reconstruction objective.
2. A **decorrelation loss** to ensure that the contributions of the two branches towards the overall reconstruction have minimal overlap. This is done by first stabilizing the Sparse Branch during the warm-up stage and then training the Residual Branch with the decorrelation loss.
3. **Quantitative and qualitative evidence**, in both vision and natural language domains, showing that DualSAE significantly improves reconstruction fidelity and interpretability.

2 Related Works

Sparse Autoencoders. Sparse Autoencoders (SAEs) decompose dense activations into sparse, inspectable features (Bricken et al., 2023; Gao et al., 2025; Rajamanoharan et al., 2024b; Nasiri-Sarvi et al., 2026), which is motivated by the superposition hypothesis (Elhage et al., 2022). Existing SAEs employ different sparsification mechanisms, such as TopK SAE (Gao et al., 2025; Makhzani & Frey, 2014), BatchTopK SAE (Bussmann et al., 2024), ReLU SAE (Ng, 2011), and JumpReLU SAE (Rajamanoharan et al., 2024b). Despite architectural differences, these methods reconstruct activations from a single sparse code, coupling interpretability with reconstruction.

Structure and Regularization in SAEs. Various methods introduce structure or additional regularization into SAE representations. Gated SAEs (Rajamanoharan et al., 2024a) explicitly separate feature detection from magnitude estimation, applying the sparsity penalty to the gating pathway to mitigate shrinkage and improve the sparsity–reconstruction trade-off. Matryoshka SAE (Bussmann et al., 2025) jointly trains nested dictionaries with reconstruction objectives at multiple sizes, encouraging general and specific features while mitigating feature splitting and absorption. Ort-SAE (Korznikov et al., 2026) penalizes high pairwise cosine similarities between decoder vectors to reduce feature absorption and composition. AlignSAE (Yang et al., 2026) uses supervised post-training to bind predefined concepts to dedicated latent slots while preserving free slots for general reconstruction. CB-SAE (Kulkarni et al., 2026) instead adopts a post-hoc approach that prunes low-utility SAE features and augments the latent space with a lightweight concept bottleneck aligned with user-defined concepts. These methods restructure, regularize, or explicitly supervise the SAE latent space to improve feature quality. In contrast, DualSAE introduces a residual branch for non-semantic information, which separates the objectives of interpretability and reconstruction.

3 MOTIVATION

Standard SAEs use a single sparse representation for both interpretability and reconstruction fidelity. There is often a tradeoff between these two objectives. To study this tension, let x denote an image embedding, $\hat{x}$ its SAE reconstruction, and $r = x - \hat{x}$ the residual. We examine whether $\hat{x}$ mostly captures the dominant semantic structure, while non-semantic information remains in r.

To test the above hypothesis, we adopt BatchTopK SAE on CLIP embeddings from ImageNet-1K using nearest-neighbor retrieval with $\hat{x}$ and r as queries. As shown in Figure 2 (left), $\hat{x}$ achieves substantially higher class purity (the percentage of samples of the dominant class). Qualitatively, given a *dog* query image, $\hat{x}$ retrieves semantically consistent *dog* samples (top-right), whereas r retrieves unrelated objects (bottom-right). This suggests that the sparse reconstruction captures the dominant semantic structure, while the residual contains non-semantic information. This observation motivates our design to separate these two types of information rather than forcing them into a single sparse representation.

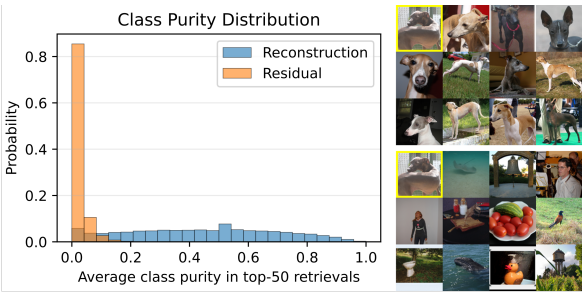

Figure 2: **Left:** Distribution of class purity for nearest neighbors retrieved using reconstruction $\hat{x}$ and residual r from BatchTopK SAE. Retrievals from $\hat{x}$ are substantially purer than those from r. **Right:** Example retrievals for the same query image using $\hat{x}$ (top) and r (bottom). The reconstruction $\hat{x}$ shows consistent semantics, while the residual is non-semantic.

4 DUALSAE

Sparse Branch. The branch is designed similarly to standard SAEs. Given an input feature $x \in \mathbb{R}^d$, it produces an overcomplete sparse latent code $z \in \mathbb{R}^m$ as $z = S(W_E x + b_E)$, and reconstructs the input as $x^M = W_D z + b_D$, where $S(\cdot)$ denotes the sparsification rule, $W_E \in \mathbb{R}^{m \times d}$ is the encoder, and $W_D \in \mathbb{R}^{d \times m}$ is the decoder ($m \gg d$). As a result, x^M is the Sparse Branch reconstruction in the original feature space, containing the dominant semantic information.

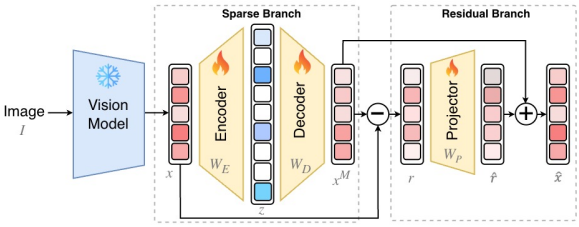

Figure 3: **DualSAE architecture**. The *Sparse Branch*, modeled as a standard SAE, captures semantic information of the input feature x in the sparse latent z, then projects it back to the original dimension as x^M. The augmented *Residual Branch* captures the non-semantic information $\hat{r}$. The final reconstructed feature combines the outputs of both branches: $\hat{x} = x^M + \hat{r}$.

Residual Branch. We do not require x^M to reconstruct all information in x, as x may contain non-semantic information that should not be learned by sparse interpretable features. We therefore introduce the Residual Branch that models the non-semantic information. Specifically, we define the residual as $r = x - x^M$, and let the Residual Branch predict $\hat{r} = Q(W_P r)$, where W_P is a projector and $Q(\cdot)$ is a nonlinear operator. The final reconstruction combines two branches as $\hat{x} = x^M + \hat{r}$.

Role of each branch. The Residual Branch is not designed to be interpretable. Instead, it provides a separate branch for non-semantic information (Figure 2 bottom-right) that is useful for reconstruction. In a standard SAE, both dominant semantic and non-semantic information must be modeled by the single sparse code (Figure 1a), forcing the Sparse Branch to serve both interpretability and reconstruction objectives. In DualSAE, the non-semantic information of the Residual Branch is separated from semantic information in the Sparse Branch (Figure 1b), allowing the sparse representations to have higher interpretability without being directly constrained by the reconstruction objective.

Reconstruction objective. We use the standard L2 error to control the reconstruction quality $\mathcal{L}_{\text{rec}} = \|x - \hat{x}\|_2^2$. As mentioned above, we do not force x^M alone to match x perfectly. Instead, we combine the outputs of both branches to obtain the reconstructed feature $\hat{x}$. In other words, the reconstruction constraint now applies to the combined sparse reconstruction and residual, rather than to the sparse reconstruction alone. This is the key design difference from traditional SAEs.

5 Discouraging Information Overlap Across Branches

Problem formulation. Let the Sparse Branch produce $x^M = W_D z^M(x)$, define the residual as $r = x - x^M$, and let the Residual Branch produce $\hat{r} = Q(W_P r) := P_{\theta_P}(r)$, where P_{θ_P} is an arbitrary nonlinear function. The final reconstruction is $\hat{x} = x^M + \hat{r}$. The reconstruction error is

$$\hat{x} - x = P_{\theta_P}(r) - r, \qquad \mathcal{L}_{\text{rec}} = \mathbb{E}_x \left[\|P_{\theta_P}(r) - r\|_2^2 \right]. \tag{1}$$

This exposes a shortcut: regardless of the quality of x^M, the Residual Branch can minimize the reconstruction loss by learning the *identity map* $\hat{r} = r$ on the residual distribution. Consequently, reconstruction alone does not require the Sparse Branch to capture meaningful information.

Information overlap. Let $\mathcal{A}(x) = \{j : z_j(x) > 0\}$ denote the active sparse latents and let $D_x = W_D[:, \mathcal{A}(x)]$ contain the active decoder directions. These directions define the sample-dependent sparse subspace $\mathcal{S}_M(x) = \text{span}(D_x)$. If the residual $\hat{r}$ aligns strongly with $\mathcal{S}_M$, the Residual Branch can overwrite information already represented by the active sparse features. To discourage this overlap, we penalize the correlations between $\hat{r}$ and the active decoder directions:

$$\mathcal{L}_{\text{dcor}} = \|D_x^\top \hat{r}\|_2^2. \tag{2}$$

Effect on the identity map. We consider the objective with respect to the Residual Branch output: $\ell_r(\hat{r}) = \|\hat{r} - r\|_2^2 + \gamma \|D_x^\top \hat{r}\|_2^2$. At the identity solution $\hat{r} = r$, the reconstruction gradient vanishes, whereas $\nabla_{\hat{r}} \ell_r(\hat{r})|_{\hat{r}=r} = 2\gamma D_x D_x^\top r$. This gradient is nonzero whenever $D_x^\top r \neq 0$, i.e., whenever the residual contains components aligned with the active sparse decoder directions. Thus, the identity map $\hat{r} = r$ is not stationary in output space.

Training objective. With the decorrelation loss, the final training loss of DualSAE is

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \lambda \mathcal{L}_{\text{sparse}} + \gamma \mathcal{L}_{\text{dcor}}, \tag{3}$$

where λ and γ control the strengths of the sparsity and the decorrelation constraints, respectively. Note that for hard-sparsity methods, e.g. TopK and BatchTopK, the sparsity loss $\mathcal{L}_{\text{sparse}}$ is not used.

6 Two-stage training

We train DualSAE with a two-stage schedule to stabilize the sparse dictionary before introducing the Residual Branch (Algorithm 1). During the warm-up stage, the Sparse Branch is optimized as a standard SAE, allowing it to capture the dominant semantic information of the input without interference from the Residual Branch. After warm-up, we activate the Residual Branch and the decorrelation loss $\mathcal{L}_{\text{dcor}}$, encouraging the Residual Branch to model non-semantic information while discouraging it from reusing active sparse decoder directions.

Algorithm 1: Two-stage training

Stage 1: Train Sparse Branch
for *stage-1 steps* **do**
  $z \leftarrow S(W_E x + b_E)$
  $x^M \leftarrow W_D z + b_D$
  $\mathcal{L} \leftarrow \mathcal{L}_{\text{rec}}(x, x^M) + \lambda \mathcal{L}_{\text{sparse}}$
  Update W_E, W_D
Stage 2: Train DualSAE
for *stage-2 steps* **do**
  $z \leftarrow S(W_E x + b_E)$
  $x^M \leftarrow W_D z + b_D$
  $\hat{r} \leftarrow Q(W_P(x - x^M))$
  $\hat{x} \leftarrow x^M + \hat{r}$
  $\mathcal{L} \leftarrow \mathcal{L}_{\text{rec}} + \lambda \mathcal{L}_{\text{sparse}} + \gamma \mathcal{L}_{\text{dcor}}$
  Update W_E, W_D, W_P

We also measure the reconstruction quality (R^2) and interpretability (MonoScore) under different warm-up lengths of DualBatchTopK on ImageNet-1K, illustrated in Figure 4. With little or no warm-up, reconstruction fidelity improves rapidly, but MonoScore remains weaker. A moderate warm-up provides a better balance: the model reaches better reconstruction while maintaining stronger monosemanticity. In contrast, an excessively long warm-up delays reconstruction improvement and does not further improve MonoScore, suggesting that the Residual Branch should be introduced after the Sparse Branch has sufficiently stabilized, while avoiding an unnecessarily long warm-up.

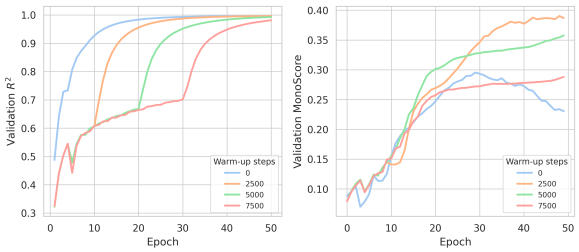

Figure 4: **Effect of warm-up length**, showing reconstruction R^2 (left) and MonoScore (right) during ImageNet-1K training. Shorter warm-up accelerates reconstruction gains; moderate warm-up improves monosemanticity. Excessive warm-up delays reconstruction gains without further improving MonoScore.

7 Experimental Analysis

Implementation details. We adopt features of ImageNet-1K (Russakovsky et al., 2015) and OpenImagesV7 (Kuznetsova et al., 2020) from three encoders: CLIP (Radford et al., 2021), SigLIP2 (Tschannen et al., 2025), and ViT-L/16 (Dosovitskiy et al., 2021). For language experiments, we obtained activations from Pythia on OpenWebText (Gokaslan et al., 2019) and GitHub code (CodeParrot, 2022). We evaluate six different SAE variants: TopK (Makhzani & Frey, 2014), BatchTopK (Bussmann et al., 2024), JumpReLU (Rajamanoharan et al., 2024b), GatedSAE (Rajamanoharan et al., 2024a), MatryoshkaSAE (Bussmann et al., 2025), and CB-SAE (Kulkarni et al., 2026) and their DualSAE counterparts. All quantitative results are averaged across three random seeds. Please refer to Appendices A and C for detailed experimental settings and metric definitions.

Experiments 1–7 analyze the behavior and effectiveness of DualSAE. We then evaluate its application to topic modeling in experiment 8 and assess language representations and LLM steering in experiment 9. Please refer to Appendix B.4 for results on hypothesis generation.

1. Reconstruction and decomposition analysis. We use t-SNE (Van der Maaten & Hinton, 2008) to examine both reconstruction fidelity and the internal decomposition learned by DualSAE. In Figure 5(a), the standard BatchTopK SAE shows visible displacement between original embeddings x and reconstructed embeddings $\hat{x}$, indicating weaker reconstruction fidelity. In contrast, Dual-BatchTopK produces much tighter overlap, suggesting that the residual branch improves reconstruction fidelity. Figure 5(b) further visualizes x^M, $\hat{r}$, and $\hat{x}$ for three ImageNet-1K classes. The sparse reconstruction x^M forms clear class-specific clusters (bottom-left), showing that the Sparse Branch captures the dominant semantic structure. However, these clusters remain separated from the corresponding $\hat{x}$ clusters (middle-top), indicating that x^M alone does not fully recover the original embedding. The residual clusters $\hat{r}$ (bottom-right) are more closely grouped because, once the dominant semantics are captured by x^M, the Residual Branch models non-semantic information. The results support that (1) *the Residual Branch improves reconstruction fidelity*, and (2) *the Sparse Branch captures interpretable semantics, while the Residual Branch provides non-semantic information for more faithful reconstruction*.

