Evaluating the Interpretability of Sparse Autoencoders with Concept Annotations

Jonas Klotz[1,2], Cassio F. Dantas[3,5], Pallavi Jain[4,5], Diego Marcos[4,5], and Begüm Demir[1,2]

[1] Berlin Institute for the Foundations of Learning and Data (BIFOLD)
[2] Technische Universität Berlin, Germany {j.klotz,demir}@tu-berlin.de
[3] INRAE cassio.fraga-dantas@inrae.fr
[4] Inria, EVERGREEN {pallavi.jain, diego.marcos}@inria.fr
[5] UMR TETIS, Univ Montpellier, France

Abstract. Sparse autoencoders (SAEs) are increasingly used to extract interpretable concepts from vision and vision language models, yet existing evaluation methods largely rely on proxy metrics or qualitative inspection rather than measuring semantic correspondence. We present a human-grounded evaluation framework that quantifies alignment between SAE latents and human-annotated concepts, without requiring user studies, and validate this matching through targeted attribute perturbations. To enable this intervention-style evaluation in vision, we construct synCUB and synCOCO, synthetic benchmarks of paired images that differ in exactly one attribute. We introduce Fully-Binary Matching Pursuit (FBMP), a coalition-based matching procedure that supports many-to-one mappings between SAE latents and annotated concepts, and consistently outperforms one-to-one baselines. For functional validation, we propose a Targeted Attribute Perturbation Alignment Score (TAPAScore), which tests whether matched concepts respond selectively and in the expected direction under targeted image-level attribute perturbations. Under sanity checks, our matching and TAPAScore are the only evaluated metrics that reliably distinguish trained SAEs from untrained ones. Across SAEs trained on CLIP and DINOv2 embeddings, we find that increased overcompleteness can reduce perturbation alignment, indicating a reduction in interpretability. Our evaluation framework suggests that moderate dictionary sizes provide the best trade-off, yielding the most interpretable SAEs. Code and datasets are available at https://github.com/JonasKlotz/sae-concept-eval.

Keywords: Sparse Autoencoders · Interpretability

1 Introduction

Large vision and vision-language models such as DINOv2 [35] and CLIP [39] rely on high-dimensional internal representations whose structure is difficult to interpret. Understanding what individual latent dimensions represent is therefore critical for trust, controllability, and scientific analysis of these models [44].

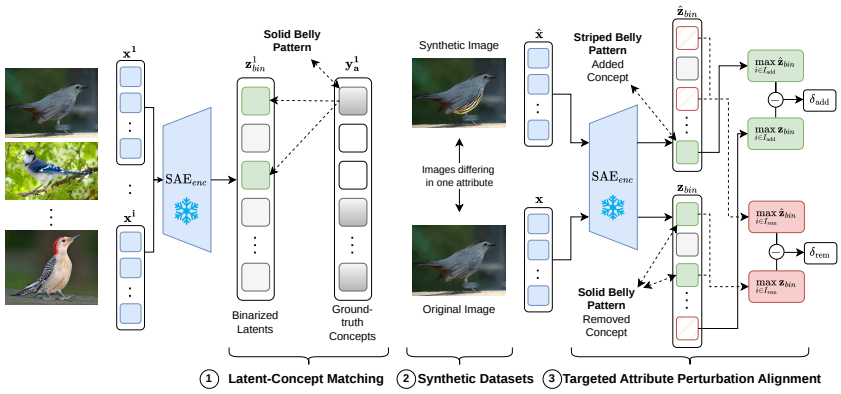

Fig. 1: Human-grounded evaluation of SAE interpretability. The framework comprises three components: **(1)** latent-concept matching, where binarized SAE latents are aligned to ground truth attribute vectors; **(2)** synthetic datasets, where paired images are constructed to differ in exactly one concept (e.g., belly pattern changed from solid to striped); and **(3)** targeted attribute perturbation alignment, where the induced latent change is measured via the signed responses δ_{add} and δ_{rem}. TAPAScore combines these components to quantify the perturbation alignment.

One promising approach is the use of sparse autoencoders (SAEs) to inspect their internal representations, a technique first developed for large language models [5, 21]. By enforcing sparsity, SAEs decompose high-dimensional activations into more localized and disentangled components that can be aligned with human-interpretable concepts. Following their success in language-model interpretability, SAEs are being applied to vision encoders to identify sparse latents associated with object parts, textures, and attributes [3, 13, 28, 41].

