Opening Up a New Layer: A Deeper Look into "Interpreting CLIP with Hierarchical Sparse Autoencoders"

Anonymous authors
Paper under double-blind review

Abstract

Sparse Autoencoders (SAEs) have become essential for decomposing model activations into interpretable concepts. However, despite their effectiveness, SAEs lack a natural ordering of features, making it difficult to prioritize important concepts under compute constraints. The Matryoshka Sparse Autoencoder (MSAE) was introduced as a means of learning nested subspaces, theoretically forcing high-level features into earlier dimensions to enable adaptive granularity. In this paper, we reproduce and analyze the MSAE framework. The reproduction of the study suffers from certain challenges, but the main claims still hold. Furthermore, this study extends on the original paper by implementing feature absorption as a metric, reducing computational cost and emissions with a method inspired by the Sandwich Rule, and deeply analyzing MSAE's ability to learn hierarchical information.

1 Introduction

Interpretability remains a fundamental challenge in machine learning, as the internal representations learned by models are often difficult to understand (Carvalho et al., 2019). Yet, interpretability is essential for uncovering the reasoning behind model predictions. Without it, validating, understanding, and ultimately trusting models becomes problematic. This concern is especially pronounced in high-stakes domains such as healthcare and law, where transparent and explainable decision-making is not merely desirable but critical.

Sparse Autoencoders (SAEs) have proven to head in a promising direction for addressing the issue of polysemanticity for LLM interpretability (Shu et al., 2025). They have also emerged as a key technique for this by disentangling the dense representations of vision-language models like CLIP into interpretable, sparse features (Zaigrajew et al., 2025). However, standard SAE approaches, such as ReLU or TopK SAEs, often struggle to balance reconstruction fidelity, how well the model preserves information with sparsity, and how interpretable the features are (Gao et al., 2024). This results in a rigid Pareto frontier where improving one metric typically degrades the other (Rajamanoharan et al., 2024).

To address this limitation, Zaigrajew et al. (2025) introduce the Matryoshka Sparse Autoencoder (MSAE). Inspired by Matryoshka Representation Learning Kusupati et al. (2022), the MSAE architecture is designed to learn features at multiple granularities simultaneously. Around the same time, Bussmann et al. (2025) published a paper on MSAE independently, claiming that the hierarchal structure of this novel variant combats the issue of choosing a fixed number of latent activations for TopK SAEs. By enforcing a nested structure on the latent space, where high-priority low-dimension features are a subset of lower-priority high-dimensional ones, the model aims to prioritize semantically significant concepts while allowing for flexible inference at varying sparsity levels. The authors claim that this approach not only surpasses existing baselines in sparsity-reconstruction trade-off but also enables progressive recovery and more effective concept-based interventions. In this reproducibility study, we aim to verify these claims, focusing on the model's ability to recover hierarchical features and its utility in detecting and mitigating bias in downstream tasks.

Beyond verifying the original claims, this study introduces targeted extensions designed to evaluate the robustness and practical limitations of the MSAE framework. First, because hierarchical feature-splitting is highly vulnerable to feature absorption, we implement a feature absorption metric to rigorously test the stability of MSAE's latent space against standard SAE baselines. Second, to address the immense compute

overhead inherent to multi-granularity training, we introduce sandwich sampling, significantly mitigating computational and emission costs without sacrificing structural performance. Finally, we leverage ImageNet's lexical hierarchy to evaluate whether MSAE's nested structure truly maps onto real-world concept granularities, bridging the gap between theoretical feature ordering and downstream interpretability.

2 Scope of Reproducibility

In this study, we aim to verify the following claims made by Zaigrajew et al. (2025):

Claim 1: MSAE achieves a superior trade-off between sparsity and reconstruction quality compared to existing baselines on CLIP embeddings.

Claim 2: MSAE actively organizes features from coarse to fine granularity, enabling gradual reconstruction.

Claim 3: MSAE discovers monosemantic, human-interpretable concepts in CLIP.

Claim 4: MSAE enables faithful concept-based interventions and bias analysis.

3 Methodology

In this section, we discuss the models, datasets, and evaluation details. For replication, we use the public repository provided by the authors, which has code for pre-computing CLIP embeddings, training the SAE models, and evaluating post-training. Our adapted code can be found on GitHub (https://anonymous.4open.science/r/MSAE-sandwich-rep-1D0B/README.md).

3.1 Model Descriptions

3.1.1 CLIP

Contrastive Language Image Pre-training (CLIP) consists of a text and image encoder (Radford et al., 2021). The text encoder, which is a transformer-based model, encodes a sentence into a vector, while the image encoder, which uses a Vision Transformer, turns pixels into a vector. CLIP then uses contrastive learning to predict the correct pairings between the images and text, pushing positive pairs together and negative pairs further away in the shared embedding space. It can operate in a zero-shot setting due to its ability to classify unseen images with an understanding of individual words and the visual patterns that relate to it, allowing for a robust model.

3.1.2 Sparse Autoencoders

SAEs are designed to learn interpretable feature representations from unlabeled data, rather than relying on manually engineered features (Ng et al., 2011). A standard autoencoder compresses data into a lower-dimensional dense representation, whereas a SAE unembeds the data into a higher-dimensional vector but forces the network to activate only a small number of hidden neurons at a time, promoting sparse, interpretable feature representations. While standard autoencoders often use an undercomplete bottleneck, SAEs often use overcomplete hidden layers combined with a penalty term to enforce sparsity, resulting in better feature disentanglement and improved interpretability.

The SAE architecture leverages principles of dictionary learning. The columns of the decoder matrix can be viewed as features in a dictionary, where the hidden activations serve as sparse coefficients that decide which elements are necessary for each sample. SAEs reconstruct the input by applying a sparse weighted sum over these columns. Several architectures exist such as ReLU, TopK, and BatchTopK SAEs. ReLU SAEs (Bricken et al., 2023) introduce sparsity in the reconstruction objective by applying ℓ_1 regularization. However, this can cause activation shrinkage. TopK SAEs (Gao et al., 2024) combat this by relying solely on the k highest activations for each sample, applying a strict sparsity constraint. BatchTopK SAEs (Bussmann et al., 2024) soften this constraint by applying the TopK operation on the entire batch. This ensures that each sample has a variable number of activations, but does not eliminate the need for a fixed top k.