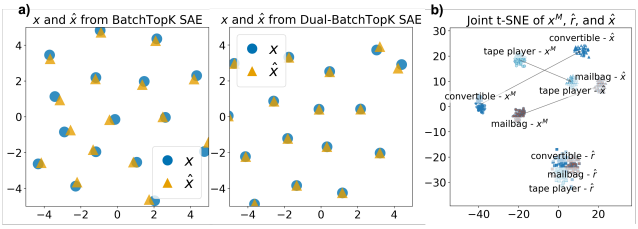

Figure 5: (a) **Input–reconstruction alignment.** Dual-BatchTopK preserves the original embedding geometry better than BatchTopK, shown by the overlap between input x and reconstructed $\hat{x}$. (b) **Semantic–residual decomposition.** The class-separated x^M clusters (bottom-left) capture dominant semantics but remain far from the reconstructed $\hat{x}$ (middle-top). The closely grouped residual $\hat{r}$ (bottom-right) captures non-semantic information, which is necessary for complete reconstruction.

2. Residual branch vs. More trainable parameters. DualSAE contains additional parameters through its residual branch. To determine whether its improvements arise solely from this increased capacity, we construct parameter-matched SAE baselines by increasing their sparse-code dimension until their total number of trainable parameters matches that of DualSAE. All other training and evaluation settings are kept unchanged. Across all settings, performance decreases when replacing DualSAE with SAEs. Therefore, simply allocating the same parameter budget to a wider sparse dictionary does not reproduce DualSAE's advantages. These results suggest that *the benefit of the Residual Branch is not merely to provide more parameters, but to provide a separate representational pathway for non-semantic information.*

Table 1: **Parameter-matched SAE comparison.** Each baseline uses a wider sparse dictionary so that its number of trainable parameters matches that of DualSAE. We report $\Delta\text{metric} = \text{metric}_{\text{matched SAE}} - \text{metric}_{\text{DualSAE}}$; negative values favor DualSAE. DualSAE outperforms the parameter-matched baselines in every setting, indicating that its gains are not simply due to additional parameters.

Encoder	Activation	Par.	ImageNet-1K					OpenImagesV7				
			ΔR^2	ΔMS	ΔPur.	ΔKNN	ΔLP	ΔR^2	ΔMS	ΔPur.	ΔKNN	ΔLP
ViT	TopK	18M	-0.002	-0.004	-0.004	-0.003	-0.007	-0.006	-0.001	-0.001	-0.009	-0.001
	BatchTopK	18M	-0.004	-0.003	-0.006	-0.009	-0.001	-0.003	-0.001	-0.002	-0.002	-0.008
	JumpReLU	18M	-0.113	-0.246	-0.235	-0.003	-0.002	-0.251	-0.059	-0.086	-0.019	-0.034
CLIP	TopK	13M	-0.003	-0.003	-0.002	-0.001	-0.005	-0.010	-0.009	-0.001	-0.011	-0.008
	BatchTopK	13M	-0.009	-0.001	-0.001	-0.001	-0.001	-0.001	-0.002	-0.007	-0.008	-0.001
	JumpReLU	13M	-0.137	-0.113	-0.179	-0.013	-0.017	-0.144	-0.039	-0.116	-0.004	-0.019
SigLIP2	TopK	20M	-0.006	-0.002	-0.002	-0.001	-0.004	-0.001	-0.001	-0.005	-0.009	-0.002
	BatchTopK	20M	-0.001	-0.006	-0.012	-0.001	-0.008	-0.002	-0.003	-0.004	-0.006	-0.001
	JumpReLU	20M	-0.119	-0.050	-0.245	-0.005	-0.009	-0.040	-0.006	-0.105	-0.017	-0.010

3. Reconstruction fidelity and interpretability evaluation. Figure 6 compares the six SAE variants with their DualSAE counterparts on ImageNet-1K across three encoders. We report five metrics: reconstruction quality R^2, MonoScore (MS) (Pach et al., 2025), class purity (Nasiri-Sarvi et al., 2026), KNN accuracy, and linear probing (LP) accuracy. Across the 18 settings, DualSAE generally expands the radar profile relative to the corresponding SAE baselines. The gains are particularly consistent in reconstruction fidelity R^2, KNN accuracy, and LP accuracy, indicating that adding the Residual Branch improves reconstruction fidelity. DualSAE also consistently improves MonoScore and class purity, showing that the additional reconstruction pathway does not compromise interpretability and can instead improve the semantic coherence of the sparse representation. DualSAE also substantially enhances the reconstruction quality and interpretability on OpenImagesV7, shown in Appendix B.2. Moreover, we also observe the advantage of DualSAE on natural language representations of two corpora (OpenWebText and GitHub code), presented in Appendix C. These results show that *separating semantic information from non-semantic information enables simultaneous improvements in reconstruction fidelity and interpretability.*

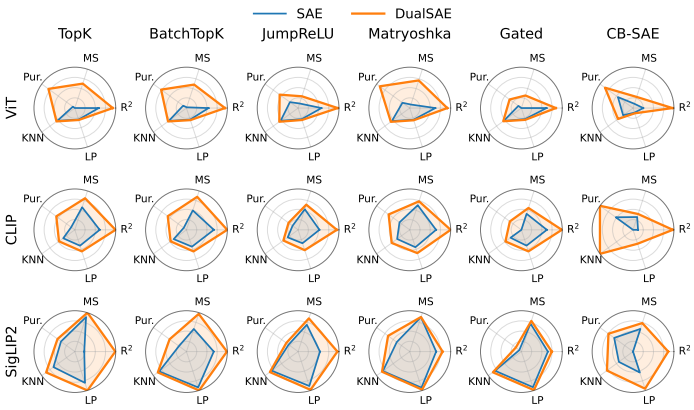

Figure 6: **Comparison of SAE and DualSAE on ImageNet-1K.** Radar plots compare each SAE variant with its DualSAE counterpart across three encoders and six SAE variants using reconstruction R^2, MonoScore, class purity, KNN accuracy, and LP accuracy. Across 18 settings, DualSAE generally improves reconstruction fidelity and interpretability. OpenImagesV7 results are shown in Figure 13.

4. Interpretability of sparse latents. We examine whether the improvements in MonoScore and class purity actually correspond to more interpretable individual latents. Figure 7 compares BatchTopK and Dual-BatchTopK trained on SigLIP2 features from ImageNet-1K. We select one latent from each model at the same MonoScore rank and visualize samples with high, medium, and low activation strengths (left). The BatchTopK latent is only partially coherent: its strongest activations mostly correspond to images of *snakes*, whereas lower-activation samples include unrelated objects, such as *vehicle* and *factory*. In contrast, the matched Dual-BatchTopK latent consistently activates on *red mushrooms* across the full activation range. This indicates that DualSAE obtains sparse latents whose semantic meaning remains stable beyond the highest-activation samples. Moreover, the MonoScore and class-purity curves across all latents (right) quantitatively show that this behavior is not limited to a single latent. These results suggest that *the sparse representation of DualSAE obtains higher interpretability via higher semantic coherence.*

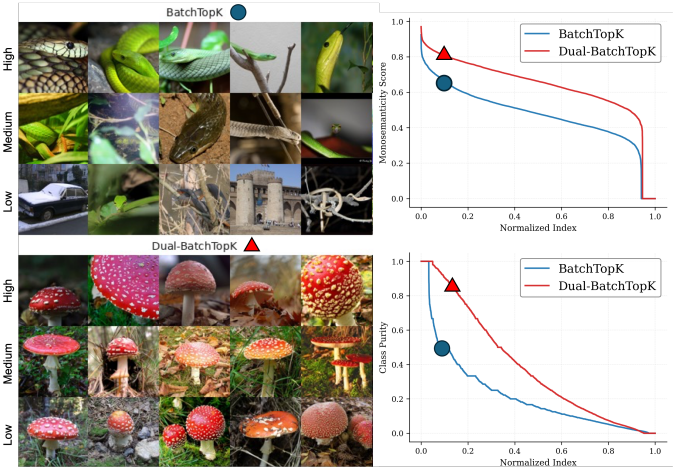

Figure 7: **Improved interpretability of sparse latents with DualSAE.** Left: For two latent dimensions selected at the same MonoScore rank, Dual-BatchTopK shows more coherent activating samples from high to low activations compared to BatchTopK. Right: the panels report MonoScore (top) and class purity (bottom) across all latent dimensions, showing consistently higher interpretability for DualSAE.

5. Effect of decorrelation loss. We use nearest-neighbor retrieval to examine whether the decorrelation loss improves separation between the two branches. Figure 8 compares retrievals from x^M and $\hat{r}$ for Dual-BatchTopK trained on CLIP embeddings on ImageNet-1K. The intuition is that x^M should capture the dominant concept of the query, while $\hat{r}$ should focus on information not already explained by the sparse reconstruction. Without the decorrelation loss, x^M retrieves coherent *bird* images, while $\hat{r}$ also retrieves several visually similar *birds*. This overlap suggests that the Residual Branch partially reuses information already captured by the Sparse Branch rather than focusing on non-semantic information. With the decorrelation loss, retrievals from x^M remain semantically coherent and continue to capture the dominant *bird* concept, whereas retrievals from $\hat{r}$ become more diverse and less redundant with those from x^M. This indicates that the decorrelation loss discourages the non-semantic residual from aligning with active sparse decoder directions. We further conduct a quantitative ablation by removing the decorrelation loss while keeping other parameters unchanged (Appendix B.1), which leads to performance degradation across all settings. These results show that *the decorrelation loss discourages branch overlap, encouraging the Sparse and Residual Branches to capture semantic and non-semantic information, respectively.*

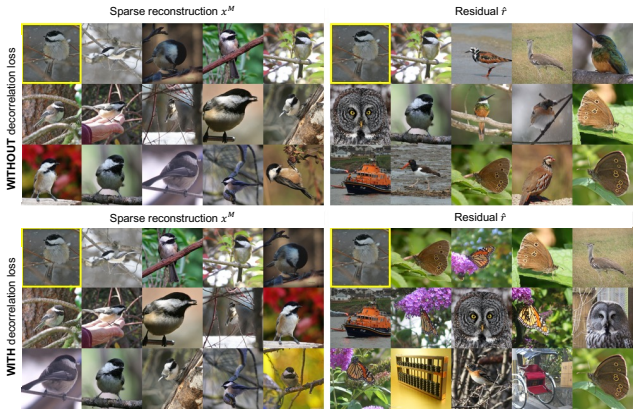

Figure 8: **Effect of decorrelation loss on branch separation.** For the same query image, we compare nearest-neighbor retrievals from the sparse reconstruction x^M and residual $\hat{r}$, with and without the decorrelation loss. The loss preserves coherent retrievals from x^M while making $\hat{r}$ retrievals less similar to the query.

6. Semantic neighborhoods in sparse latent space. We use t-SNE (Van der Maaten & Hinton, 2008) to qualitatively compare the sparse latent spaces of Dual-JumpReLU and standard JumpReLU across 20 randomly selected ImageNet-1K classes. Unlike the previous reconstruction analysis, this experiment directly examines the learned sparse code z, rather than the reconstructed embedding $\hat{x}$. The goal is to assess whether the sparse representation organizes images into meaningful semantic neighborhoods. Intuitively, if z captures coherent semantic structure, semantically related classes should occupy nearby regions, while unrelated classes should remain separated. As shown in Figure 9, the DualSAE latent space exhibits clearer and more coherent local neighborhoods. Related categories occupy more structured regions, suggesting that the Sparse Branch captures dominant semantics more cleanly. In contrast, the baseline exhibits stronger central crowding, with semantically different classes appearing more mixed in the latent space. This suggests that a single sparse bottleneck may entangle semantic and non-semantic information within the same representation, resulting in a less organized sparse code. These results suggest that, *by assigning residual information to the Residual Branch, DualSAE allows the sparse latent code to focus more strongly on coherent semantic structure.*

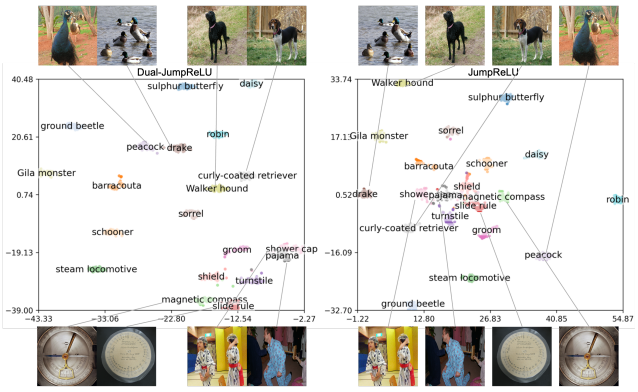

Figure 9: **t-SNE visualization of sparse representations.** We visualize the sparse codes of 20 randomly selected ImageNet-1K classes for Dual-JumpReLU and JumpReLU SAE. DualSAE exhibits better separated local semantic neighborhoods compared to the centrally crowded and intermixed semantic classes of the baseline.

7. Reconstruction refinement during training. We examine how the Residual Branch affects image retrieval during training. For a fixed query image, we compare its similarity to other images before and after adding the residual to its reconstructed feature $\hat{x}$. At five different checkpoints during training, we show the images with the largest increase in similarity to the query. Figure 10 shows a clear progression. During warm-up, before the Residual Branch is trained, the promoted images are noisy at the first checkpoint and then capture the main semantics (*dog*) at the second checkpoint. After warm-up, as the Residual Branch is optimized, the promoted images become increasingly aligned. They match not only the broad *dog* category, but also finer visual attributes of the query. This behavior supports the motivation for two-stage training: the Sparse Branch first establishes the dominant semantic structure, after which the Residual Branch learns non-semantic information that remains unexplained by the sparse reconstruction. *These observations are consistent with our design: the Sparse Branch captures the dominant semantic structure, while the Residual Branch adds non-semantic information for more faithful reconstruction.*

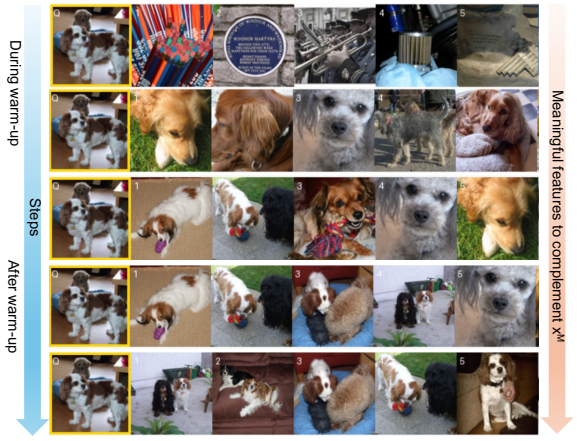

Figure 10: **Reconstruction refinement during training.** For a fixed query image, each row shows the images whose similarity to the query increases the most after adding the residual to the sparse reconstruction at a given training checkpoint. During warm-up, the retrieved images shift from unrelated objects to dogs, matching the query's dominant semantics. After warm-up, they increasingly match finer visual attributes of the query, including coat color and texture.