However, SAEs are known to exhibit systematic failure modes, such as feature splitting, where a single concept is fragmented across multiple latents [5]; feature absorption, where a general feature develops blind spots covered by specialized latents [10]; and feature composition, where co-occurring concepts merge into a single latent [53]. SAE latents are furthermore often organized hierarchically, with both fine-grained and higher-level abstractions distributed across latents [8]. Unless the SAE captures inductive biases that reflect the semantic structure present in the data, there is no guarantee that learned features will align with the concepts that human observers identify as meaningful [14]. Consequently, we claim that the alignment between SAE latents and human-understandable concepts cannot be assumed from architectural design or reconstruction quality alone, but must be evaluated explicitly.

Evaluation metrics for SAEs were primarily developed in the language domain and are commonly categorized into structural and functional [46]. Structural metrics assess representational properties of the SAE itself, such as sparsity or reconstruction fidelity [16]. Functional metrics validate whether learned features correspond to meaningful semantic concepts through behavioral tests such as feature steering, causal interventions, and editing experiments [20, 32, 42, 45]. Whereas structural metrics transfer naturally from the language to the vision domain, functional evaluation often requires controlled interventions that isolate a single semantic change. This condition is particularly difficult to satisfy in vision, where attribute changes in images rarely occur in isolation. Thus,

SAE evaluation for vision models remains dominated by qualitative examples, and structural proxies, which measure representational organization rather than whether features correspond to semantic concepts.

To address this limitation, we operationalize interpretability as the degree to which learned SAE latents align with human-understandable concepts present in the data, as in prior work on representation interpretability [2, 23]. Human evaluation is considered the gold standard of interpretability assessment [11], and human-annotated concepts/attributes provide a practical measure of this standard, as they precisely reflect the factors that human observers consistently identify as meaningful. Figure 1 summarizes our proposed evaluation framework. We first quantify semantic correspondence by matching binarized SAE activations to binary ground truth attribute vectors. We then construct controlled image pairs that differ in exactly one attribute, enabling intervention-style tests where the target change is precisely known. Finally, we introduce TAPAScore, which measures whether the latents previously matched to the perturbed attribute respond in the expected direction. In summary, **our contributions are**:

- We propose matching metrics between SAE latents and human-annotated attributes, including a coalition-based formulation, fully-binary matching pursuit (FBMP), that supports many-to-one mappings. We find that existing metrics fail basic sanity checks that our proposed metrics pass (Fig. 4) and that FBMP consistently outperforms one-to-one matchings (Figs. 5, 6, *top*).
- We construct two synthetic datasets, synCUB and synCOCO, consisting of paired images that differ in exactly one attribute or object label, enabling intervention-style evaluation of SAEs in vision.
- We propose a targeted attribute perturbation alignment score (TAPAScore) that tests whether matched latent sets respond selectively and in the right direction under targeted attribute perturbations. We show that increased overcompleteness can degrade perturbation alignment (Figs. 5, 6, *bottom*), and that matching score and TAPAScore are positively correlated (Fig. 7).

Beyond the framework itself, our evaluation yields consistent empirical guidance for practitioners: increasing overcompleteness improves statistical matching but tends to degrade perturbation alignment; moderate dictionary sizes offer the best trade-off. Based on these findings, we recommend FBMP $F_{0.5}$ matching paired with TAPAScore as the default evaluation protocol.

2 Background and Related Work

Sparse Autoencoders (SAEs) [5, 21] are a method to solve sparse dictionary learning, where signals are represented as sparse linear combinations of basis elements (atoms) from an overcomplete dictionary [43, 48]. More recently, SAEs have been used to identify interpretable concepts in neural network activations. This is motivated by evidence that individual neurons are frequently polysemantic and encode multiple unrelated concepts, suggesting that deep networks store

information in superposed representations [12]. SAEs disentangle these representations by learning a sparse and overcomplete representation in which the number of learned features exceeds the dimensionality of the activation space. SAEs implement dictionary learning through a neural network consisting of an encoder $\mathbf{W}_{\text{enc}} \in \mathbb{R}^{D \times L}$, decoder $\mathbf{W}_{\text{dec}} \in \mathbb{R}^{L \times D}$, and shared bias $\mathbf{b} \in \mathbb{R}^D$. Given an embedding $\mathbf{x} \in \mathbb{R}^D$ from a pretrained model, the SAE computes sparse activations $\mathbf{z} \in \mathbb{R}^L$ and the corresponding reconstruction $\hat{\mathbf{x}}$ as:

$$\mathbf{z} = \sigma\big(\mathbf{W}_{\text{enc}}^\top(\mathbf{x} - \mathbf{b})\big), \qquad \hat{\mathbf{x}} = \mathbf{W}_{\text{dec}}^\top \mathbf{z} + \mathbf{b}. \tag{1}$$