3.1.3 Matryoshka Sparse Autoencoder

To allow SAEs to learn natural feature hierarchies, Zaigrajew et al. (2025) combined traditional TopK SAEs with Matryoshka representation learning (Kusupati et al., 2022) by concurrently applying TopK operations with increasing values of k, h times. This simultaneous optimization across k values allows for a nested structure, where high-level features captured by lower k values form a subset of the more fine-grained features captured by higher k values. The implementation by Zaigrajew et al. (2025) sets all values of k as powers of 2 up to a dimension d, providing a balance between coverage and computational costs. For an input x, the latent feature vector z and reconstructed output $\hat{x}$ are computed at different granularity as:

$$z_i = \text{ReLU}(\text{TopK}_i(W_{enc}(x - b_{pre}) + b_{enc}))$$
$$\hat{x}_i = W_{dec} z_i + b_{pre}$$

for each k, with encoder matrix $W_{enc} \in \mathbb{R}^{n \times d}$, encoder bias $b_{enc} \in \mathbb{R}^d$, decoder matrix $W_{dec} \in \mathbb{R}^{d \times n}$, and preprocessing bias $b_{pre} \in \mathbb{R}^n$. The reconstruction objective is:

$$\mathcal{L}(x) := \sum_{i=1}^{h} \alpha_i ||x - \hat{x}_i||_2^2,$$

where h represents the number of granularities and α_i are the weighting coefficients for a given granularity. The authors propose two configurations for the values of α_i: Uniform Weighting (UW) and Reverse Weighting (RW). UW sets all α_i to 1, whereas RW gives more weight to lower k values by setting $\alpha_i = h - i + 1$. During inference, any granularity of TopK operations can be applied, and can also be omitted for the model to solely rely on ReLU activations. This flexibility removes the rigid number of active neurons of TopK SAEs, ensuring that the model can activate any number of features necessary to minimize the reconstruction error.

3.1.4 Concept Discovery

Rao et al. (2024) developed an automated neuron assignment method that leverages CLIP features encoded into a high-dimensional sparse latent space. The SAE decoder $\in \mathbb{R}^{M \times N}$ is defined as a learned dictionary $D = \{d_1, d_2, \ldots, d_M\}$, where each row vector $d_j \in \mathbb{R}^n$ represents a specific "neuron". We define a vocabulary $V = \{v_1, v_2, \ldots, v_K\}$ consisting of natural language concepts. For interpretability, each concept $v_k \in V$ is processed through the CLIP text encoder to generate the corresponding set of natural language embeddings, $E_{text} : t_k = \text{CLIPencoder}(v_k)$. For a specific neuron j, represented by dictionary vector d_j, we compute the cosine similarity between d_j and the text embedding v_k:

$$Sim(j, k) = \frac{d_j \cdot t_k}{\|d_j\| \|t_k\|}$$

The neuron is assigned to a specific concept for the j-th neuron, defined by $v_{A(j)}$, which comes from the vocabulary, yielding the maximum alignment score:

$$A(j) = \underset{k \in \{1, \ldots, K\}}{\arg\max} \; Sim(j, k)$$

3.2 Datasets

To reproduce the experiments and provide extensions of the paper, we use the following datasets: **CC12M** (Changpinyo et al., 2021), **ImageNet** (Russakovsky et al., 2014), **CelebA** (Liu et al., 2015), **Laion** (Schuhmann et al., 2021), **WordNet**(Fellbaum, 1998).

CC12M consists of roughly 12 million image-caption pairs from images and raw descriptions taken from the web. In our study, we use CC12M to extract 3 million images, which is the amount used by Zaigrajew et al. (2025) in their CC3M dataset. The images from CC12M with features extracted from CLIP ViT-L/14 is used to train all the SAE models.

ImageNet is an image database organized according to the WordNet hierarchy, consisting of over 20,000 categories with on average 1000 images for each category. **ImageNet1k** and **ImageNet100** are subsets of ImageNet with 1000 and 100 categories respectively.

CelebA is a large-scale dataset containing over 200,000 images of celebrity faces, annotated with 40 facial attributes. This dataset is used for the fourth claim where a single-layer classifier is trained on CLIP embeddings for binary gender classification. This revealed that concept magnitudes can increase the strength of concept-gender associations in a model.

Laion is a dataset with CLIP-filtered 400 million image-text pairs and their CLIP embeddings. These tokens were used to assign names to the neurons in experimentation.

WordNet is a lexical dataset that links labels from ImageNet into semantic relations and provides a hierarchical structure.

3.3 Metrics

$\mathbf{L_0}$ measures sparsity and is defined as the mean proportion of zero-valued elements in the SAE activations.

Fraction of variance unexplained (FVU) measures reconstruction fidelity by quantifying how much of the input variance is not captured by the reconstruction (Gao et al., 2024). This is done by normalizing the mean squared reconstruction error $\mathcal{L}(x)$ by the mean squared value of the mean-centered input.

Cosine Similarity (CS) measures the similarity between two vectors by calculating the cosine of the angle between them.

Linear probing (LP) is used in the downstream task to evaluate how well semantic information is preserved in the reconstructed embeddings. A linear probe is trained on ImageNet100 using CLIP embeddings, and predictions from original and reconstructed embeddings are compared using two metrics: Kullback–Leibler (KL) divergence between predicted class distributions and classification accuracy.

Centered Kernel Nearest Neighbor Alignment (CKNNA) captures how well SAE activations align with the original input embeddings in kernel space by measuring representation alignment based on mutual nearest neighbors (Huh et al., 2024).

Decoder Orthogonality (DO) assesses the degree of orthogonality among monosemantic feature directions, by computing the mean cosine similarity over the lower triangular region of the SAE decoder weight matrix.

Number of Dead Neurons (NDN) indicates unused model capacity and a failure to learn meaningful semantic features by counting neurons that remain consistently inactive across all inputs during training.

Feature Absorption (FA): Inspired by the feature absorption analysis of SAEs (Chanin et al., 2025), we implement an additional metric to compare the performance of MSAEs against SAE variants. As the number of features in an SAE increases, hierarchical features tend to split into more specific child features. However, this process is not always robust as features that appear monosemantic may fail to activate in appropriate contexts and instead become absorbed by their descendant features. Prior work reports that MSAEs exhibit the lowest feature absorption rate among evaluated models, indicating improved robustness to this failure mode (Karvonen et al., 2025). Further explanation can be found in Section 4.5.1.

4 Experiments

We used the publicly available codebase by Zaigrajew et al. (2025) to reproduce the key claims in the study.

4.1 Claim 1: Sparsity-Fidelity Trade-off

To verify **Claim 1**, we assess the performance of various SAE models using the metrics explained above. For most metrics, the SAE models were validated on the ImageNet1k validation dataset and ImageNet100 for linear probing. Following the first table in the paper, we train the models with expansion factor 8, since the

authors claim this had the best performance. We compare the following SAEs: ReLU ($\lambda = 0.03$), BatchTopK ($k = 256$), Matryoshka (both uniform (UW) and reverse weighting (RW) α coefficients), and Matryoshka with sandwich sampling. The models were trained on a subset of CC12M Clip embeddings and evaluated on the CC12M validation set and ImageNet-1k test set.