8. SAE as a topic model. We assess whether DualSAE's sparse representation yields better topic modeling than conventional SAEs. The intuition is that each sparse SAE feature can be viewed as a *topic atom*: for a given image, its active features indicate which visual concepts are present and their activation magnitudes reflect their importance. We apply the same topic-modeling pipeline (Girrbach & Akata, 2026) to standard SAE features and DualSAE's sparse features, so that differences in the resulting topics primarily reflect the quality of their learned representations. Figure 11 presents an example of TopK, BatchTopK, and their DualSAE counterparts. Across both SAE variants, DualSAE produces concrete labels that capture the dominant object, attribute, or scene, such as *Shiny Metal Object* for a metal serving set. In contrast, the standard SAE frequently assigns topics based on weak or spurious associations, e.g., the same example is labeled *Ceramic Containers*. Similar discrepancies appear for color, where DualSAE identifies *Vibrant Yellow Patterns* and *Vibrant Color Palette*, while SAE returns unrelated topics such as *Time-Driven Motion* and *Cozy Child Comfort*. We also conduct quantitative evaluation on ImageNet-1K and OpenImagesV7 for a comprehensive comparison. Table 7 in the appendix quantitatively shows that DualSAE obtains substantially higher performance in almost all settings. Please refer to Appendix B.3 for detailed experimental settings, quantitative results, and more qualitative examples. These results suggest that *separating the dominant semantic structure from residual information yields more concrete, visually grounded, and interpretable topics for downstream topic modeling.*



Figure 11: **Image–topic assignment comparison**. Labels show the dominant topic using DualSAE and SAE representations. DualSAE yields visually grounded topics, while the standard SAE assigns unrelated concepts.

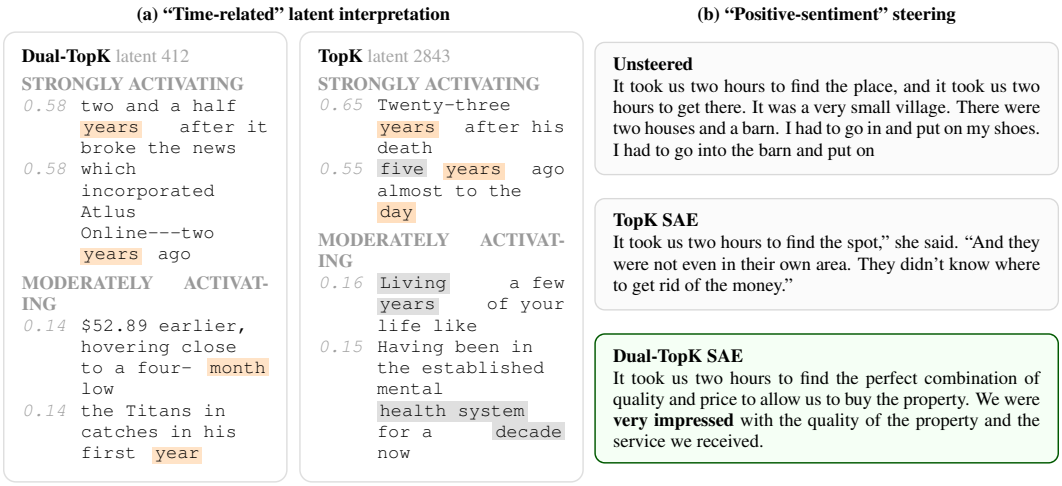

Figure 12: **Qualitative analysis on language representations. (a)** "Time-related" latents learned by Dual-TopK and TopK. Dual-TopK responds more selectively to genuine "time-related" concepts, whereas TopK exhibits missed temporal concepts and activations on unrelated tokens. **(b)** Positive-sentiment steering. Compared with the unsteered generation and TopK steering, Dual-TopK produces a coherent flow containing clear positive-sentiment expressions.

9. Language representation evaluation. We further evaluate DualSAE on language-model representations using Pythia activations on OpenWebText (OWT) and GitHub code (GHC). We evaluate SAEs' ability to preserve the original language-model behavior using ΔCE and KL divergence, reconstruct activations using R^2, and learn interpretable features using MonoScore (MS). As shown in Table 2, DualSAE consistently improves reconstruction quality and interpretability while substantially reducing perturbations to the original model across both corpora, suggesting that DualSAE's advantages generalize beyond vision representations to both natural language and code. Moreover, we also provide qualitative examples for activating pattern and LLM steering. In Figure 12a, both TopK and Dual-TopK learn latents associated with temporal expressions, but the Dual-TopK latent responds more selectively to genuine time-related concepts such as *years* and *months*. In contrast, TopK SAE latent misses some temporal concepts and also activates on unrelated tokens. Figure 12b further illustrates feature steering using a *positive-sentiment* latent to modify Pythia's activations at layer 12. While the unsteered and TopK-steered generations do not exhibit a clear positive shift, steering with Dual-TopK produces a coherent continuation containing explicit positive expressions. Please refer to Appendix C for the detailed setup, metric definition, and more qualitative examples. Together, these results indicate that *DualSAE's advantages generalize to language activations and LLM steering.*

Table 2: **Language-model evaluation** on OpenWebText (OWT) and GitHub code (GHC) at block 12 of Pythia. We compare standard SAE variants with their DualSAE counterparts. Lower ΔCE and KL indicate better preservation of the original language-model behavior, while higher R^2 and MS indicate better reconstruction and monosemanticity.

Corpus	Method	ΔCE ↓	KL ↓	R^2 ↑	MS ↑
OWT	JumpReLU	$0.7412_{0.0201}$	$0.5498_{0.0011}$	$0.8994_{0.1209}$	$0.4995_{0.1225}$
	Dual-JumpReLU	$\mathbf{0.1423}_{\mathbf{0.0097}}$	$\mathbf{0.0532}_{\mathbf{0.0101}}$	$\mathbf{0.9732}_{\mathbf{0.0464}}$	$\mathbf{0.6882}_{\mathbf{0.0743}}$
	TopK	$0.5538_{0.0193}$	$0.5608_{0.0201}$	$0.9677_{0.0559}$	$0.4317_{0.0541}$
	Dual-TopK	$\mathbf{0.0105}_{\mathbf{0.0022}}$	$\mathbf{0.0131}_{\mathbf{0.0016}}$	$\mathbf{0.9896}_{\mathbf{0.0180}}$	$\mathbf{0.6247}_{\mathbf{0.0236}}$
	BatchTopK	$0.6587_{0.0442}$	$0.6661_{0.0441}$	$0.9432_{0.0984}$	$0.4982_{0.1455}$
	Dual-BatchTopK	$\mathbf{0.2118}_{\mathbf{0.0229}}$	$\mathbf{0.3129}_{\mathbf{0.0206}}$	$\mathbf{0.9887}_{\mathbf{0.0196}}$	$\mathbf{0.9695}_{\mathbf{0.0078}}$
GHC	JumpReLU	$0.7843_{0.1923}$	$0.6709_{0.1009}$	$0.9223_{0.0976}$	$0.5328_{0.0843}$
	Dual-JumpReLU	$\mathbf{0.2544}_{\mathbf{0.0029}}$	$\mathbf{0.3212}_{\mathbf{0.0084}}$	$\mathbf{0.9865}_{\mathbf{0.0234}}$	$\mathbf{0.8420}_{\mathbf{0.0498}}$
	TopK	$0.4108_{0.1877}$	$0.4310_{0.1865}$	$0.9676_{0.0561}$	$0.4194_{0.1420}$
	Dual-TopK	$\mathbf{0.1954}_{\mathbf{0.0903}}$	$\mathbf{0.2033}_{\mathbf{0.0861}}$	$\mathbf{0.9937}_{\mathbf{0.0109}}$	$\mathbf{0.5182}_{\mathbf{0.0534}}$
	BatchTopK	$2.4065_{0.1980}$	$2.4066_{0.1890}$	$0.9386_{0.1063}$	$0.5637_{0.1236}$
	Dual-BatchTopK	$\mathbf{1.7612}_{\mathbf{0.1293}}$	$\mathbf{0.7730}_{\mathbf{0.0302}}$	$\mathbf{0.9790}_{\mathbf{0.0364}}$	$\mathbf{0.9870}_{\mathbf{0.0039}}$

8 Conclusion

We introduced DualSAE, which augments standard SAEs with a non-semantic residual branch to separate the reconstruction and interpretability objectives. Specifically, instead of forcing all information through a single sparse dictionary as standard SAEs, DualSAE uses the Sparse Branch to learn semantic information and the Residual Branch to model non-semantic residual information. Our proposed decorrelation loss effectively encourages the Residual Branch to refine the reconstruction features without redundantly aligning with active sparse decoder directions.

AI USE STATEMENT

In this work, we used generative AI tools to implement methods, specifically to write and refactor portions of our codebase. All AI-generated code was manually inspected, tested, and verified for correctness by the authors. We have not used generative AI tools to propose or refine hypotheses, to design or provide feedback on research methodology or experiments, or to interpret results. The remaining required-disclosure tasks (generating synthetic data sets, developing theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, translation, cleaning and reformatting datasets, and supporting qualitative and thematic data analysis) are not applicable to this work. Additionally, we used generative AI tools to edit the paper for grammar and readability and to format the LaTeX source. All research ideas, experimental designs, and scientific claims originate from the authors. We have reviewed all AI-assisted work and take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The code of DualSAE is anonymously available at `https://anonymous.4open.science/r/DualSAE-F4B2`.

REFERENCES

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Bart Bussmann, Patrick Leask, and Neel Nanda. Batchtopk sparse autoencoders. In *NeurIPS 2024 Workshop on Scientific Methods for Understanding Deep Learning*, 2024. URL `https://openreview.net/forum?id=d4dpOCqybL`.

Bart Bussmann, Noa Nabeshima, Adam Karvonen, and Neel Nanda. Learning multi-level features with matryoshka sparse autoencoders, 2025. URL `https://arxiv.org/abs/2503.17547`.

CodeParrot. Github code dataset. `https://huggingface.co/datasets/codeparrot/github-code`, 2022. Hugging Face dataset.

Jingyi Cui, Qi Zhang, Yifei Wang, and Yisen Wang. On the limits of sparse autoencoders: A theoretical framework and reweighted remedy. In *The Fourteenth International Conference on Learning Representations*, 2026. URL `https://openreview.net/forum?id=DSOTgzeH3w`.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL `https://openreview.net/forum?id=YicbFdNTTy`.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition, 2022. URL `https://arxiv.org/abs/2209.10652`.

Leo Gao, Tom Dupre la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=tcsZt9ZNKD`.

Leander Girrbach and Zeynep Akata. Sparse autoencoders are topic models. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*, 2026.

Aaron Gokaslan, Vanya Cohen, Ellie Pavlick, and Stefanie Tellex. Openwebtext corpus. http://Skylion007.github.io/OpenWebTextCorpus, 2019.

Robert Huben, Hoagy Cunningham, Logan Riggs Smith, Aidan Ewart, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=F76bwRSLeK.

Anton Korznikov, Andrey V. Galichin, Alexey Dontsov, Oleg Rogov, Ivan Oseledets, and Elena Tutubalina. OrtSAE: Orthogonal sparse autoencoders uncover atomic features. In *ICLR 2026 Workshop on Representational Alignment (Re^4-Align)*, 2026. URL https://openreview.net/forum?id=f0xiIBVbNk.

Akshay Kulkarni, Tsui-Wei Weng, Vivek Narayanaswamy, Shusen Liu, Wesam Sakla, and Kowshik Thopalli. Interpretable and steerable concept bottleneck sparse autoencoders. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020.

Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Ng. Efficient sparse coding algorithms. *Advances in neural information processing systems*, 19, 2006.

Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro. Online dictionary learning for sparse coding. In *Proceedings of the 26th annual international conference on machine learning*, pp. 689–696, 2009.

Alireza Makhzani and Brendan Frey. k-sparse autoencoders, 2014. URL https://arxiv.org/abs/1312.5663.

Rajiv Movva, Kenny Peng, Nikhil Garg, Jon Kleinberg, and Emma Pierson. Sparse autoencoders for hypothesis generation. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=4R0pugRyN5.

Ali Nasiri-Sarvi, Hassan Rivaz, and Mahdi S. Hosseini. SPARC: Concept-aligned sparse autoencoders for cross-model and cross-modal interpretability. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=IJfvoc2GbZ.

Andrew Ng. Sparse autoencoder. *CS294A Lecture notes*, 72(2011):1–19, 2011.

Bruno A Olshausen and David J Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, 1996.

Mateusz Pach, Shyamgopal Karthik, Quentin Bouniot, Serge Belongie, and Zeynep Akata. Sparse autoencoders learn monosemantic features in vision-language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=DaNnkQJSQf.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Tom Lieberum, Vikrant Varma, Janos Kramar, Rohin Shah, and Neel Nanda. Improving sparse decomposition of language model activations with gated sparse autoencoders. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024a. URL https://openreview.net/forum?id=zLBlin2zvW.

Senthooran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. Jumping ahead: Improving reconstruction fidelity with jumprelu sparse autoencoders, 2024b. URL https://arxiv.org/abs/2407.14435.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

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

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features, 2025. URL https://arxiv.org/abs/2502.14786.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

Minglai Yang, Xinyu Guo, Zhengliang Shi, Jinhe Bi, Steven Bethard, Mihai Surdeanu, and Liangming Pan. AlignSAE: Concept-aligned sparse autoencoders. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=I9UjKxW4nq.

APPENDIX CONTENTS

A Experimental details 14

A.1 Hyper-parameters 14

A.2 Datasets 14

A.3 Vision Encoders 15

A.4 Quantitative metrics 15

B Additional results in the vision domain 18

B.1 Ablation study on decorrelation loss 18

B.2 Detailed results on ImageNet-1K and OpenImagesV7 18

B.3 SAE as a topic model 21

B.4 SAE for hypothesis generation 24

C Evaluation on Language-Model Representations and LLM steering 27

D Limitations and Broader Impact. 37

D.1 Limitations 37

D.2 Broader impact. 37

A EXPERIMENTAL DETAILS

A.1 HYPER-PARAMETERS

Unless otherwise specified, all SAE variants are trained on frozen image embeddings for 50 epochs with batch size 2048, learning rate 10^{-4}, Adam $\epsilon = 6.25 \times 10^{-10}$, and random seeds 42, 43, and 44. We use 8192 latent dimensions for all models. For TopK-based models, we use $k = 64$. Training is performed on a CUDA device with 4 dataloader workers. We use ImageNet-1K and OpenImagesV7 as datasets, and train separate models on embeddings extracted from frozen vision encoders, including CLIP, ViT, and SigLIP2.