The model parameters are learned by minimizing an objective $\mathcal{L}(\mathbf{x}) = \mathcal{R}(\mathbf{x}) + \lambda\mathcal{S}(\mathbf{x})$ with a reconstruction and a sparsity-inducing term. SAE architectures primarily differ in how sparsity is enforced. Vanilla SAEs [5] use ReLU activation with L^1 penalty and L^2 reconstruction. TopK SAEs [16] and BatchTopK SAEs [7] replace soft sparsity penalties with hard constraints, retaining only the K largest activations per sample or batch. JumpReLU SAEs [40] instead learn a per-latent activation threshold under a direct L^0 penalty. Matryoshka SAEs [8] learn nested dictionaries with grouped activations.

Evaluation Metrics in Language-Model Interpretability. Evaluation metrics for SAEs are commonly categorized into *structural* and *functional* metrics [46]. Structural metrics assess whether the learned representation of the SAE preserves properties of the original embedding space, including reconstruction error, recovery of known features, and ablation-based diagnostics [16,30,47]. The centered kernel nearest neighbor alignment (CKNNA) metric [55] quantifies how well the neighborhood structure of the original embedding space is preserved after transformation. While essential for diagnosing training behavior, these measures do not directly evaluate whether learned features correspond to meaningful semantic concepts. Functional metrics assess whether SAE representations are semantically interpretable and practically useful. Feature Monosemanticity Score (FMS) [18] trains a predictor to relate latents to semantic attributes, while automated interpretability approaches [38] score features by prompting an LLM to generate and test natural language explanations. A complementary direction focuses on intervention-based evaluation, testing whether manipulating identified features produces the intended semantic effect [4, 32, 45], a paradigm widely applied in mechanistic analysis [20] and activation steering [17, 42]. However, applying this paradigm to vision is non-trivial, as it often requires counterfactual data that isolates the influence of individual visual attributes.

SAEs for Vision Models. While SAEs were initially developed and widely applied in natural language processing [10, 46], recent work has transferred this methodology to vision models [3, 13, 28]. In this setting, SAEs aim to recover sparse visual features corresponding to object parts, textures, attributes, or other interpretable visual concepts [15]. This enables applications such as concept discovery, representation probing, and interpretability analysis [34, 41, 51, 55]. Empirical studies further indicate that the internal representations of the CLIP vision encoder organize along interpretable concept directions, which can be uncovered through sparse feature learning [3, 41, 51].

Evaluating vision SAEs, however, presents unique challenges: while structural metrics transfer naturally across domains, functional evaluation must be adapted to the demands of visual semantics. Pach et al. [36] introduce the MonoSemanticity (MS) score, which measures how consistently a latent is activated by semantically similar inputs. The metric computes the activation-weighted similarity between images that strongly activate a given latent, with higher similarity indicating that the latent represents a more focused concept. Complementary to these, steerability metrics quantify how strongly manipulating a feature alters model output distributions and concept coverage [22].

An approach towards ground-truth evaluation uses synthetic benchmarks with known generative factors. Fel et al. [14] propose a Soft Identifiability Benchmark where images are constructed by collaging distinct objects, evaluating SAE features by whether each object class has a corresponding latent that activates when present. However, the synthetic scenes lack visual richness, which may limit the evaluation of larger SAEs. The SUB [1] dataset offers more realistic concept-level evaluation via CUB-derived bird images with controlled concept substitutions, but is designed for concept bottleneck models rather than SAEs, leaving a gap for ground-truth evaluation on complex data.

3 Human-Grounded Evaluation of SAE Concepts

Evaluating whether SAE features correspond to meaningful concepts requires a concrete operationalization of interpretability. Following prior work [2, 23], we define interpretability as the degree to which SAE features align with human-understandable concepts, and use human-annotated concepts as a scalable proxy for human judgment. This forms the basis of two complementary evaluation components: latent-concept matching, which quantifies statistical alignment between SAE latents and annotated concepts, and targeted attribute perturbation alignment, which tests whether matched latents respond selectively and in the correct direction to images differing in exactly one concept.