Model	$\mathbf{L_0}\uparrow$	**FVU** $\downarrow$	**CS** $\uparrow$	**LP (KL)** $\downarrow$	**LP (Acc)** $\uparrow$	**CKNNA** $\uparrow$	**DO** $\downarrow$	**NDN** $\downarrow$	**FA (%)** $\downarrow$
ReLU ($\lambda = 0.03$)	$.808_{\pm.015}$	$.287_{\pm.044}$	$.950_{\pm.011}$	$.013_{\pm.010}$	$.973_{\pm.163}$	0.657	0.0025	0	17.5
BatchTopK ($k = 256$)	$\mathbf{.891_{\pm.006}}$	$.029_{\pm.026}$	$.987_{\pm.009}$	$.008_{\pm.006}$	$.988_{\pm.107}$	0.682	0.0024	0	**10.0**
Matryoshka (UW)	$.891_{\pm.012}$	$.103_{\pm.053}$	$.964_{\pm.012}$	$.006_{\pm.018}$	$.990_{\pm.100}$	0.526	0.0025	0	16.4
Matryoshka (RW)	$.796_{\pm.012}$	$\mathbf{.007_{\pm.006}}$	$\mathbf{.997_{\pm.002}}$	$\mathbf{.002_{\pm.002}}$	$\mathbf{.994_{\pm.075}}$	**0.868**	**0.0022**	0	13.2
Matryoshka (RW) sandwich	$.762_{\pm.010}$	$.029_{\pm.011}$	$.987_{\pm.004}$	$.003_{\pm.008}$	$.993_{\pm.082}$	0.760	0.0023	1	12.0

Table 1: Quantitative comparison of SAE models. We compare the following SAEs with expansion rate 8: ReLU with sparsity regularization of 0.03, BatchTopK with 256 active neurons, Matryoshka using reverse weighting (RW) and uniform weighting (UW) α coefficients, and sandwich sampling.

Table 1 confirms that MSAE (RW) achieves a superior trade-off between sparsity and reconstruction quality compared to the other models, with the best results for FVU, CS, LP (KL), LP (Acc), CKNNA, DO, and FA.

4.2 Claim 2: Gradual Reconstruction

For **Claim 2**, the authors used the metrics listed in Section 3.3 to perform progressive reconstruction using the TopK neurons while the k value increased. The original authors validated this on ImageNet-1k, but this experiment was run on the CC12M validation set of 190464 images to see if similar results patterns as seen in the original paper emerge. Furthermore, we introduce new experiments in Section 4.5.2 to learn additional properties of progressive recovery.

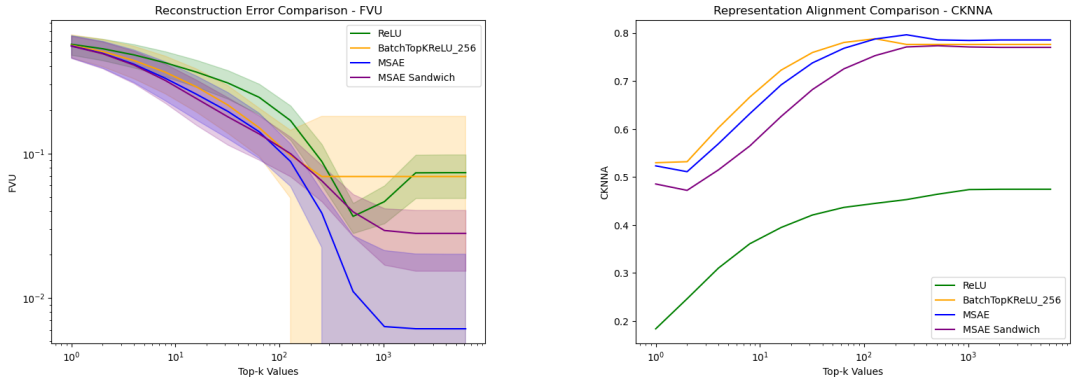
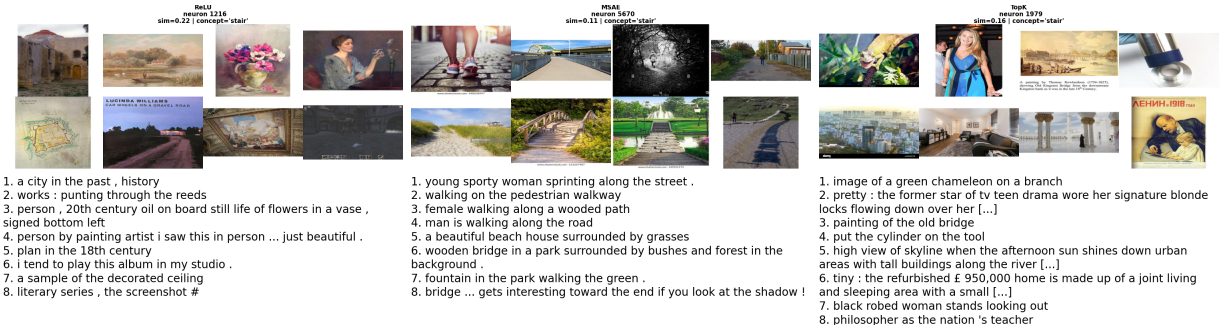


Figure 1: Reconstruction (FVU) and Representation (CKNNA) metrics during progressive recovery run on 190464 images from CC12M dataset for ReLU, BatchTopK, MSAE, and MSAE Sandwich variant.

In Figure 1, we see how different model reconstructions improve as we include more TopK neurons. For the FVU metric, the TopK model's performance stops after it reaches it's fixed TopK value of 256, as only 256 neurons can be active in the latent space. However, MSAE and ReLU performance improves as more neurons are used for reconstruction, with MSAE outperforming ReLU. This matches the behavior seen in the original paper. Additionally, the Sandwich variant of MSAE also performs better than ReLU but could not reach the performance of the base MSAE.

For representation alignment between the CLIP embeddings and SAE latent space, measured by CKNNA metric, we see the ReLU model performs worse than in the original paper. The performance of BatchTopK and MSAE models show that BatchTopK is better than MSAE at lower granularity, but MSAE eventually surpasses other models. The Sandwich variant of MSAE also performs well but not better than base MSAE and BatchTopK.



(a) ReLU model (b) MSAE model (c) Top-K model

Figure 2: Sparse version of the top-8 images and captions for the concept "stair" across different models.