For the standard TopK SAE, the encoder uses TopK activation with $k = 64$, tied encoder-decoder weights, and an auxiliary TopK loss with coefficient $1/32$. The auxiliary TopK parameter is set to the same value as the main TopK parameter when not otherwise specified. The TopK SAE uses input normalization in our training configuration.

For BatchTopK SAE, we use the same number of latents and $k = 64$. The model is trained with the same optimizer settings as TopK SAE and uses gradient clipping with maximum norm 1.0. The auxiliary TopK loss coefficient is also set to $1/32$. For JumpReLU SAE, we use the same 8192 latent dimensions and optimizer settings. JumpReLU SAE uses the same sparsity coefficient and a bandwidth of 10^{-3} for the JumpReLU threshold smoothing.

DualSAE uses the same sparse latent dimensionality and the same sparse activation setting as the corresponding baseline SAE. In the main experiments, the Sparse Branch uses the selected sparsity mode, such as TopK, BatchTopK, or JumpReLU, while the Residual Branch models the residual component. The Residual Branch is parameterized as a one-layer projector with hidden width 512 unless otherwise specified. We use a decorrelation coefficient of 0.01.

Table 3: **Common training hyperparameters.** Unless otherwise specified, these settings are shared across SAE and DualSAE variants.

Hyperparameter	Value
Number of latents	8192
Training epochs	50
Batch size	2048
Learning rate	10^{-4}
Adam ϵ	6.25×10^{-10}
Random seeds	42, 43, 44
Number of workers	4
Device	1 NVIDIA H100
TopK k	64
Maximum gradient norm	1.0
Auxiliary TopK coefficient	$1/32$
Dead neuron threshold	10^7 steps
Dead neuron check interval	100 steps

A.2 DATASETS

We evaluate DualSAE on two large-scale image datasets: ImageNet-1K (Russakovsky et al., 2015) and OpenImagesV7 (Kuznetsova et al., 2020). These datasets provide complementary evaluation settings. ImageNet-1K is a single-label classification benchmark with 1,000 object categories, making it suitable for measuring class-level semantic coherence, retrieval purity, and downstream classification accuracy. OpenImagesV7 is a large-scale multi-label dataset, where each image may contain multiple object categories. This provides a more challenging setting for evaluating whether SAE and DualSAE representations preserve semantic information when images contain multiple concepts, objects, and visual contexts.

For both datasets, we first extract frozen image embeddings using pretrained vision encoders, including ViT, CLIP, and SigLIP2. The SAE and DualSAE models are then trained directly on these fixed embeddings rather than on raw pixels. This setup isolates the representation-decomposition

Table 4: **Model-specific hyperparameters.** DualSAE uses the same sparse branch configuration as the corresponding SAE baseline, with an additional Residual Branch and decorrelation loss.

Model	Hyperparameters
TopK SAE	TopK activation, $k = 64$, tied weights, normalization enabled, AuxK coefficient $1/32$.
BatchTopK SAE	BatchTopK activation, $k = 64$, AuxK coefficient $1/32$, maximum gradient norm 1.0.
JumpReLU SAE	BatchTopK activation, sparsity coefficient 10^{-4}, bandwidth 10^{-3}, maximum gradient norm 1.0.
Matryoshka SAE	BatchTopK activation, group fractions of 1/32, 1/16, 1/8, 1/4, and 17/32.
GatedSAE	Learning rate of 5e–5, and no grad-norm clip.
CB-SAE	Concept set of VLG–CBM ImageNet-filtered, retain of 1024 latents, k_{cb} of 64, and CB epochs/batch/lr: 25/2048/1e–4.
DualSAE	Same sparse mode as the corresponding SAE baseline. For TopK variants, $k = 64$. The Residual Branch uses one layer with hidden width 512. We use a decorrelation coefficient of 0.01 and warm-up steps of 2500. We used ReLU as a nonlinear operator in the residual branch.

problem from encoder training and allows us to compare how different SAE variants reconstruct and decompose the same pretrained embedding space.

On ImageNet-1K, we use the standard class labels to compute retrieval-based class purity, KNN accuracy, and linear probing accuracy. Since each image has a single ground-truth class, KNN and LP are reported as top-1 accuracy. On OpenImagesV7, we adapt the evaluation to the multi-label setting. Class purity is computed based on the maximum label support among retrieved or activated samples, and KNN/LP performance is reported using micro-F1 with a threshold of 0.5. Together, these two datasets allow us to evaluate DualSAE under both clean single-label semantics and more realistic multi-label visual scenes.

A.3 Vision Encoders

We study SAE and DualSAE decompositions on frozen image embeddings extracted from pretrained vision encoders. In our main experiments, we use ViT (Dosovitskiy et al., 2021), CLIP (Radford et al., 2021), and SigLIP2 (Tschannen et al., 2025). These encoders provide different representation spaces and allow us to test whether the proposed decomposition is robust across model families. ViT provides a standard supervised vision representation, while CLIP and SigLIP2 provide vision-language-aligned embeddings that often contain richer semantic and contextual information.

For each image I, a frozen encoder f produces an embedding $x = f(I) \in \mathbb{R}^d$. We train all SAE variants directly on these fixed embeddings rather than updating the vision encoder. This isolates the representation decomposition problem from encoder training and ensures that differences among SAE variants are due to the learned sparse and residual decompositions rather than changes in the backbone model.

A.4 Quantitative metrics

Let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ denote the evaluation set, where $x_i \in \mathbb{R}^d$ is the input embedding and y_i is its label. For single-label datasets, $y_i \in \{1, \dots, C\}$; for multi-label datasets, $y_i \in \{0,1\}^C$ is a multi-hot vector. Given an autoencoder, we obtain latent activations $z_i \in \mathbb{R}^m$ and reconstruction $\hat{x}_i \in \mathbb{R}^d$. For latent j, we define its activating set as

$$A_j = \{i \in \{1, \dots, N\} : z_{ij} > 0\},$$

and denote the set of active latents by

$$\mathcal{J}_{\text{act}} = \{j \in \{1, \dots, m\} : |A_j| > 0\}.$$

Reconstruction R^2. We measure reconstruction fidelity using the coefficient of determination,

$$R^2 = 1 - \frac{\sum_{i=1}^N \|x_i - \hat{x}_i\|_2^2}{\sum_{i=1}^N \|x_i - \bar{x}\|_2^2}, \qquad \bar{x} = \frac{1}{N}\sum_{i=1}^N x_i.$$

Higher R^2 indicates better reconstruction quality.

MonoScore (MS). MonoScore measures whether each latent consistently activates on semantically similar images. Let $\mathcal{D} = \{x_n\}_{n=1}^N$ be the evaluation set and let a_{jn} denote the activation of latent j on image embedding x_n. We use a frozen semantic encoder $E(\cdot)$ to obtain the ℓ_2-normalized image embedding

$$h_n = \frac{E(x_n)}{\|E(x_n)\|_2}, \qquad s_{nm} = h_n^\top h_m,$$

where s_{nm} is the cosine similarity between image embeddings x_n and x_m.

To make activation magnitudes comparable across latents, we independently min–max normalize the activations of each latent:

$$\widetilde{a}_{jn} = \begin{cases} \dfrac{a_{jn} - \min_{n'} a_{jn'}}{\max_{n'} a_{jn'} - \min_{n'} a_{jn'}}, & \text{if } \max_{n'} a_{jn'} > \min_{n'} a_{jn'}, \\ 0, & \text{otherwise.} \end{cases}$$

The MonoScore of latent j is the activation-weighted average semantic similarity over all unordered image pairs:

$$MS_j = \frac{\displaystyle\sum_{1 \le n < m \le N} \widetilde{a}_{jn}\widetilde{a}_{jm} h_n^\top h_m}{\displaystyle\sum_{1 \le n < m \le N} \widetilde{a}_{jn}\widetilde{a}_{jm}}.$$

Here, $\widetilde{a}_{jn}\widetilde{a}_{jm}$ is the co-activation weight of the pair (x_n, x_m), assigning greater importance to pairs that strongly activate latent j.

We evaluate only latents for which the denominator is positive:

$$\mathcal{J}_{\mathrm{MS}} = \left\{ j \;\middle|\; \sum_{1 \le n < m \le N} \widetilde{a}_{jn}\widetilde{a}_{jm} > 0 \right\}.$$

The model-level MonoScore is then the mean across these valid latents:

$$MS = \frac{1}{|\mathcal{J}_{\mathrm{MS}}|} \sum_{j \in \mathcal{J}_{\mathrm{MS}}} MS_j.$$

A larger MS_j indicates that images strongly activating latent j are more semantically similar; consequently, a larger model-level MS indicates more monosemantic active latents on average.

Binary purity. Binary purity measures whether a latent is associated with a dominant class among the samples that activate it.

For a single-label dataset, the purity of latent j is

$$\mathrm{Purity}_j = \max_{c \in \{1,\dots,C\}} \frac{1}{|A_j|} \sum_{i \in A_j} \mathbb{I}[y_i = c].$$

For a multi-label dataset with multi-hot labels $y_i \in \{0,1\}^C$, we define the class support of latent j for class c as

$$\mathrm{Supp}_j(c) = \frac{1}{|A_j|} \sum_{i \in A_j} y_{i,c},$$

and define purity as the maximum class support,

$$\mathrm{Purity}_j = \max_{c \in \{1,\dots,C\}} \mathrm{Supp}_j(c).$$

This matches the implementation in our code: purity is the largest fraction of activating samples that share any one binary label. In both cases, the reported binary purity is the mean over active latents,

$$\mathrm{Purity} = \frac{1}{|\mathcal{J}_{\mathrm{act}}|} \sum_{j \in \mathcal{J}_{\mathrm{act}}} \mathrm{Purity}_j.$$

KNN accuracy from reconstructed embeddings. To evaluate whether semantic label information is preserved after reconstruction, we build a training bank of reconstructed embeddings

$$\mathcal{B} = \{(\hat{x}_i^{\mathrm{tr}}, y_i^{\mathrm{tr}})\}_{i=1}^{N_{\mathrm{tr}}},$$

where each $\hat{x}_i^{\mathrm{tr}}$ is L2-normalized. For an evaluation example, we reconstruct and normalize its embedding,

$$q = \frac{\hat{x}}{\|\hat{x}\|_2},$$

retrieve its k nearest neighbors in the bank by cosine similarity, and denote their labels by $\{y_{n_1}, \ldots, y_{n_k}\}$.

For single-label datasets, we use majority voting:

$$\hat{y} = \arg\max_{c \in \{1,\ldots,C\}} \sum_{t=1}^{k} \mathbb{I}[y_{n_t} = c],$$

and report top-1 accuracy,

$$\mathrm{KNN} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}[\hat{y}_i = y_i].$$

For multi-label datasets, we average the neighbors' multi-hot labels,

$$s = \frac{1}{k} \sum_{t=1}^{k} y_{n_t} \in [0,1]^C, \qquad \hat{y}_c = \mathbb{I}[s_c \geq 0.5].$$

We then report micro-F1 over all sample-class pairs,

$$\mathrm{KNN} = \mathrm{MicroF1}(\hat{Y}, Y),$$

where

$$\mathrm{MicroF1} = \frac{2TP}{2TP + FP + FN}.$$

Here TP, FP, and FN are accumulated over all classes and all evaluation examples.

Linear probing accuracy from reconstructed embeddings. We also train a linear classifier on reconstructed training embeddings. Let

$$\hat{X}^{\mathrm{tr}} = \{\hat{x}_i^{\mathrm{tr}}\}_{i=1}^{N_{\mathrm{tr}}}, \qquad \hat{X}^{\mathrm{ev}} = \{\hat{x}_i^{\mathrm{ev}}\}_{i=1}^{N_{\mathrm{ev}}}.$$

For single-label datasets, we train a linear classifier

$$f(\hat{x}) = W\hat{x} + b \in \mathbb{R}^C$$

using cross-entropy loss,

$$\mathcal{L}_{\mathrm{CE}} = -\frac{1}{N_{\mathrm{tr}}} \sum_{i=1}^{N_{\mathrm{tr}}} \log \frac{\exp(f(\hat{x}_i^{\mathrm{tr}})_{y_i})}{\sum_{c=1}^{C} \exp(f(\hat{x}_i^{\mathrm{tr}})_c)}.$$

Predictions are obtained by

$$\hat{y} = \arg\max_c f(\hat{x})_c,$$

and LP is reported as top-1 accuracy,

$$\mathrm{LP} = \frac{1}{N_{\mathrm{ev}}} \sum_{i=1}^{N_{\mathrm{ev}}} \mathbb{I}[\hat{y}_i = y_i].$$

For multi-label datasets, we use a linear layer with C outputs and optimize binary cross-entropy with logits,

$$\mathcal{L}_{\mathrm{BCE}} = -\frac{1}{N_{\mathrm{tr}} C} \sum_{i=1}^{N_{\mathrm{tr}}} \sum_{c=1}^{C} \Big(y_{i,c} \log \sigma(f(\hat{x}_i^{\mathrm{tr}})_c) + (1 - y_{i,c}) \log(1 - \sigma(f(\hat{x}_i^{\mathrm{tr}})_c)) \Big),$$

where $\sigma(\cdot)$ is the sigmoid function. At evaluation time, predictions are thresholded at 0.5,

$$\hat{y}_{i,c} = \mathbb{I}[\sigma(f(\hat{x}_i^{\text{ev}})_c) \geq 0.5],$$

and LP is reported as micro-F1,

$$\text{LP} = \text{MicroF1}(\hat{Y}, Y).$$

Clarification about KNN and LP on $\hat{x}$ instead of the sparse code z. Our goal is to evaluate reconstruction quality by measuring how well the reconstructed embedding $\hat{x}$ retains the semantic information of the original embedding. The sparse code z instead indicates which monosemantic dictionary atoms are active in an input and is expressed in a different feature space. Therefore, directly using z to evaluate semantic preservation would be misleading. MonoScore and Purity are two more reasonable metrics for evaluating the sparse code z.

B Additional results in the vision domain

B.1 Ablation study on decorrelation loss

To isolate the contribution of the decorrelation loss, we keep all the settings fixed while removing $\mathcal{L}_{\text{dcor}}$. Table 5 reports the performance difference between the ablated and full models. Removing the decorrelation loss reduces every metric across all encoders, activation functions, and datasets. The degradation is particularly pronounced for MonoScore and purity, indicating that allowing the two branches to reuse the same directions weakens the monosemantic structure of the Sparse Branch. The consistent reduction in R^2, KNN accuracy, and LP accuracy further shows that the loss improves the branch decomposition without sacrificing reconstruction or downstream utility. These results suggest that $\mathcal{L}_{\text{dcor}}$ *is important for both interpretability and reconstruction in DualSAE.*

Table 5: **Ablation of the decorrelation loss.** We report the metric changes after removing $\mathcal{L}_{\text{dcor}}$. Every entry is negative, indicating that removing the decorrelation loss degrades performance. In other words, $\mathcal{L}_{\text{dcor}}$ allows the Sparse Branch to learn more interpretable representations (higher MS and Purity) and better reconstruction (higher R^2, KNN, and LP).