3.1 Latent-Concept Matching

To measure statistical alignment between SAE latents and human-annotated concepts, we compare binary ground-truth attribute annotations with binarized SAE activations. Let $\mathbf{Y} \in \{0,1\}^{A \times N}$ denote the attribute annotation matrix for N samples and A attributes, and let $\mathbf{y}_a \in \{0,1\}^N$ be the row corresponding to attribute a. For a latent unit l, let $\mathbf{z}_l \in \mathbb{R}^N$ denote its activation vector across all samples, where $(\mathbf{z}_l)_i$ is the activation produced by the SAE for sample i. We define the binarized activation vector $\mathbf{z}^l_{\text{bin}} \in \{0,1\}^N$ such that $(\mathbf{z}^l_{\text{bin}})_i = 1$ if $(\mathbf{z}_l)_i > 0$ and 0 otherwise. Alignment between latent unit l and attribute a is then evaluated by comparing $\mathbf{z}^l_{\text{bin}}$ and $\mathbf{y}_a$ using standard binary matching metrics.

One-to-one matching. The SAE latent that best aligns with attribute a under the F_1 score is given by:

$$l^*_a = \arg\max_l F_1(\mathbf{y}_a, \mathbf{z}^l_{\text{bin}}). \tag{2}$$

One-to-one matching assumes that each attribute is represented by exactly one SAE latent. However, feature splitting [5] causes a unified concept to fragment across multiple specialized latents. For instance, 'striped belly' may be split into one latent responding to stripe pattern and one to belly region, such that neither alone achieves high F_1 even though together they fully encode the attribute. This motivates a many-to-one matching formulation, described next.

Many-to-one matching (Fully-Binary Matching Pursuit). Rather than selecting the single best-aligned latent, we seek a small subset (coalition) of latents whose combined activations better reconstruct a given attribute annotation. A naïve approach would select the top-k latents by individual binary similarity score (e.g., F_1), but this tends to produce redundant coalitions rather than complementary ones. We therefore adopt a sequential selection procedure that greedily builds a more informative coalition. The proposed approach is based on well-established greedy techniques from sparse reconstruction literature, namely Matching Pursuit [31] and its variants [37,54]. At each step, the latent that best complements the current coalition is selected, and its contribution is removed from the residual before the next selection. This ensures that each newly added latent is complementary to the previously selected ones rather than redundant. Although best subset selection is NP-hard [33], greedy approaches of this kind have proven effective in practice and optimal under certain conditions [49, 50].

We propose Fully-Binary Matching Pursuit (FBMP), a variant of Matching Pursuit tailored for the reconstruction of binary signals. While binary matching pursuit has been explored in prior work [27, 54], existing formulations target binary coefficient vectors while leaving the input vector and candidate atoms continuous. In our setting, both the attribute annotation vectors and the latent activations are binary, which precludes the use of standard inner products and vector sums. We therefore adapt the matching pursuit procedure to operate entirely in the binary domain. The resulting approach, described in Algorithm 1, is, to the best of our knowledge, novel. In the proposed procedure: (1) a binary similarity metric (e.g., F_β score[6]) is used instead of standard inner-product-based correlation for best-atom selection; (2) logical OR ($\vee$) operations are used to sum the contributions of each atom in the reconstruction, instead of simple sums; (3) logical AND ($\wedge$) and NOT ($\neg$) operations are used to update residuals instead of standard subtraction operations. A stopping criterion terminates the algorithm as soon as a newly-selected latent no longer improves the F_1 score of the current approximation. Although the maximum coalition size k may be set to a large value, the algorithm typically returns a smaller subset (see Appendix S2.2).

While FBMP is many-to-one *per attribute call*, as each attribute selects a subset of latents, the overall matching is many-to-many at the set level: a latent can be selected by multiple attributes simultaneously (see Appendix Fig. S9). Binarizing the latent activations discards magnitude information. This is deliberate: latents that encode several distinct concepts in their magnitude [25] fire

[6] We propose using F_β score with some $\beta \leq 1$ to favor precision over recall. Because multiple latents are to be combined, high recall is not crucial for a single latent.

indiscriminately once binarized and thus align poorly with any single attribute, so they are penalized. We regard such magnitude-encoded polysemanticity as insufficient disentanglement, and therefore as less interpretable. FBMP nonetheless outperforms a magnitude-aware non-negative orthogonal matching pursuit (NN-OMP) [6] baseline in causal alignment (TAPAScore, Sec. 3.2) at matched coalition size (see Appendix Sec. S6.3).