A notable performance deviation emerged within representation alignment (CKNNA). This is especially highlighted with the worse performance of the standard ReLU model when compared to the results report in the original paper. A possible explanation is the shift from ImageNet-1k to the more diverse CC12M dataset, where higher variability may interact less favorably with ℓ_1-based sparsity. In contrast, BatchTopK and Matryoshka models appear more robust in preserving neighborhood structure under this setting. However, this remains a hypothesis and would require controlled ablations to confirm.

4.3 Claim 3: Monosemanticity

To evaluate **Claim 3**, a qualitative analysis of neuron behavior is performed by examining whether individual neurons consistently activate on semantically coherent images and captions. Following the neuron-x assignment introduced by Rao et al. (2024), we represent neuron activations using dictionary elements learned by the MSAE, which is the basis for the interpretation of individual neurons. We embed images and captions with CLIP, and the cosine similarity scores are computed between the embeddings and corresponding dictionary representations. For selected neurons, the top-k activating images and captions are retrieved and visualized to assess whether the neurons aligned with a single human interpretable semantic concept.

To quantify monosemanticity, a threshold is applied to identify neurons whose activations are both selective and semantically coherent. The goal of these thresholds is to filter for neurons that respond strongly to a single semantic concept as opposed to broadly activating across many. Following the paper, a similarity threshold of 0.42 is applied. Our quantitative replication reveals that Claim 3 is highly fragile and structurally fails to generalize under the baseline paper's rigid parameter choices.

Specifically, the paper proposes a ratio threshold that requires the highest scoring concept to be significantly more active than the second highest. In our setting, this threshold is not enforced, as Table 2 shows that it limits the number of selected neurons. Preliminary analysis showed that the top 10 ranked concepts are often semantically equivalent or synonyms, with cosine similarities above 0.7. As a result, applying a strict ratio test risks discarding neurons that appear monosemantic. Thus, we choose to keep the top 10 neurons with a cosine similarity above 0.7.

Model	Similarity	Best vector	Above and best	Ratio ≥ 2.0	All conditions
$\geq 0.26 \mid \geq 0.42$					
ReLU (λ=0.003)	344 | 15	1808 | 1808	317 | 15	4 | 4	0 | 0
BatchTopKReLU (k=256)	125 | 7	1489 | 1489	124 | 7	4 | 4	0 | 0
TopKReLU (k=64, RW)	157 | 7	754 | 754	154 | 7	11 | 11	0 | 0

Table 2: Number of neurons satisfying different monosemanticity criteria across models under two similarity thresholds.

As demonstrated in Table 2, applying the original similarity threshold of 0.42 produces a critically low yield, as only 7 valid neurons survive throughout the entire architecture. This indicates that the original threshold does not represent a universal boundary for interpretability but is instead coupled to the data volume and dictionary dimensions of the original work. In our experiment, we found that the scaled down threshold of 0.26 provides a similar affect.

Figure 8 shows the top-8 image-caption pairs associated with the concept *stair* across different models. In all cases, the retrieved examples exhibit strong semantic coherence, indicating that individual neurons consistently align with the target concept, suggesting a high degree of monosemanticity for *stair*. Figure 2 shows that with sparse representations, ReLU and Top K retrieve unrelated images for *stair*, while MSAE retrieves semantically related structures such as walking and bridges, which are closer to the concept. Table 2 summarizes the number of neurons satisfying each criteria. The ratio coherence is highly restrictive and no neurons satisfy both criteria simultaneously, indicating this may be unsuitable.

4.3.1 Bias Preprocessing

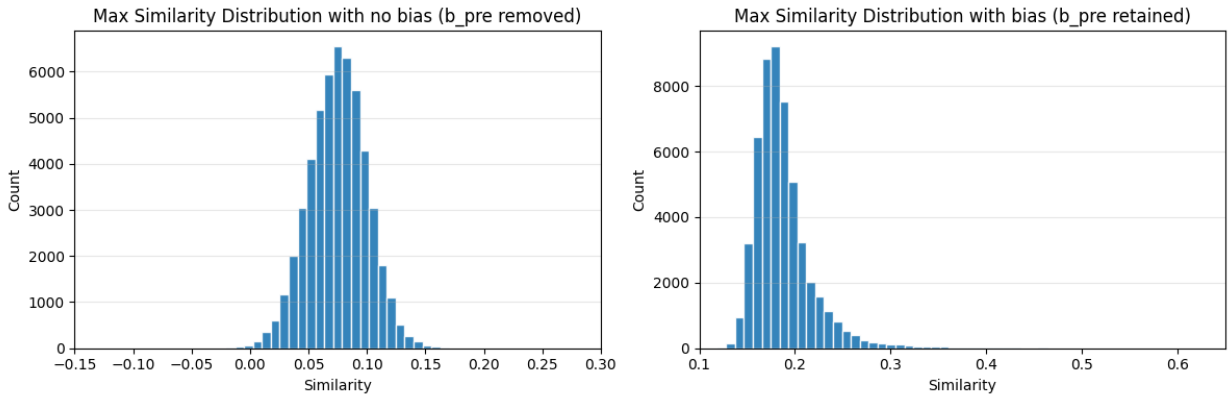


Figure 3: Effect of pre-activation bias preprocessing on neuron-concept similarity distributions.

To study the effect of pre-activation bias on monosemanticity, neuron-word similarities are evaluated using the LAION unigram and bigram datasets. The neuron activations are centered by subtracting the learned pre-activation bias prior to computing the similarities. This removes the offset that can lead to uniformly high similarity scores across many concepts, thereby obscuring neuron selectivity. The resulting similarity distributions show the impact of bias preprocessing on the emergence of monosemantic neurons.

The effects on neuron-concept similarity distributions is seen in Figure 3, based on the LAION 400m unigram dataset. When we omit bias preprocessing, cosine similarity scores cluster near 0.07, concealing the meaningful neuron-concept relations. In contrast, incorporating bias preprocessing reveals a broader similarity distribution, restoring relations that are consistent with monosemantic behavior.

4.4 Claim 4: Concept-based Intervention and Bias Analysis

To verify **Claim 4**, we show that MSAE learns causally significant features used to detect and manipulate bias in downstream tasks. Specifically, we test whether intervening on specific features can directly alter the predictions of a gender classifier trained on CLIP embeddings. We first extract embeddings of the images and attributes from CelebA using CLIP ViT-L/14. We train a linear classifier on these embeddings, achieving a F1 score of 99%, the same as the paper.

We use a MSAE (RW) to decompose CLIP embeddings into sparse latent dimensions. We identify specific latent MSAE directions corresponding to semantic concepts often associated with gender bias ("bearded", "blonde", "glasses"). We perform the intervention by clamping the activation of the target feature i to a fixed magnitude α ranging from 0 to 200. To maintain the model's sparsity constraint, if the target feature was not naturally active, we forced its inclusion by replacing the lowest-magnitude active feature. The edited

latent z_{edited} was decoded back into the CLIP embedding space $\hat{x}_{edited}$ and passed to the linear classifier. We record the shift in predicted probability for "Female" as a function of the intervention strength α.