Encoder	Activation	ImageNet-1K					OpenImagesV7				
		ΔR^2	ΔMS	ΔPur.	ΔKNN	ΔLP	ΔR^2	ΔMS	ΔPur.	ΔKNN	ΔLP
ViT	TopK	-0.014	-0.098	-0.230	-0.029	-0.175	-0.028	-0.033	-0.042	-0.019	-0.005
	BatchTopK	-0.021	-0.072	-0.179	-0.045	-0.001	-0.027	-0.029	-0.041	-0.077	-0.004
	JumpReLU	-0.013	-0.232	-0.362	-0.004	-0.015	-0.016	-0.045	-0.040	-0.001	-0.006
CLIP	TopK	-0.022	-0.051	-0.015	-0.022	-0.001	-0.037	-0.023	-0.038	-0.004	-0.001
	BatchTopK	-0.027	-0.075	-0.007	-0.056	-0.001	-0.046	-0.052	-0.021	-0.003	-0.001
	JumpReLU	-0.025	-0.037	-0.111	-0.003	-0.003	-0.040	-0.051	-0.031	-0.001	-0.005
SigLIP2	TopK	-0.022	-0.013	-0.054	-0.012	-0.008	-0.029	-0.227	-0.002	-0.010	-0.006
	BatchTopK	-0.025	-0.009	-0.028	-0.015	-0.001	-0.060	-0.273	-0.026	-0.002	-0.009
	JumpReLU	-0.016	-0.017	-0.077	-0.001	-0.002	-0.061	-0.031	-0.039	-0.001	-0.005

B.2 Detailed results on ImageNet-1K and OpenImagesV7

Table 6 compares standard SAE variants with their DualSAE counterparts across three vision encoders, six activation mechanisms, and two datasets (with results averaged over three random seeds). The most consistent trend is the improvement in reconstruction quality. Across all encoder and activation settings, DualSAE substantially increases reconstruction R^2 on both ImageNet-1K and OpenImagesV7. This indicates that the Residual Branch effectively recovers information that is difficult to preserve through a single sparse bottleneck.

DualSAE also improves latent interpretability in most settings. On ImageNet-1K, DualSAE consistently increases MonoScore and binary purity across nearly all SAE variants, showing that the Sparse Branch learns more semantically coherent and class-consistent latent features. The improvement in purity is especially large, suggesting that active latents in DualSAE tend to correspond to more concentrated visual concepts. On OpenImagesV7, the gains in purity remain strong across

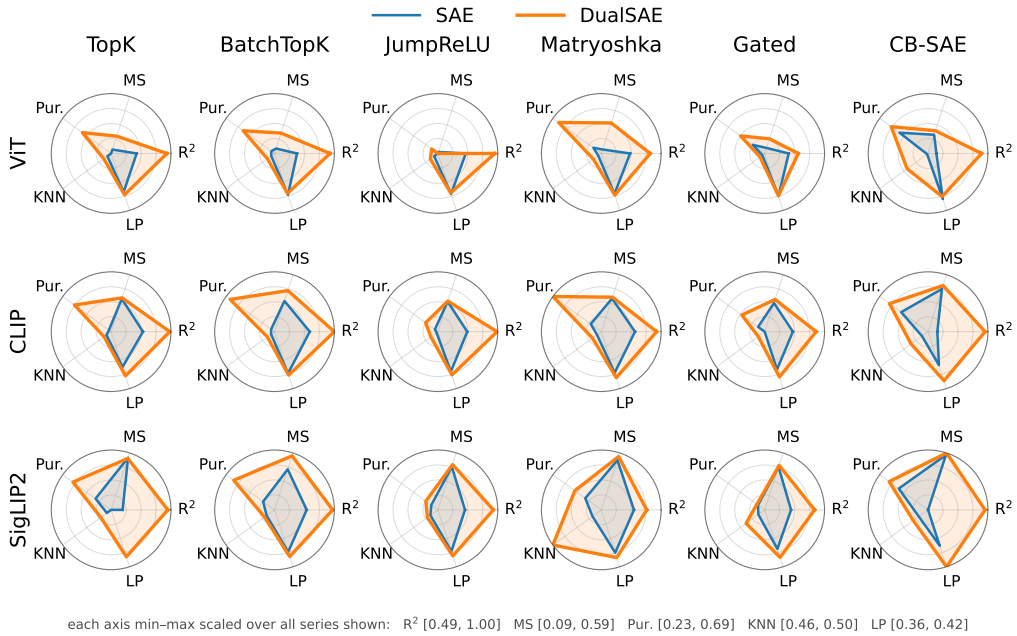

Figure 13: **Comparison of SAE and DualSAE on OpenImagesV7.** Radar plots compare each SAE variant with its DualSAE counterpart across three encoders and six SAE variants using reconstruction R^2, MonoScore, class purity, KNN accuracy, and LP accuracy. Across 18 settings, DualSAE generally improves reconstruction fidelity, monosemanticity, and downstream performance.

encoders and activation functions, even though the dataset is multi-label and therefore more semantically complex. MonoScore improvements are more mixed on OpenImagesV7, which is expected because images may contain multiple objects and overlapping concepts, making monosemanticity harder to measure with a single dominant visual concept.

Downstream performance also benefits from the DualSAE decomposition. On ImageNet-1K, DualSAE improves both KNN and LP accuracy for nearly every encoder and activation setting, indicating that the reconstructed embeddings preserve more label-relevant information. On OpenImagesV7, KNN performance also improves consistently, while LP performance is generally competitive and often slightly better. The smaller LP gains on OpenImagesV7 suggest that multi-label classification is a harder and more saturated setting, but DualSAE still improves reconstruction and latent purity without substantially harming downstream utility.

Overall, the table shows that DualSAE improves the reconstruction-interpretability trade-off across different encoders, activation mechanisms, and datasets. The consistent R^2 gains demonstrate the benefit of modeling residual information separately, while the improvements in MonoScore, purity, and KNN/LP performance suggest that this additional capacity does not simply make the representation less interpretable. Instead, separating the sparse reconstruction from the residual component allows DualSAE to preserve semantic structure while reconstructing the original embedding more faithfully.

Table 6: Comparison of SAE variants across vision encoders on ImageNet-1K and OpenImagesV7. Method abbreviations: TK = TopK, BTK = BatchTopK, JR = JumpReLU, MAT = Matryoshka SAE, GATE = Gated SAE, and CB = CB-SAE. Prefix D- denotes the corresponding DualSAE variant; CB-DUAL denotes CB-SAE augmented with the DualSAE residual branch. MS denotes MonoScore; Pur. denotes binary purity; KNN and LP denote KNN- and linear-probe-based downstream performance, respectively. For ImageNet-1K, both are reported as top-1 accuracy; for OpenImagesV7, both are reported as micro-F1.

		ImageNet-1K					OpenImagesV7				
Enc.	**Meth.**	**R2**	**MS**	**Pur.**	**KNN**	**LP**	**R2**	**MS**	**Pur.**	**KNN**	**LP**
ViT	TK	0.8081 ±0.0008	0.2375 ±0.0011	0.1271 ±0.0007	0.8203 ±0.0011	0.8230 ±0.0006	0.7037 ±0.0011	0.1283 ±0.0007	0.2259 ±0.0003	0.4675 ±0.0011	0.4022 ±0.0008
	D-TK	**0.9661** ±0.0003	**0.4874** ±0.0008	**0.5666** ±0.0002	**0.8226** ±0.0008	**0.8239** ±0.0004	**0.9611** ±0.0006	**0.2446** ±0.0008	**0.5028** ±0.0001	**0.4701** ±0.0010	**0.4049** ±0.0005
	BTK	0.7879 ±0.0007	0.2450 ±0.0009	0.1513 ±0.0004	0.8207 ±0.0005	0.8230 ±0.0005	0.6782 ±0.0012	0.1369 ±0.0008	0.2569 ±0.0004	0.4672 ±0.0013	**0.4050** ±0.0009
	D-BTK	**0.9725** ±0.0004	**0.4775** ±0.0005	**0.5481** ±0.0002	**0.8228** ±0.0003	**0.8242** ±0.0005	**0.9593** ±0.0006	**0.2708** ±0.0003	**0.5284** ±0.0002	**0.4700** ±0.0006	0.4035 ±0.0003
	JR	0.8020 ±0.0007	0.2840 ±0.0008	0.2391 ±0.0006	0.8201 ±0.0008	0.8221 ±0.0007	0.7238 ±0.0008	**0.1075** ±0.0009	0.2273 ±0.0004	0.4675 ±0.0006	**0.4041** ±0.0007
	D-JR	**0.9843** ±0.0003	**0.3576** ±0.0007	**0.4240** ±0.0002	**0.8239** ±0.0006	**0.8239** ±0.0006	**0.9701** ±0.0006	0.0940 ±0.0003	**0.2863** ±0.0003	**0.4703** ±0.0003	0.4030 ±0.0003
	MAT	0.8273 ±0.0002	0.2733 ±0.0017	0.2233 ±0.0018	0.8213 ±0.0002	0.8233 ±0.0001	0.7316 ±0.0003	0.1288 ±0.0004	0.3023 ±0.0010	0.4670 ±0.0003	0.4041 ±0.0003
	D-MAT	**0.9612** ±0.0008	**0.5201** ±0.0011	**0.6305** ±0.0016	**0.8242** ±0.0001	**0.8245** ±0.0003	**0.9025** ±0.0008	**0.3596** ±0.0016	**0.6373** ±0.0042	**0.4705** ±0.0002	**0.4052** ±0.0003
	GATE	0.8225 ±0.0007	0.2371 ±0.0005	0.1462 ±0.0005	0.8208 ±0.0005	0.8220 ±0.0008	0.6908 ±0.0014	0.1421 ±0.0016	0.3400 ±0.0028	0.4665 ±0.0002	**0.4060** ±0.0001
	D-GATE	**0.9268** ±0.0075	**0.3657** ±0.0032	**0.3044** ±0.0036	**0.8220** ±0.0005	**0.8236** ±0.0002	**0.7723** ±0.0030	**0.2214** ±0.0007	**0.4594** ±0.0003	**0.4682** ±0.0001	0.4059 ±0.0004
	CB	0.6550 ±0.0027	0.2809 ±0.0009	0.3640 ±0.0066	0.7988 ±0.0014	0.8078 ±0.0019	0.5572 ±0.0013	0.2564 ±0.0009	0.5018 ±0.0007	0.4655 ±0.0006	**0.4091** ±0.0011
	CB-DUAL	**0.9823** ±0.0024	**0.3511** ±0.0023	**0.5963** ±0.0145	**0.8133** ±0.0004	**0.8093** ±0.0003	**0.9418** ±0.0031	**0.2919** ±0.0091	**0.5820** ±0.0612	**0.4800** ±0.0003	0.4067 ±0.0002
CLIP	TK	0.8189 ±0.0010	0.4680 ±0.0007	0.1572 ±0.0003	0.8036 ±0.0007	0.8326 ±0.0005	0.7575 ±0.0007	0.3798 ±0.0005	0.2472 ±0.0003	0.4682 ±0.0007	0.3990 ±0.0007
	D-TK	**0.9964** ±0.0002	**0.5603** ±0.0007	**0.4256** ±0.0002	**0.8155** ±0.0002	**0.8450** ±0.0004	**0.9858** ±0.0006	**0.3879** ±0.0006	**0.5800** ±0.0002	**0.4699** ±0.0002	**0.4077** ±0.0005
	BTK	0.8472 ±0.0009	0.4357 ±0.0008	0.1410 ±0.0006	0.8069 ±0.0008	0.8347 ±0.0009	0.7863 ±0.0008	0.3607 ±0.0006	0.2584 ±0.0008	0.4673 ±0.0008	0.4053 ±0.0007
	D-BTK	**0.9972** ±0.0004	**0.5750** ±0.0003	**0.4287** ±0.0004	**0.8156** ±0.0004	**0.8451** ±0.0003	**0.9957** ±0.0003	**0.4510** ±0.0003	**0.6543** ±0.0006	**0.4700** ±0.0005	**0.4070** ±0.0006
	JR	0.7792 ±0.0005	0.4477 ±0.0006	0.1920 ±0.0003	0.8002 ±0.0009	0.8272 ±0.0009	0.7361 ±0.0009	0.3557 ±0.0009	0.2553 ±0.0004	0.4658 ±0.0010	0.4035 ±0.0009
	D-JR	**0.9745** ±0.0006	**0.4947** ±0.0005	**0.2642** ±0.0003	**0.8124** ±0.0006	**0.8424** ±0.0007	**0.9901** ±0.0002	**0.3599** ±0.0004	**0.3443** ±0.0004	**0.4699** ±0.0007	**0.4068** ±0.0008
	MAT	0.8351 ±0.0003	0.4889 ±0.0003	0.2763 ±0.0017	0.8064 ±0.0006	0.8359 ±0.0003	0.7734 ±0.0002	0.3825 ±0.0008	0.3276 ±0.0026	0.4679 ±0.0002	0.4047 ±0.0007
	D-MAT	**0.9098** ±0.0002	**0.5289** ±0.0045	**0.4721** ±0.0077	**0.8218** ±0.0008	**0.8488** ±0.0002	**0.9572** ±0.0008	**0.3939** ±0.0087	**0.6932** ±0.0044	**0.4712** ±0.0001	**0.4095** ±0.0002
	GATE	0.8256 ±0.0006	0.4013 ±0.0005	0.0920 ±0.0008	0.8015 ±0.0009	0.8324 ±0.0004	0.7264 ±0.0005	0.3431 ±0.0017	0.2891 ±0.0021	0.4648 ±0.0002	0.4015 ±0.0008
	D-GATE	**0.9940** ±0.0005	**0.4608** ±0.0026	**0.2995** ±0.0033	**0.8159** ±0.0002	**0.8450** ±0.0001	**0.9246** ±0.0043	**0.3741** ±0.0054	**0.4466** ±0.0049	**0.4695** ±0.0002	**0.4088** ±0.0003
	CB	0.5959 ±0.0012	0.3679 ±0.0137	0.4027 ±0.0052	0.7709 ±0.0035	0.7978 ±0.0036	0.5653 ±0.0066	0.4653 ±0.0022	0.4857 ±0.0041	0.4692 ±0.0009	0.3972 ±0.0013
	CB-DUAL	**0.9865** ±0.0003	**0.3989** ±0.0088	**0.6872** ±0.0044	**0.8641** ±0.0001	**0.8283** ±0.0003	**0.9715** ±0.0009	**0.4952** ±0.0123	**0.5981** ±0.0842	**0.4771** ±0.0004	**0.4125** ±0.0002
SigLIP2	TK	0.6457 ±0.0009	0.5903 ±0.0010	0.3412 ±0.0010	0.8311 ±0.0010	0.8663 ±0.0004	0.5846 ±0.0011	**0.5473** ±0.0011	0.3758 ±0.0009	0.4679 ±0.0009	0.3644 ±0.0008
	D-TK	**0.9944** ±0.0005	**0.6354** ±0.0008	**0.4036** ±0.0008	**0.8517** ±0.0008	**0.8830** ±0.0005	**0.9690** ±0.0008	0.5416 ±0.0005	**0.5944** ±0.0003	**0.4723** ±0.0004	**0.4104** ±0.0008
	BTK	0.8462 ±0.0008	0.4689 ±0.0009	0.2270 ±0.0009	0.8477 ±0.0009	0.8763 ±0.0008	0.7604 ±0.0012	0.4473 ±0.0013	0.3356 ±0.0005	0.4705 ±0.0008	0.4056 ±0.0006
	D-BTK	**0.9942** ±0.0003	**0.6231** ±0.0009	**0.3958** ±0.0009	**0.8516** ±0.0005	**0.8830** ±0.0004	**0.9814** ±0.0006	**0.5635** ±0.0010	**0.6191** ±0.0002	**0.4720** ±0.0005	**0.4102** ±0.0004
	JR	0.7853 ±0.0006	0.5120 ±0.0006	0.2631 ±0.0006	0.8451 ±0.0007	0.8732 ±0.0007	0.7172 ±0.0009	0.4673 ±0.0009	0.2899 ±0.0003	0.4702 ±0.0007	0.4051 ±0.0008
	D-JR	**0.9806** ±0.0002	**0.5770** ±0.0003	**0.3046** ±0.0003	**0.8504** ±0.0002	**0.8816** ±0.0005	**0.9583** ±0.0005	**0.4873** ±0.0004	**0.3421** ±0.0002	**0.4724** ±0.0005	**0.4095** ±0.0003
	MAT	0.8362 ±0.0000	0.5894 ±0.0005	0.3261 ±0.0012	0.8466 ±0.0008	0.8768 ±0.0005	0.7645 ±0.0004	0.5261 ±0.0003	0.3805 ±0.0008	0.4712 ±0.0002	0.4064 ±0.0003
	D-MAT	**0.9083** ±0.0003	**0.5911** ±0.0023	**0.4819** ±0.0043	**0.8490** ±0.0003	**0.8812** ±0.0003	**0.8731** ±0.0010	**0.5581** ±0.0019	**0.4812** ±0.0038	**0.4999** ±0.0002	**0.4112** ±0.0001
	GATE	0.8372 ±0.0004	0.5236 ±0.0016	0.1291 ±0.0024	0.8452 ±0.0007	0.8758 ±0.0006	0.7107 ±0.0015	0.4668 ±0.0023	0.2902 ±0.0022	0.4694 ±0.0002	0.4027 ±0.0009
	D-GATE	**0.8812** ±0.0052	**0.5512** ±0.0048	**0.1923** ±0.0022	**0.8512** ±0.0011	**0.8822** ±0.0009	**0.9110** ±0.0002	**0.4812** ±0.0004	**0.3011** ±0.0002	**0.4783** ±0.0002	**0.4110** ±0.0002
	CB	0.5399 ±0.0011	0.4693 ±0.0139	0.4312 ±0.0016	0.8108 ±0.0036	0.8437 ±0.0034	0.4865 ±0.0028	0.5710 ±0.0085	0.5088 ±0.0080	0.4715 ±0.0004	0.3996 ±0.0031
	CB-DUAL	**0.9348** ±0.0004	**0.5291** ±0.0102	**0.5341** ±0.0132	**0.8444** ±0.0003	**0.8791** ±0.0002	**0.9712** ±0.0009	**0.5891** ±0.0112	**0.6012** ±0.0065	**0.4755** ±0.0003	**0.4202** ±0.0005