In Figure 4, increasing the activation of the features of "bearded," "blonde," and "glasses" systematically shifts the probability of the model to classify an image as "Male". Specifically, we observe an inverse relationship with the "blonde" neuron, increasing its activation reduces the "Male" probability, confirming that the linear probe associates blonde hair with "Female". Conversely, amplifying the "glasses" and "bearded" features increases the probability of a "Male" classification, a trend that is consistent with the paper. These results replicate the findings of the original study, confirming that MSAE successfully disentangles these concepts into distinct directions that align with the decision boundary of the linear probe.

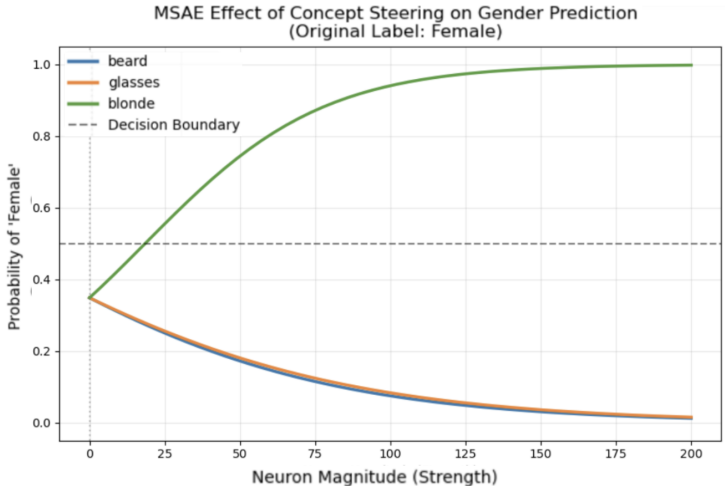


Figure 4: Increasing the concept magnitudes for "bearded", "glasses", and "blonde" in SAE space shows changes in the probabilities in gender classification in the CLIP space.

While the directional changes in classification probability support that SAE features align with meaningful semantic concepts, we further analyze intervention strength beyond the linear regime ($\alpha \to 200$). In this range, we observe a clear asymptotic saturation effect where the classifier's response plateaus. This suggests that after a certain activation magnitude, additional feature scaling does not produce proportional changes in the downstream prediction, indicating a bounded sensitivity of the linear probe in embedding space. Moreover, under the forced-inclusion protocol, we find that modifying a single sparse feature is often sufficient to induce measurable changes in prediction without destabilizing other latent dimensions. These results highlight that SAE-based interventions can reliably steer downstream decisions, although their effect size is not unbounded and exhibits diminishing returns at high activation strengths.

4.5 Extensions

4.5.1 Feature Absorption Metric

To evaluate the robustness of SAE models on feature absorption, we compare activations induced by general concepts ("dog"), and specific concepts (dog breeds). These were collected by leveraging WordNet's hierarchical structure. The general concepts are a collection of synonyms and the specific concepts are hyponyms. Both sets of concepts are encoded using CLIP ViT-L/14 to obtain text embeddings, which are passed through the SAE encoder to produce latent activations. For general concepts, we compute the mean activation of each latent feature across concepts and select the top-k features with the highest average activation. These features are treated as general candidate features. We then measure the average activation of these same features under the specific concepts. The absorption score is defined as one minus the ratio between the mean activation of the selected features for specific concepts and their mean activation for general concepts.

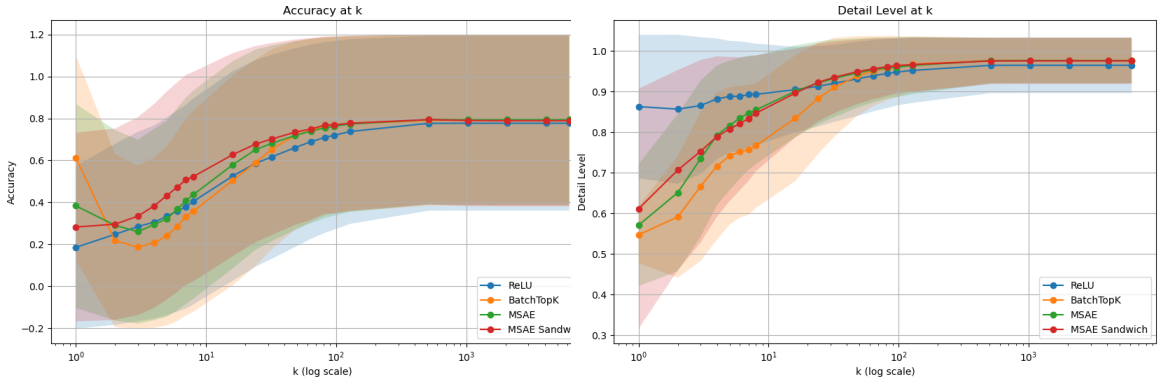
Intuitively, a higher score indicates that features strongly associated with general concepts fail to activate for more specific concepts, suggesting greater feature absorption.

4.5.2 Hierarchical Reconstruction on ImageNet

Figure 5: Image of a sea snake from node n01751748 with the hypernym path `entity -> physical_entity -> object -> whole -> living_thing -> organism -> animal -> chordate -> vertebrate -> reptile -> diapsid -> snake -> sea_snake`

To further investigate the hierarchical learning capabilities of Matryoska latent representations, we make use ImageNet's hierarchical labels. In Figure 5, we show a sample image from ImageNet along with the hierarchical labels extracted via it's hypernym path.

We measure *Accuracy* and *Level of Detail* using these classification labels. For an image to be correctly classified, the label must be present in the hypernym path. *Level of Detail* is a measure of abstractness between the range of $[0, 1]$, where 0 maps to the abstract labels (e.g. entity for Figure 5) and 1 maps to it's most specific label (e.g. sea_snake for Figure 5). This is calculated using the label's index in the hypernym path divided by the length of the path.



(a) Accuracy

(b) Level of Detail

Figure 6: Accuracy and Level of Detail of classification at different levels of reconstruction at $k = [1, 2, 3, 4, 5, 6, 7, 8, 16, 24, 32, 48, 64, 82, 96, 128, 512, 1024, 2048, 4096, 6188]$ for ImageNet-100.

Figure 6 shows different model reconstructions at the classification task. The metrics were gathered for the following k values: `[1, 2, 3, 4, 5, 6, 7, 8, 16, 24, 32, 48, 64, 82, 96, 128, 512, 1024, 2048, 4096, 6188]`. At lower k values, ReLU presents relatively higher detail level but low accuracy, suggesting the model tends to reconstruct the exact original embedding in a noisy, random manner. While BatchTopK and MSAE models have relatively higher accuracy and lower detail level, suggesting that the models tend to reconstruct the original embedding in a hierarchical manner. The low granularity reconstruction maps to an abstract concept and specific details are added as the granularity increases. However, these reconstructions are still noisy as seen with the high standard deviations.

Interestingly, BatchTopK has high accuracy at $k = 1$. As more neurons are used, the accuracy drops and reaches the same level as 1 neuron at 48 neurons. This is similar for MSAE, but only takes around 6 neurons to reach the same mean accuracy as 1 neuron. This could suggest that while specific details are added to

the embedding, these details are not priorities in terms of class-specific significant details. However, further experimentation is needed to verify this claim and investigate the true cause of this behavior.

In Appendix A.3, we provide *Accuracy* and *Level of Detail* plots for randomly selected classes from ImageNet100.

4.5.3 Sandwich sampling

To reduce the computational cost of training a MSAE, we apply a method inspired by the sandwich rule introduced by Yu and Huang (2019). The authors of this paper improved the efficiency of universally slimmable networks (US-Nets) by training the model at smallest width, largest width, and $(n-2)$ random widths at each iteration. To make this method applicable for MSAEs, we compute the reconstruction loss for a subset of granularities. The smallest granularity guarantees robust core features, the largest granularity ensures full model capacity, and a randomly sampled middle layer retains the MSAE hierarchy. The resulting loss is scaled up by the fraction of total layers to sampled layers to ensure that the gradient magnitude is consistent with the original MSAE implementation. We train the MSAE with sandwich sampling using the same hyperparameters used for the other MSAE models, only training for 20 epochs instead of 30.

Table 1 reveals that MSAE with sandwich sampling has competitive performance on most evaluation metrics while achieving a feature absorption rate of 12.0% and lower sparsity, improving upon the standard MSAE (RW) model. The model achieves these metrics at a much lower environmental cost compared to existing SAEs, requiring approximately 67% less energy during training as shown in Table 3.

4.5.4 Emission tracking

We used codecarbon (Lottick et al., 2019) to track the emissions for training different SAE models. We focus on the following metrics: emissions, cpu energy, gpu energy, ram energy, and energy consumed.

Model	**Duration (HH:MM:SS)**	**Emissions [CO_2eq] (kg)**	**Energy (kWh)**			
			CPU	**GPU**	**RAM**	**Total**
ReLU ($\lambda = 0.03$)	02:09:03	.164	.109	.464	.042	.614
BatchTopK ($k = 256$)	02:13:24	.166	.110	.466	.043	.618
MSAE (UW)	03:17:38	.199	.168	.513	.064	.744
MSAE (RW)	03:17:39	.161	.165	.373	.064	.602
MSAE (RW) sandwich	**01:43:49**	**.053**	**.081**	**.080**	**.033**	**.197**

Table 3: Emissions during training SAE models.

Table 3 shows that MSAE (UW) is the most computationally expensive, while MSAE (RW) was surprisingly efficient, consuming less total energy compared to ReLU and BatchTopK. MSAE with sandwich sampling significantly outperforms the other SAE models with the lowest CPU, GPU, and RAM energy consumption. Table 4 in Appendix A.4 shows the performance versus training emissions across all model variants. This indicates that if raw performance is priority, MSAE (RW) is the optimal choice. However, if sustainability and computational efficiency are considered, MSAE (RW) sandwich serves as compromise, slashing emissions to a fraction of the cost while losing only a minor amount of accuracy.

5 Discussion & Conclusion

Through our experiments, we prove that the main claims introduced by Zaigrajew et al. (2025) are reproducible. However, while the architectural assertions are reproducible, our replication reveals that the evidence for Claim 3 regarding monosemanticity is less conclusive, as its rigid validation constraints do not generalize well. Despite these semantic limitations, the model's core structural claims hold true, with MSAE successfully establishing a new leading Pareto frontier between reconstruction quality and sparsity. To address the shortcomings found in Claim 3, we extend the original work by introducing a new thresholding methodology that removes the strict ratio criterion, providing a more robust framework to evaluate feature validation and reconstructed hierarchically.

We extend upon the original paper's claims with our own experiments in order to investigate how we can further interpret the behavior of the model and improve it's efficiency. Firstly, we introduce the feature absorption metric to gain more insight into model monosemanticity. Moreover, we conduct further experiments to examine whether features are reconstructed hierarchically. Specifically, we investigate whether low-granularity features represent abstract concepts, and if increasing granularity gradually adds finer details. Our findings indicate that ReLU is not able to reconstruct features in a hierarchal manner, whereas MSAE and BatchTopK can. Further analysis is required to investigate the impact of hierarchical reconstruction on improved reconstruction quality. Lastly, MSAE with sandwich sampling presents competitive performance on many evaluation metrics while significantly reducing emission costs. In the future, we hope to further analyze and evaluate the sandwich method by finetuning all the hyperparameters.

References

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. 2023. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread* 2 (2023).

Bart Bussmann, Patrick Leask, and Neel Nanda. 2024. BatchTopK Sparse Autoencoders. arXiv:2412.06410 [cs.LG] `https://arxiv.org/abs/2412.06410`

Bart Bussmann, Noa Nabeshima, Adam Karvonen, and Neel Nanda. 2025. Learning Multi-Level Features with Matryoshka Sparse Autoencoders. arXiv:2503.17547 [cs.LG] `https://arxiv.org/abs/2503.17547`

Diogo V Carvalho, Eduardo M Pereira, and Jaime S Cardoso. 2019. Machine learning interpretability: A survey on methods and metrics. *Electronics* 8, 8 (2019), 832.

Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. 2021. Conceptual 12M: Pushing Web-Scale Image-Text Pre-Training To Recognize Long-Tail Visual Concepts. arXiv:2102.08981 [cs.CV] `https://arxiv.org/abs/2102.08981`

David Chanin, James Wilken-Smith, Tomáš Dulka, Hardik Bhatnagar, Satvik Golechha, and Joseph Bloom. 2025. A is for Absorption: Studying Feature Splitting and Absorption in Sparse Autoencoders. arXiv:2409.14507 [cs.CL] `https://arxiv.org/abs/2409.14507`

Christiane Fellbaum (Ed.). 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA.