B.3 SAE AS A TOPIC MODEL

Table 7: **Topic modeling results** on ImageNet-1K and OpenImagesV7. We compare SAE variants with different activation functions and encoders across topic counts $K \in \{50, 100, 200, 300, 500\}$. Diversity D evaluates topic non-redundancy, and intruder accuracy C_I measures whether unrelated concepts can be identified from each topic.

Method / Encoder	ImageNet-1K					OpenImagesV7				
	K=50	K=100	K=200	K=300	K=500	K=50	K=100	K=200	K=300	K=500
Diversity D										
TopK / CLIP	**+0.11**	**+0.10**	**+0.10**	**+0.11**	**+0.10**	**+0.05**	**+0.03**	**+0.02**	**+0.04**	**+0.05**
TopK / SigLIP2	−0.01	**+0.00**	**+0.03**	**+0.06**	**+0.12**	**+0.04**	**+0.03**	**+0.03**	**+0.05**	**+0.08**
TopK / ViT	**+0.10**	**+0.08**	**+0.06**	**+0.06**	**+0.03**	**+0.05**	**+0.04**	**+0.03**	**+0.02**	**+0.03**
BatchTopK / CLIP	**+0.17**	**+0.16**	**+0.16**	**+0.16**	**+0.17**	**+0.12**	**+0.09**	**+0.10**	**+0.10**	**+0.13**
BatchTopK / SigLIP2	**+0.07**	**+0.09**	**+0.11**	**+0.15**	**+0.20**	**+0.10**	**+0.08**	**+0.08**	**+0.12**	**+0.16**
BatchTopK / ViT	**+0.05**	**+0.06**	**+0.04**	**+0.04**	**+0.03**	**+0.04**	**+0.03**	**+0.04**	**+0.03**	**+0.04**
JumpReLU / CLIP	**+0.13**	**+0.12**	**+0.09**	**+0.07**	**+0.09**	**+0.12**	**+0.12**	**+0.10**	**+0.10**	**+0.11**
JumpReLU / SigLIP2	**+0.07**	**+0.06**	**+0.04**	**+0.04**	**+0.05**	**+0.15**	**+0.14**	**+0.11**	**+0.10**	**+0.11**
JumpReLU / ViT	**+0.11**	**+0.10**	**+0.08**	**+0.06**	**+0.05**	**+0.18**	**+0.17**	**+0.14**	**+0.13**	**+0.14**
Intruder C_I (%)										
TopK / CLIP	**+5.11**	**+5.22**	**+0.44**	**+2.48**	**+1.31**	**+5.11**	−2.11	**+1.28**	−1.44	−0.07
TopK / SigLIP2	**+0.89**	−0.89	−3.50	**+0.56**	**+2.27**	**+10.44**	**+5.89**	**+2.28**	**+1.26**	**+2.87**
TopK / ViT	**+9.11**	**+5.33**	**+3.50**	**+0.59**	**+2.04**	**+2.00**	**+7.44**	**+1.22**	**+1.78**	**+1.49**
BatchTopK / CLIP	**+7.78**	**+0.78**	**+0.17**	**+3.26**	**+3.29**	**+5.56**	**+4.11**	**+3.00**	**+1.41**	**+3.04**
BatchTopK / SigLIP2	−4.44	−2.00	**+0.06**	**+1.59**	**+1.93**	**+3.56**	**+3.56**	**+0.28**	**+1.78**	**+1.44**
BatchTopK / ViT	**+3.78**	**+5.89**	**+1.11**	**+0.89**	**+2.53**	**+8.22**	**+4.22**	**+1.94**	**+0.52**	**+2.51**
JumpReLU / CLIP	**+1.33**	**+3.33**	**+0.89**	**+0.04**	**+1.31**	**+0.89**	−0.78	−0.67	**+1.63**	**+0.11**
JumpReLU / SigLIP2	**+4.67**	**+6.56**	**+1.39**	**+1.63**	**+3.84**	**+2.00**	**+5.11**	**+3.22**	**+1.52**	**+2.64**
JumpReLU / ViT	−0.67	−0.11	**+1.22**	**+2.96**	**+1.78**	−3.33	−5.22	−9.06	−7.07	−5.71

Description. We assess whether DualSAE's Sparse Branch yields better *topic models* than conventional SAEs. Following the SAE-TM pipeline (Girrbach & Akata, 2026), we interpret each dictionary feature with a distribution over caption vocabulary, cluster the features into K topics, and summarize every topic by its highest-probability words. We report two complementary metrics. *Diversity* D measures topic non-redundancy (the semantic spread of the top words across topics) and *intruder accuracy* C_I measures whether an unrelated "intruder" concept injected into a topic can be correctly identified; higher values indicate a more self-consistent topic. For both metrics, higher is better. To isolate the effect of the two-branch decomposition, we compare each DualSAE only against its *activation-matched* conventional baseline (TopK $\leftrightarrow$ Dual-TopK, BatchTopK $\leftrightarrow$ Dual-BatchTopK, JumpReLU $\leftrightarrow$ Dual-JumpReLU) and report the difference Δ = DualSAE − baseline; positive entries in Table 7 thus denote improvements attributable to DualSAE.

Settings. We evaluate on two datasets, ImageNet-1K and OpenImagesV7, and three vision encoders: CLIP (ViT-L/14, DataComp), SigLIP2 (SoViT-400m/14 at 384px), and a supervised ViT-L/16 (at 224px). For each encoder we train SAEs with three sparsity mechanisms—TopK, BatchTopK, and JumpReLU—and their DualSAE counterparts, all with a dictionary of 8192 atoms and matched sparsity budget (k=64). Topics are formed for $K \in \{50, 100, 200, 300, 500\}$. Intruder scores are produced by the same LLM judge for every method, and all reported values are averaged over three random seeds; we never average across K, encoder, or activation, reporting every configuration separately.

Diversity. DualSAE improves topic diversity almost universally: ΔD is positive in nearly every one of the 90 configurations, the sole exception being TopK/SigLIP2 at K=50 on ImageNet-1K (-0.01). The gains are stable across K and both datasets, and are largest for BatchTopK/CLIP (consistently $\approx +0.16$). This indicates that decorrelating the Sparse Branch and Residual Branch removes redundant directions, yielding topics whose vocabularies overlap less.

Intruder accuracy. Intruder accuracy is predominantly improved—DualSAE topics are more often self-consistent enough for the injected concept to be spotted. Gains are especially large for TopK/ViT on ImageNet-1K ($+9.11$ at K=50) and are positive for most configurations across both datasets. The same weak spot recurs: JumpReLU/ViT on OpenImagesV7 degrades (-3 to -9), and a few small-K BatchTopK/SigLIP2 cells are mildly negative.

Qualitative results. Figure 14 qualitatively compares image–topic assignments from DualSAE and activation-matched traditional SAEs across multiple encoder and sparsification settings. A consistent pattern is that DualSAE produces more concrete topics that closely describe the dominant visual content of each image, whereas baseline SAE topics are often driven by incidental context, broad scene semantics, or abstract appearance. For example, images containing a helmet, beetle, bread, and frog are assigned by DualSAE to topics such as *Protective Gear*, *Iridescent Beetle*, *Rustic Garden Charm*, and *Amphibian Habitat*, while the corresponding baseline topics include less directly related concepts such as *Urban Vehicles*, *Verdant Leaf Expanse*, *Kitchenware Display*, and *Intricate Earthy Texture*. Similar behavior appears across CLIP and ViT encoders and different activation functions, suggesting that the improvement is not restricted to a particular SAE configuration. Overall, these examples indicate that DualSAE concentrates the sparse branch on salient, semantically coherent visual concepts, leading to topic assignments that are more directly grounded in image content.

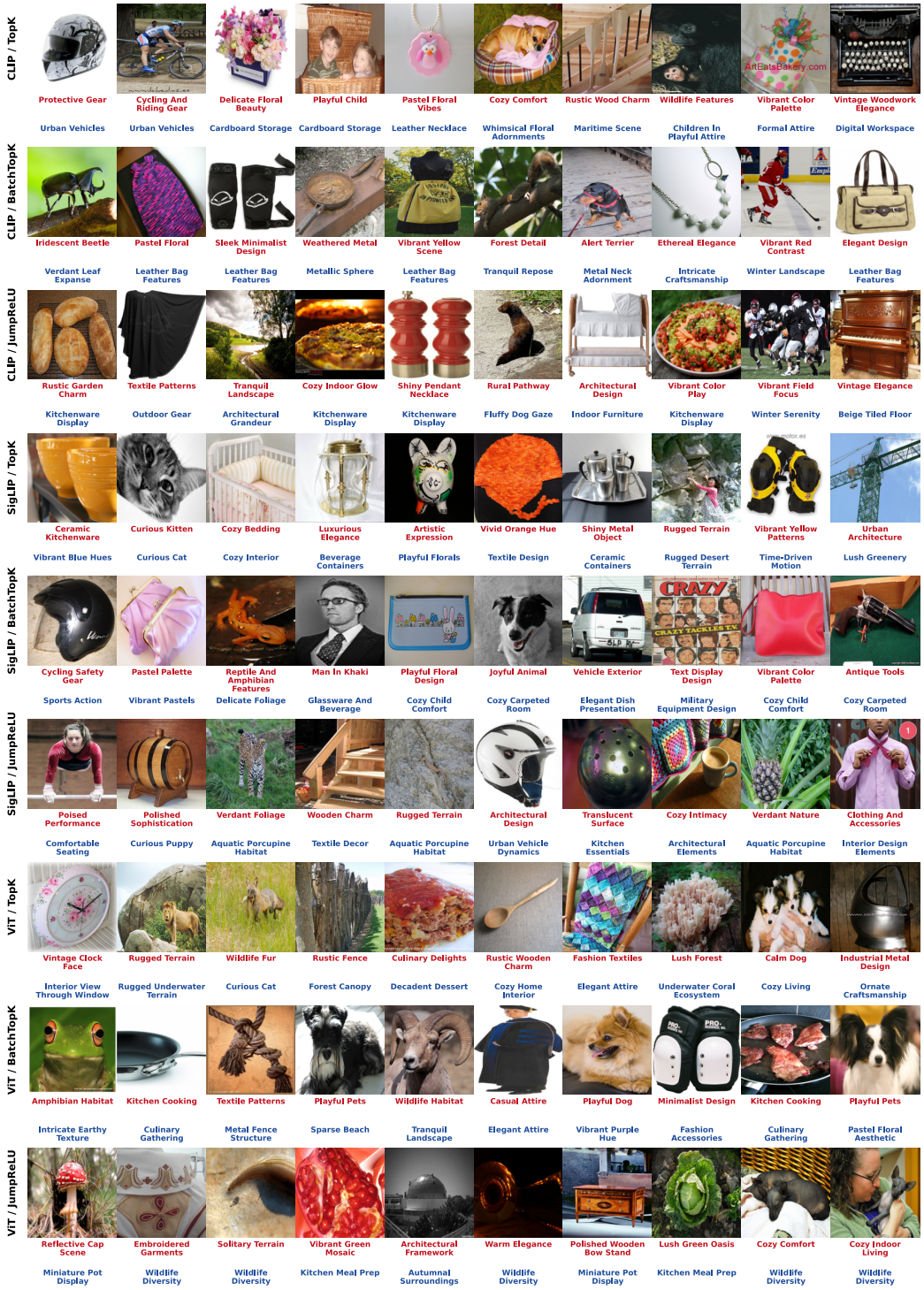

Figure 14: **Image–topic assignments between DualSAE and traditional SAEs** (one row per encoder/activation, $K=100$). Labels show each image's dominant topic under DualSAE (red) and the activation-matched baseline (blue); topic names are LLM summaries of top topic words. DualSAE yields concrete, object-specific topics, while the baseline entangles objects with context or abstract style.