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

Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. 2024. The Platonic Representation Hypothesis. arXiv:2405.07987 [cs.LG] `https://arxiv.org/abs/2405.07987`

Adam Karvonen, Can Rager, Johnny Lin, Curt Tigges, Joseph Bloom, David Chanin, Yeu-Tong Lau, Eoin Farrell, Callum McDougall, Kola Ayonrinde, Matthew Wearden, Arthur Conmy, Samuel Marks, and Neel Nanda. 2025. SAEBench: A Comprehensive Benchmark for Sparse Autoencoders in Language Model Interpretability. arXiv:2503.09532 [cs.LG] `https://arxiv.org/abs/2503.09532`

Aditya Kusupati, Gantavya Bhatt, Aniket Rege, Matthew Wallingford, Aditya Sinha, Vivek Ramanujan, William Howard-Snyder, Kaifeng Chen, Sham Kakade, Prateek Jain, and Ali Farhadi. 2022. Matryoshka Representation Learning. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 30233–30249. `https://proceedings.neurips.cc/paper_files/paper/2022/file/c32319f4868da7613d78af9993100e42-Paper-Conference.pdf`

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2015. Deep Learning Face Attributes in the Wild. arXiv:1411.7766 [cs.CV] `https://arxiv.org/abs/1411.7766`

Kadan Lottick, Silvia Susai, Sorelle A. Friedler, and Jonathan P. Wilson. 2019. Energy Usage Reports: Environmental awareness as part of algorithmic accountability. arXiv:1911.08354 [cs.LG] https://arxiv.org/abs/1911.08354

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

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. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 139)*, Marina Meila and Tong Zhang (Eds.). PMLR, 8748–8763. https://proceedings.mlr.press/v139/radford21a.html

Senthooran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. 2024. Jumping Ahead: Improving Reconstruction Fidelity with JumpReLU Sparse Autoencoders. arXiv:2407.14435 [cs.LG] https://arxiv.org/abs/2407.14435

Sukrut Rao, Sweta Mahajan, Moritz Böhle, and Bernt Schiele. 2024. Discover-then-name: Task-agnostic concept bottlenecks via automated concept discovery. In *European Conference on Computer Vision*. Springer, 444–461.

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. 2014. ImageNet Large Scale Visual Recognition Challenge. *arXiv preprint arXiv:1409.0575* (2014). https://arxiv.org/abs/1409.0575

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. 2021. LAION-400M: Open Dataset of CLIP-Filtered 400 Million Image-Text Pairs. arXiv:2111.02114 [cs.CV] https://arxiv.org/abs/2111.02114

Dong Shu, Xuansheng Wu, Haiyan Zhao, Daking Rai, Ziyu Yao, Ninghao Liu, and Mengnan Du. 2025. A Survey on Sparse Autoencoders: Interpreting the Internal Mechanisms of Large Language Models. arXiv:2503.05613 [cs.LG] https://arxiv.org/abs/2503.05613

Jiahui Yu and Thomas S. Huang. 2019. Universally Slimmable Networks and Improved Training Techniques. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Vladimir Zaigrajew, Hubert Baniecki, and Przemyslaw Biecek. 2025. Interpreting CLIP with Hierarchical Sparse Autoencoders. In *Forty-second International Conference on Machine Learning*. https://openreview.net/forum?id=5MQQsenQBm

A Appendix

A.1 Gender Bias Analysis in CelebA

Consistent with the original framework, we analyze gender bias via a CLIP-based linear classifier trained on the CelebA dataset. We validate the hypothesis that distinct latent concepts exhibit strong gender-specific associations by performing evaluations across both female (Figure 4) and male classes (Figure 7).

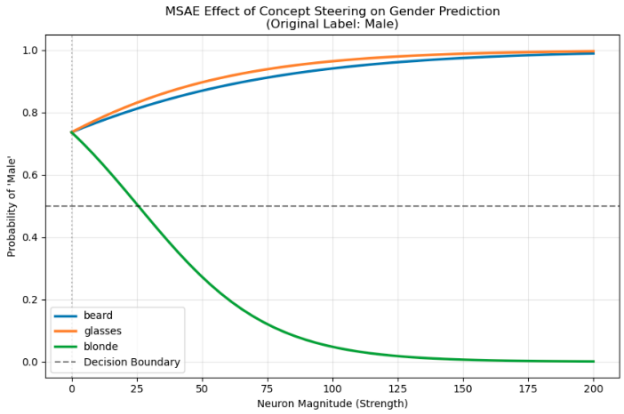
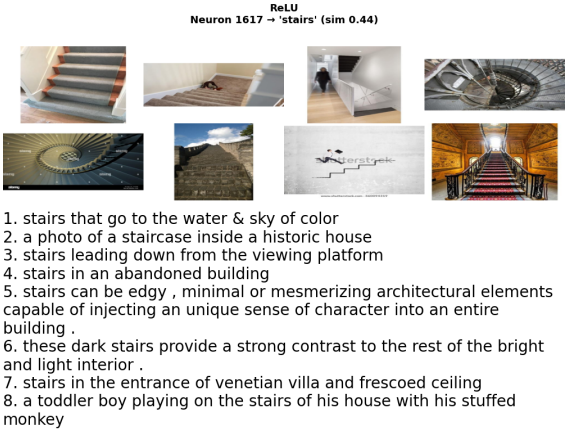
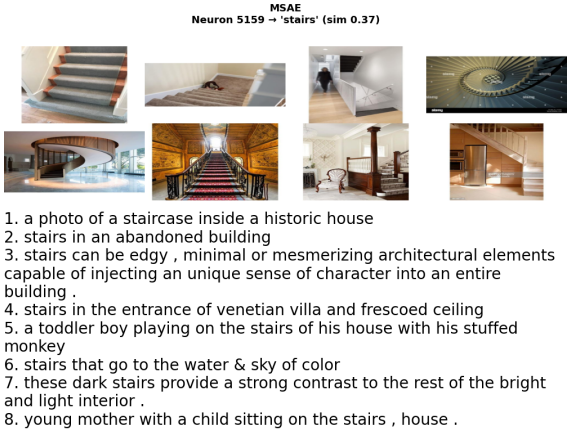


Figure 7: We showcase the male association of the words "bearded", "glasses", and "blondes". This follows the original paper as it behaves exactly opposite of the female case.