B.4 SAE FOR HYPOTHESIS GENERATION

Description. We assess whether DualSAE's Sparse Branch yields better *natural-language hypotheses* than conventional SAEs. Following the HypotheSAEs pipeline (Movva et al., 2025), adapted to vision, we (i) select, for every class, the K dictionary features most discriminative of that class, (ii) ask a VLM to describe each selected feature as a short natural-language hypothesis ("why the classifier predicts this class"), and (iii) score how faithfully those hypotheses explain the class on held-out images. We report two complementary metrics. *AUROC* measures the predictive power of a hypothesis set: a VLM annotates whether each held-out image satisfies each hypothesis, and we fit a logistic model that predicts class membership from these binary annotations. *#Significant* counts how many of the K hypotheses remain significantly associated with the class after Bonferroni correction (family-wise α=0.1) — i.e., how many individually meaningful explanations the model surfaces. For both metrics, higher is better. We isolate the effect of the two-branch decomposition by comparing each DualSAE only against its *activation-matched* conventional baseline and report Δ = DualSAE − baseline; positive entries in Table 8 denote improvements attributable to DualSAE.

Settings. Datasets, encoders, and SAE configurations are identical to those used in the other experiments (ImageNet-1K and OpenImagesV7; CLIP ViT-L/14, SigLIP2, supervised ViT-L/16; dictionaries of 8192 atoms at a matched sparsity budget k=64). For each class we select K=16 latents; hypotheses are proposed by InternVL2.5-8B from each latent's 8 top-activating and 8 non-activating training images, and the same VLM produces the binary faithfulness annotations for every method. Faithfulness is evaluated one-vs-rest on 100 randomly sampled classes per dataset (for the multi-label OpenImages, positives are images containing the class, subsampled to at most 200 per class), and all values are macro-averaged over classes and averaged over three random seeds; we never average across encoder, activation, or dataset, reporting every configuration separately. Because the optimal latent-selection rule depends on the sparsity mechanism — ℓ_1-regularised selection systematically under-selects informative latents from the dense JumpReLU code, depressing absolute faithfulness for *both* models — we sweep three selection rules (LASSO, correlation, separation score) and report each configuration using its best-performing rule, always applying the same rule to DualSAE and its baseline so the comparison remains matched.

Significant hypotheses. DualSAE surfaces more significant hypotheses almost universally: Δ#Significant is positive in 17 of the 18 encoder/activation/dataset configurations, the sole exception being TopK/ViT on ImageNet-1K (-0.05, effectively parity). Gains are largest for BatchTopK/SigLIP on ImageNet-1K ($+0.52$), TopK/SigLIP on OpenImages-V7 ($+0.39$), and JumpReLU/ViT on ImageNet-1K ($+0.33$), and are consistent across both datasets. This indicates that the Sparse Branch yields features whose natural-language descriptions are individually predictive of the class, rather than diffusing class evidence across many weakly associated directions.

Predictive power. AUROC differences are small in the near-saturated TopK/BatchTopK regimes (baseline AUROC $\approx$ 0.93–0.99), where DualSAE is at parity or slightly ahead in most configurations (every deficit is $\leq$ 0.005). The clearest AUROC gains appear for JumpReLU, the hardest setting for both models: DualSAE improves AUROC in all six JumpReLU configurations, by up to $+0.024$ (JumpReLU/ViT on ImageNet-1K).

Table 8: **Hypothesis-generation results** for SAE variants across ImageNet-1K and OpenImagesV7. We report only the performance differences between the DualSAE and baseline models. ΔAUROC measures the change in predictive quality of generated hypotheses, while Δ#Significant measures the change in the average number of statistically significant hypotheses.

Encoder	Activation	ImageNet-1K		OpenImagesV7	
		ΔAUROC	Δ#Significant	ΔAUROC	Δ#Significant
CLIP	TopK	**+0.001**	**+0.11**	**+0.002**	**+0.31**
	BatchTopK	−0.003	**+0.12**	−0.001	**+0.07**
	JumpReLU	**+0.011**	**+0.31**	**+0.010**	**+0.15**
SigLIP2	TopK	**+0.001**	**+0.01**	+0.000	**+0.39**
	BatchTopK	**+0.005**	**+0.52**	−0.005	**+0.25**
	JumpReLU	**+0.010**	**+0.22**	**+0.006**	**+0.15**
ViT	TopK	−0.002	−0.05	+0.000	**+0.19**
	BatchTopK	**+0.002**	**+0.26**	**+0.005**	**+0.19**
	JumpReLU	**+0.024**	**+0.33**	**+0.012**	**+0.12**

Qualitative results. Figure 15 provides qualitative examples of class-level hypothesis generation for the *prayer rug* and *bull mastiff* classes. For *prayer rug*, the baseline TopK SAE selects only partially relevant or polysemantic latents: one focuses on generic intricate rug patterns, while another responds primarily to large mosque structures rather than the target object itself. In contrast, Dual-TopK identifies more class-specific concepts, including *red rugs with intricate patterns* and *bold, colorful rugs with intricate patterns*, whose highest-activating images consistently exhibit these visual attributes. A similar pattern appears for *bull mastiff*. The baseline JumpReLU SAE selects off-topic latents associated with unrelated scene context or other animals, such as *dogs in a car with snow and fur animals in the background* and *a black bear in a tree*. In contrast, Dual-JumpReLU recovers concepts more directly associated with the target class, such as *large, muscular dogs* and *a dog with a black nose*. Together, these examples suggest that separating the semantic sparse representation from residual information enables DualSAE to produce more coherent and class-relevant latents, rather than mixing target concepts with correlated or unrelated visual context.

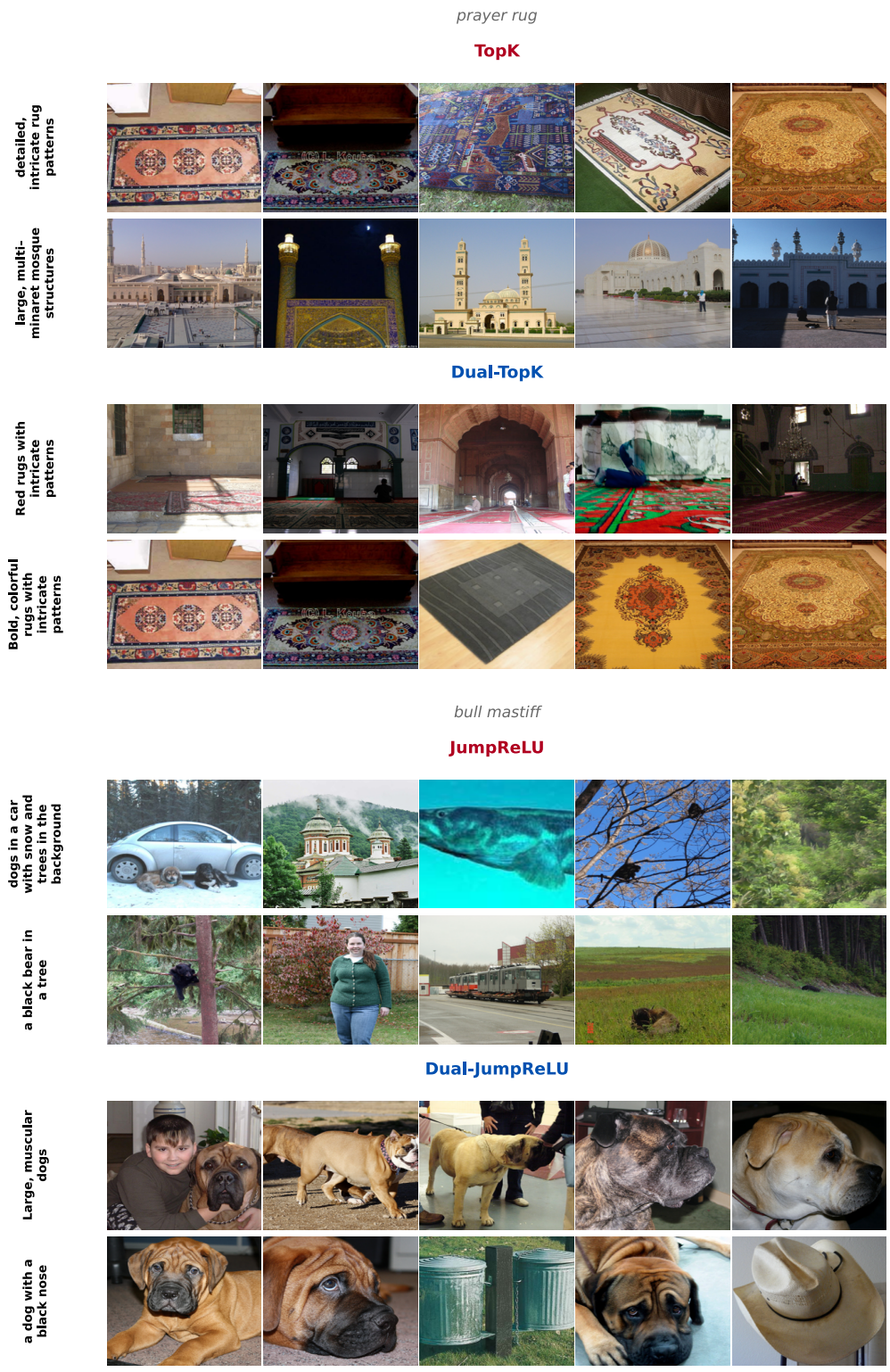

Figure 15: **Class-level hypotheses for the *prayer rug* and *bull mastiff* classes.** We show the top-2 selected latents for the baseline SAE (red) and DualSAE (blue), each accompanied by its VLM-generated hypothesis and five highest-activating images. Baseline latents are often polysemantic or off-topic, whereas DualSAE produces coherent, class-relevant concepts.

C Evaluation on Language-Model Representations and LLM steering

Language model and activation extraction. We evaluate DualSAE on residual-stream activations from `EleutherAI/pythia-410m-deduped`. The model contains 24 transformer blocks with hidden dimension $d = 1024$. We extract residual-stream activations after block 12, corresponding to a middle-layer representation. Activations are collected using forward hooks and stored once for each corpus and data split, ensuring that all SAE variants evaluated under the same setting receive identical activations.

Corpora. We consider two substantially different training corpora. **OpenWebText** represents general web text, while **GitHub code** is obtained from `codeparrot/github-code-clean` and contains mixed-language source code. This allows us to evaluate whether DualSAE generalizes across distinct residual-stream distributions. Each SAE is trained on a single corpus, and activations are never mixed across corpora.

For the GitHub corpus, we remove empty and duplicated documents. We additionally verify that the corpus is not ordered by programming language: the first shard contains 30 languages across 108,751 runs and 126,925 rows, with different languages interleaved throughout the shard.

Data preparation. Documents are packed into contexts of 128 tokens without padding. Each training stream contains exactly 64M tokens, matching the optimization budget such that each SAE observes one pass over its training stream. We construct two disjoint held-out sets of 200K tokens each from documents occurring strictly after those used for training. The first held-out split is used for model selection, while the second is reserved for final evaluation.

Activation normalization. For each corpus, we normalize activations using a single global scale computed as the root-mean-square activation norm over a 2M-token subset of the corresponding training split:

$$s = \sqrt{\mathbb{E}_x \left[\|x\|_2^2\right]}, \qquad \tilde{x} = \frac{x}{s}.$$

We use a global scale rather than per-token normalization so that variation in activation magnitude is preserved, which is important when reconstructed activations are spliced back into the language model. The resulting scale at block 12 is 125.65 for OpenWebText and 127.97 for GitHub code.

SAE configuration. All SAE variants use 8,192 latent features with target sparsity $L_0 = 64$. We train each model for 31,250 optimization steps with batch size 2,048 using Adam with learning rate 10^{-4}. Each configuration is evaluated over three random seeds. DualSAE follows our two-stage training scheme: during the first stage, the residual branch is frozen and only the sparse branch is optimized; the residual branch is subsequently enabled for joint training. All comparisons are made between models trained on the same corpus using the same block-12 activations.

Evaluation metrics. We evaluate each SAE using four metrics: ΔCE, KL divergence, R^2, and MonoScore (MS).

Cross-entropy change (ΔCE). To evaluate whether the reconstructed representation preserves the language model's behavior, we splice the SAE reconstruction $\hat{x}$ back into the residual stream at block 12 and run the remainder of the language model normally. We compute

$$\Delta\mathrm{CE} = \mathrm{CE}_{\mathrm{splice}} - \mathrm{CE}_{\mathrm{clean}},$$

where $\mathrm{CE}_{\mathrm{clean}}$ is the next-token cross-entropy of the original model and $\mathrm{CE}_{\mathrm{splice}}$ is obtained after replacing the residual-stream activation with its SAE reconstruction. A smaller $\Delta\mathrm{CE}$ indicates better preservation of the original model behavior.

KL divergence. We additionally measure the KL divergence between the clean and spliced next-token distributions:

$$D_{\mathrm{KL}}\left(p_{\mathrm{clean}} \,\|\, p_{\mathrm{splice}}\right).$$

Unlike ΔCE, which depends only on the probability assigned to the ground-truth next token, KL divergence measures changes across the full output distribution. Lower KL indicates better preservation of the model's predictive distribution.

Reconstruction quality (R^2). We measure reconstruction fidelity using pooled R^2:

$$R^2 = 1 - \frac{\sum_i \|x_i - \hat{x}_i\|_2^2}{\sum_i \|x_i - \bar{x}\|_2^2},$$

where $\bar{x}$ denotes the mean activation over the evaluation set. Higher R^2 indicates more faithful reconstruction.

Monosemanticity (MS). We evaluate the semantic coherence of the learned sparse features using MonoScore (MS). MonoScore measures the semantic consistency of examples that strongly activate each latent feature, with larger values indicating more monosemantic features.

Overall, lower ΔCE and KL indicate better behavioral preservation, while higher R^2 and MS indicate better reconstruction fidelity and monosemanticity, respectively.

More qualitative examples.

“Person Name” latents

Dual-BatchTopK latent 7597

STRONGLY ACTIVATING (5)

0.60 a show I heard on NPR last week. The reporter is Stephen Beard , ’s bureau chief and what his commentary was nothing

0.57 Sturgeon’s Law still hold true? You can connect with Richard on Google+. <|endoftext|>The Advertising Standards Authority (ASA

0.53 .com/R2suBp06fS -- Jim McElwain (@CoachMcElwain)

0.49 trade globalization. To keep investment and jobs at home, William Greider has urged that Washington "rewrite trade law,"

0.47 -- Editor’s note: When reached for comment, Michael Burton , the commercial broker for the bindery building and its

MODERATELY ACTIVATING (3)

0.28 Khouri, Undersecretary for the Ministry of Finance, Abdullah Salem Al Dhaheri , CEO of Adnoc

0.24 now." In other words-- he likes him. -- Christy Chirinos August 7

0.14 rights, his spokesman, Stephane Dujarric , said on Thursday in New York.

BatchTopK latent 3329

STRONGLY ACTIVATING (5)

0.55 yster selection and entrees at this jewel box seafood restaurant from Larry Maguire ’s team. Counter Cafe

0.45 electronic cigarettes. As a blog, I have competition, like Steve Vape, Ecig Advanced, Vapor Joes

0.42 !" For more, read: Here’s who Steve Wozniak is voting for in 2016 The

0.41 L) co-founder Steve Wozniak has a vision to create a new kind

0.40 of hard work from IndyCar helping us being the scenes, Michael Andretti too, and it wouldn’t have happened

MODERATELY ACTIVATING (3)

0.23 Franz Liszt ,the other towering virtuoso pianist

0.13 this kind of stuff," he said. Premier Brad Wall ,who took to Twitter over the weekend to ask for

0.12 Salzberg, Atlanta Gastroenterology Specialists; Pierre Desreumaux , Hopital Claude Huriez

Figure 16: **Qualitative comparison of “person name” latents learned by Dual-BatchTopK and standard BatchTopK.** Both latents respond strongly to person names. Dual-BatchTopK exhibits more consistent activation on complete person-name spans, whereas the standard BatchTopK latent also activates on unrelated or only partially relevant tokens, as highlighted in gray.

"Professional Title" latents

Dual-JumpReLU latent 4637

STRONGLY ACTIVATING (5)

0.48 At a Monday news conference, White House deputy national security adviser Ben Rhodes encouraged the would-be alliance members to ``stay

0.45 (AP) -- An investigation has found that former Portland Police Chief Larry O'Dea brought discredit to the city,

0.43 Faulconer said the mayor spoke with San Diego Police Chief Shelley Zimmerman who said the claims were unfounded

0.42 These are the trends a rational person should expect following FCC Chairman Pai's holiday gifts to the biggest ISPs.

0.38 judicial killings, forced labor, and torture,'' Acting Undersecretary for Terrorism and

MODERATELY ACTIVATING (3)

0.23 land Basin,'' said Samuel Moody, Rockhopper's CEO . Related: Christie's White House Ambitions

0.20 Interview with Sundance DiGiovanni (CEO MLG) Interview with Liquid Haypr

0.19 also noted a potential conflict of interest where PC Gamer executive editor Tyler Wilde was also roommates with an employee of

JumpReLU latent 3533

STRONGLY ACTIVATING (5)

0.46 (AP) -- An investigation has found that former Portland Police Chief Larry O'Dea brought discredit to the city,

0.45 judicial killings, forced labor, and torture,'' Acting Undersecretary

0.44 Faulconer said the mayor spoke with San Diego Police Chief Shelley Zimmerman who said the claims were unfounded

0.44 At a Monday news conference, White House deputy national security adviser Ben Rhodes encouraged the would-be alliance members to ``stay

0.39 a public statement regarding disclosure PC Gamer executive editor Tyler Wilde is in a personal relationship with a Ubiso

MODERATELY ACTIVATING (3)

0.22 water Horizon. Critics say former chief executive Lord Browne created all these disasters with relentless cost-

0.19 AND IRRESPONSIBLE! Ilford North MP Mr Streeting said the public would conclude Mr Corbyn was ``

0.12 ops'' for his miscues during the presidential debates, and Gov Jerry Brown dismissed the Texan's recruiting drive as ``

Figure 17: **Qualitative comparison of "professional title" latents learned by Dual-JumpReLU and standard JumpReLU.** Both latents respond to professional and organizational titles such as *Chief*, *CEO*, *Chairman*, and *editor*. Dual-JumpReLU activates more consistently on complete title-related spans, whereas the standard JumpReLU latent also responds to surrounding or semantically different tokens, highlighted in gray. The moderately activating examples further reveal this secondary behavior, suggesting that DualSAE yields a more semantically focused and less entangled latent representation.

"End-of-Text" latents

Dual-Gated latent 7795

STRONGLY ACTIVATING (5)

0.45 actualResult->getList()); } } <|endoftext|> package com.winterwell.depot;

0.44 } } } #endif <|endoftext|> package zorn co .reploid craft . network ;

0.44 NAGatedVC : UIViewController @end <|endoftext|> using System; using System.Collections.Generic

0.42 uint16_t Delay; #endif <|endoftext|> package info .bati11.otameshi.db

0.38 }, posts: posts }); }); <|endoftext|> /* * Copyright (c) 2008--2018 SL

MODERATELY ACTIVATING (3)

0.12 AGatedVC : UIViewController @end <|endoftext|> using System; using System.Collections.Generic;

0.12 > </body> </html> <|endoftext|> using System.Reflection; using System.Runtime.Compiler

0.12 </body> </html> <|endoftext|> --- status : publish

Gated latent 3113

STRONGLY ACTIVATING (5)

0.45 ; double y; }; #endif <|endoftext|> --- title

0.44 Ctrl.quote); module.exports = router; <|endoftext|> <?php /** * Magento * *

0.44 port=Config.PORT, host=Config.HOST) <|endoftext|> ** Form generated from reading UI file '

0.40 registerTask('default', ['uglify']); }; <|endoftext|> use strict'; angular : module('adsApp

0.40 accessor :api_key end end <|endoftext|> # Xamarin.Plugin.AudioState The

MODERATELY ACTIVATING (3) — WHERE A SECOND SENSE SHOWS UP

0.23 - 1)); } <|endoftext|> require 'basalt/packages/repo' require '

0.22 cg_sfun.h'' #endif <|endoftext|> --- layout: page-disqus title:

0.17 width: height: : --- <|endoftext|> # encoding: utf-8 require File.expand

Figure 18: **Qualitative comparison of "end-of-text" latents learned by Dual-Gated and standard Gated SAEs.** Both latents respond strongly to the <|endoftext|> token at boundaries between code fragments or files. Dual-Gated exhibits a cleaner and more consistent association with the boundary token, whereas the standard Gated latent also activates on surrounding punctuation and formatting-related tokens, highlighted in gray. The moderately activating examples further reveal this secondary behavior, suggesting that DualSAE yields a more semantically focused and less entangled boundary feature.

“class Attribute” latents

Dual-Matryoshka latent 1106

STRONGLY ACTIVATING (5)

```
0.34 /X5eoF"  target
     ="_blank"><span  class
     ="icon-right-open">
     </span> View in map
0.33 </div>  <!--  <div><a
     class ="picture-cta button
     button-lrg alert"
0.33 Codemotion</a>
     <div><span  class
     ="picture-where">Madrid
     / Spain</span>
0.30 div  class ="row">  <!--
     <div  class ="columns">
     <h1  style ="clear:
0.30 ui3-u first-col">  <div
     class ="content">
     </div>  </div
```

MODERATELY ACTIVATING (3)

```
0.15 preserve\"> <path
     fill  =\"#AAAAAA\"  d
     =\"M15.1,4.3c-2
0.15 <span  class  ='break'>
     </span>memory<span  class
     ='break'> </span>.<span
     class ='break'>
0.14 <span  class  ='break'>
     </span>widget<span  class
     ='break'> </span>.<span
     class ='break'>
```

Matryoshka latent 1893

STRONGLY ACTIVATING (5)

```
0.46 <div  class
     ="picture-caption">  <a
     class ="bottom-logo"  href
     ="/">Codemotion</
0.42 <link  href
     ="http://example.com/1"
     rel  ="alternate"  type
     ="application/atom+xml"/>
0.42 <link  href
     ="http://example.com/3"
     rel  ="alternate"  type
     ="application/rss+xml"/>
0.40 </div>  <!--  <div><a
     class ="picture-cta button
     button-lrg alert"
0.39 <link  href
     ="http://example.com/1"
     rel  ="alternate"  type
     ="application/atom+xml"
```

MODERATELY ACTIVATING (3)

```
0.14 </a></li> <li><a  href
     ="../symbols/
0.13 a></li> <li><a  href
     ="../symbols/
0.12 </a></li> <li><a  href
     ="../symbols/
```

Figure 19: **Qualitative comparison of “class attribute” latents learned by Dual-Matryoshka and standard Matryoshka SAEs.** Both latents respond to HTML attributes, but Dual-Matryoshka is more selectively aligned with the `class` attribute across diverse contexts. In contrast, the standard Matryoshka latent also activates on other attributes such as `href`, `rel`, and `type`, highlighted in gray. The moderately activating examples show the same pattern, with Dual-Matryoshka continuing to focus on `class` while the baseline shifts toward other attribute types, suggesting a cleaner and less entangled semantic feature.

Steering toward "Sadness"

Prompt. `Yesterday I found an old photograph`

Unsteered

Yesterday I found an old photograph of an elderly man. He was in a wheelchair and I was taking pictures. I found his name on the back of the photograph and when I tried to read it I found out that he was my father.

BatchTopK

Yesterday I found an old photograph of the church on the Internet and I have decided to put it up.

Here is the original picture:

In the photograph there is a large cross and a cross-shaped figure, I think it's a Roman cross.

Dual-BatchTopK

Yesterday I found an old photograph of my friend's daughter and I felt a little bit sad . I knew that my daughter was sad that her dad was taking away her and she was very upset. I felt that my heart was very sad . I wanted to cry and I ...

Figure 20: **Qualitative steering example for the "sadness" concept.** Given the same initial prompt, the unsteered generation remains largely neutral, while steering with BatchTopK SAE does not reliably induce the target concept and instead shifts the continuation toward unrelated content. In contrast, steering with Dual-BatchTopK SAE produces a continuation containing multiple semantically consistent expressions of sadness, including *sad*, *upset*, and *cry*, demonstrating stronger concept-specific control.

Steering toward “Fear”

Prompt. `The book on the table was about`

Unsteered

`The book on the table was about to be torn apart by a man who, after an initial discussion, then proceeded to take his own life. The man’s suicide was described as an act of self-destruction.`

`The book was written in the 1930s`

JumpReLU

`The book on the table was about the city of Tokyo, and it was a good book, but the book on the table was about the city of New York. I’m not so sure I’m in the same place as the book on the table.`

Dual-JumpReLU

`The book on the table was about a `` big bad '' that would be in the future, and so he was afraid to use the word ``terrorist'' so much.`

`But the last word is on the tip of my tongue. I’m worried about ...`

Figure 21: **Qualitative steering example for the “fear” concept.** Given the same prompt, the unsteered continuation develops an unrelated narrative, while JumpReLU SAE steering does not clearly induce the target emotion and instead produces a largely neutral continuation. In contrast, steering with Dual-JumpReLU SAE introduces multiple fear-related expressions, including *afraid* and *worried*, producing a continuation more consistently aligned with the intended concept.

Steering toward “Sports”

Prompt. `He picked up the pen and wrote`

Unsteered

`He picked up the pen and wrote ``I am a writer.''`

Gated [0, 0.02]

`He picked up the pen and wrote, ``A.B.''`

Dual-Gated [2, 1.00]

`He picked up the pen and wrote a few lines. The next day, the Yankees won the series , and he hit .269 with four homers and 13 RBIs in a loss in the Yankees' home opener. He was recalled to the Red Sox in the ...`

Figure 22: **Qualitative steering example for the “sports” concept.** Given the same prompt, the unsteered and GatedSAE continuations do not introduce sports-related content. In contrast, Dual-Gated SAE steering yields a fluent continuation containing multiple sport-related concepts, including the *Yankees*, *series*, *homers*, *RBIs*, and *Red Sox*, demonstrating more effective concept-specific steering while preserving generation quality.

Steering toward "Anger"

Prompt. `The bakery is next to`

Unsteered

`The bakery is next to the school, and I have a feeling it will be nice. The staff is also very nice and helpful.`

`My husband and I went to the bakery for breakfast and lunch. It was so nice. The staff was very friendly and`

Matryoshka

`The bakery is next to the mall and the mall is next to the bakery.`

Dual-Matryoshka [2, 0.94]

`The bakery is next to the bus station.`

`I'm so sick of people in a rush to get out of a terrible place and I hate that I'm in a world where I can't even make it to the airport in time.`

`The airport is ...`

Figure 23: **Qualitative steering example for the "anger" concept.** Given the same prompt, the unsteered generation remains positive and neutral, while Matryoshka-SAE steering produces a repetitive continuation without expressing the target emotion. In contrast, Dual-Matryoshka SAE steering shifts the generation toward anger-related language, including *sick*, *terrible*, and *hate*, while maintaining a coherent continuation.

D LIMITATIONS AND BROADER IMPACT

D.1 LIMITATIONS

DualSAE has two main limitations. First, although the Sparse Branch produces sparse and more interpretable latents, the residual produced by the Residual Branch is less directly interpretable and often requires indirect analyses such as residual-effect retrieval or neighborhood visualization. Second, DualSAE introduces additional training choices, including the decorrelation coefficient, Residual Branch design, and warm-up length. These hyperparameters may affect the balance between reconstruction fidelity and interpretability, and future work should develop more adaptive training schedules and more rigorous metrics for residual interpretability.

D.2 BROADER IMPACT

DualSAE is intended as a tool for interpreting foundation model representations. Modern vision and vision-language foundation models encode rich mixtures of object identity, attributes, texture, context, and dataset-specific correlations in dense embeddings. By separating dominant sparse semantic structure from residual information, DualSAE may help researchers better understand which concepts are represented cleanly, which information remains entangled, and where a model may rely on spurious or context-dependent cues. This can support model auditing, debugging, dataset analysis, and the development of more transparent foundation models.

At the same time, interpretability tools can pose risks. A better understanding of visual representations could be misused to extract sensitive attributes, probe private information, or improve models for surveillance and profiling applications. Since our work is evaluated on standard image datasets and does not introduce a deployed decision system, we do not anticipate direct harms. However, future applications of DualSAE to real-world foundation models should carefully consider privacy, bias, fairness, and surveillance risks. We encourage using DualSAE primarily as an auditing and diagnostic tool to improve transparency and reliability, rather than as a mechanism for profiling individuals or enabling harmful downstream decisions.