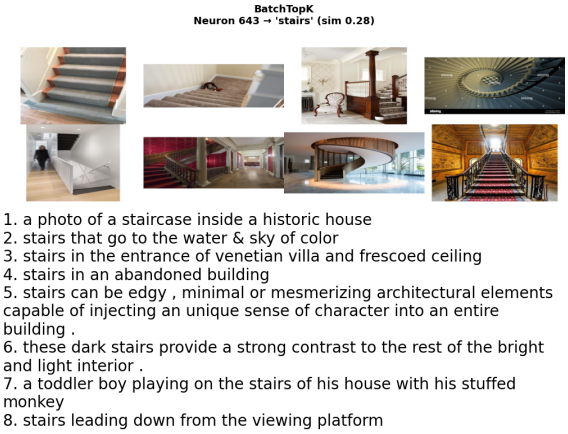
A.2 Monosemanticity



(a) ReLU model



(b) MSAE model



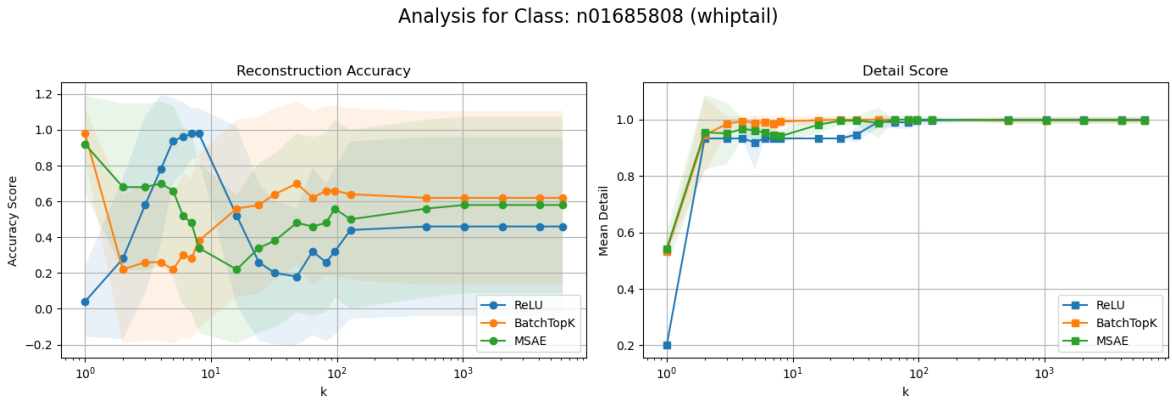
(c) Top-K model

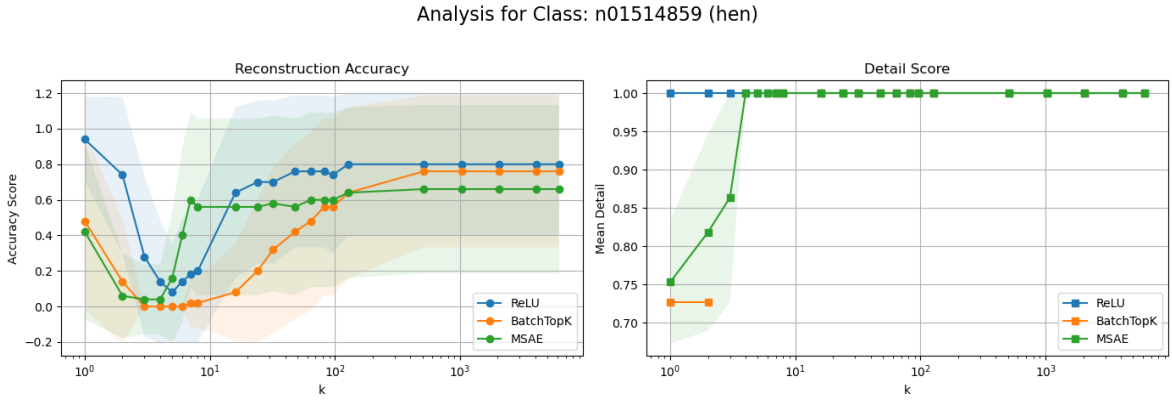
Figure 8: Top-8 images and captions for the concept "stair" across different models.

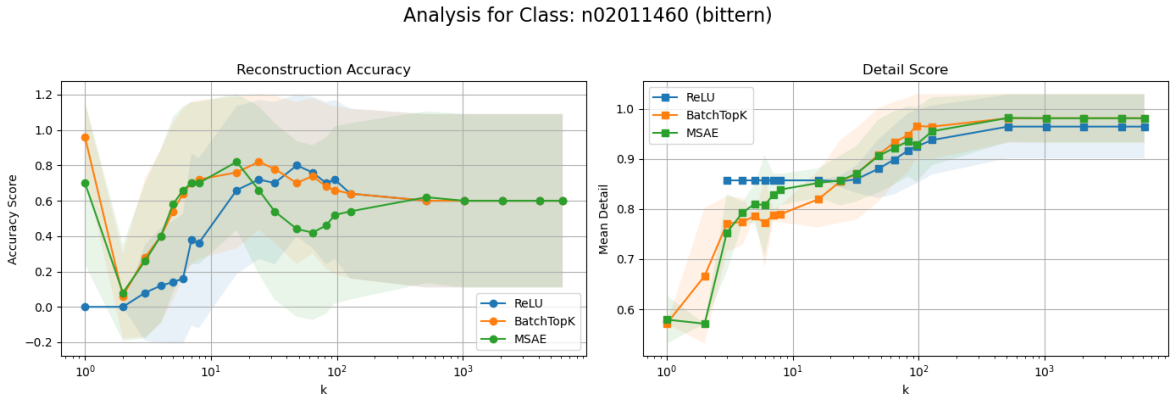
Figure 8 presents the top-8 images and captions associated with the concept "*stair*" across different models. The models show monosemanticity, confirming neurons successfully align with specific concepts

A.3 Accuracy and Level of Detail for 3 random classes in ImageNet100

Below are the Accuracy and Level of Detail plots for 3 randomly selected classes from ImageNet100. For certain classes, there were no correct classification for certain k values, leading to gaps in the Level of Detail plots.







A.4 Performance vs. Emissions

Table 4 provides a comprehensive comparison of the evaluated SAE variants across reconstruction quality, downstream performance, and environmental impact. MSAE (RW) secures both the lowest fraction of variance unexplained (FVU $= .007 \pm .006$) and the highest neighborhood accuracy (CKNNA $= 0.868$). Notably, the MSAE (RW) sandwich variant offers an exceptional Pareto-optimal trade-off for resource-constrained deployment. It cuts training carbon emissions by approximately 67% down to a mere 0.053 kg of CO_2eq, while still maintaining highly competitive performance that handily outpaces the baseline configurations.

Model	**FVU** $\downarrow$	**CKNNA** $\uparrow$	**Emissions [CO_2eq] (kg)** $\downarrow$
ReLU ($\lambda = 0.03$)	$.287_{\pm.044}$	0.657	.164
BatchTopK ($k = 256$)	$.029_{\pm.026}$	0.682	.166
MSAE (UW)	$.103_{\pm.053}$	0.526	.199
MSAE (RW)	$\mathbf{.007_{\pm.006}}$	**0.868**	.161
MSAE (RW) sandwich	$.029_{\pm.011}$	0.760	**.053**

Table 4: Performance vs. training emissions for all evaluated SAE model variants.