Algorithm 1 Fully-Binary Matching Pursuit (FBMP)

Require: Set of binarized latent activation vectors $\mathbf{z}^l_{\text{bin}} \in \{0,1\}^N$, with $l \in \{1,\dots,L\}$
Target attribute annotations $\mathbf{y} \in \{0,1\}^N$
Max iterations k (max subset size), $\beta \leq 1$ (for concept selection criterion),
Binary similarity metric SIM_{bin} (e.g., F_β).
Ensure: Latents subset $\mathcal{S} \subset \{1,\dots,L\}$ that well approximate $\mathbf{y}$

$\mathbf{r}^0 \leftarrow \mathbf{y}$ ▷ Initialize residual
$\hat{\mathbf{y}}^0 \leftarrow \mathbf{0}$ ▷ Initialize approximation
$\mathcal{S} \leftarrow \emptyset$ ▷ Initialize empty subset of selected concepts
for $\kappa = 0$ **to** $k-1$ **do**
  $l^* \leftarrow \arg\max_l \text{SIM}_{\text{bin}}(\mathbf{r}^\kappa, \mathbf{z}^l_{\text{bin}})$ ▷ Select most-aligned concept
  $\mathbf{r}^{\kappa+1} \leftarrow \mathbf{r}^\kappa \wedge \neg \mathbf{z}^{l^*}_{\text{bin}}$ ▷ Update residual
  $\hat{\mathbf{y}}^{\kappa+1} \leftarrow \hat{\mathbf{y}}^\kappa \vee \mathbf{z}^{l^*}_{\text{bin}}$ ▷ Update approximation
  if $F_1(\mathbf{y}, \hat{\mathbf{y}}^{\kappa+1}) \leq F_1(\mathbf{y}, \hat{\mathbf{y}}^\kappa)$ **then** ▷ Stopping criterion
    break
  $\mathcal{S} \leftarrow \mathcal{S} \cup \{l^*\}$ ▷ Add to subset
return $\mathcal{S}$

Matching Score. Given a set of concepts $\mathcal{S}_a$ matched to attribute a, the corresponding matching score is defined as the F_1 similarity between the coalition of selected latents and the ground-truth annotations $\mathbf{y}_a$. The overall MATCHScore is then obtained by averaging over attributes:

$$\text{MATCHScore} = \frac{1}{A} \sum_{a=1}^{A} F_1(\mathbf{y}_a, \textstyle\bigwedge_{l \in \mathcal{S}_a} \mathbf{z}^l_{\text{bin}}) \tag{3}$$

For one-to-one matching, we simply have $\mathcal{S}_a = \{l^*_a\}$ as defined in Eq. (2). Finally, to enable fair comparisons between SAEs of different sizes, we introduce an *adjusted* matching score. Larger dictionaries provide a larger pool of candidate latent units, which artificially inflates matching performance (see Appendix Fig. S27). To account for this effect, we subtract $F_{1,\text{rng}}$, the matching score achieved by an untrained SAE (random weights) using the same matching strategy, to compute the ΔMATCHScore, which reflects improvement over a random baseline:

$$\Delta\text{MATCHScore} = \text{MATCHScore} - F_{1,\text{rng}}. \tag{4}$$

3.2 Evaluation via Targeted Attribute Perturbation Alignment

A high latent-concept matching score indicates statistical alignment between the SAE's latent activations and the annotated attributes. However, correlation does not imply that a latent encodes the underlying semantic attribute. A latent may correlate with an attribute due to confounding structure, co-occurrence patterns, or background cues, without encoding the semantic factor itself. If a latent truly encodes an attribute, then changing that attribute while holding all other attributes constant should induce a structured and directionally consistent change in the latent activation. We use targeted perturbations to measure this change, by isolating a single semantic attribute while minimizing confounding variation. By constructing paired inputs $(\mathbf{x}, \hat{\mathbf{x}})$ that differ in exactly one annotated attribute, we can test whether latents matched to the perturbed attribute respond in the correct direction, increasing for additions and decreasing for removals.

Let $\mathbf{x}$ denote the original image embedding produced by a pretrained model f, and let $\hat{\mathbf{x}}$ denote the perturbed version that differs in exactly one attribute. Let $\mathbf{z}_{\text{bin}}, \hat{\mathbf{z}}_{\text{bin}} \in \mathbb{R}^L$ denote the binarized SAE latent representations of $\mathbf{x}$ and $\hat{\mathbf{x}}$, respectively, with dictionary size L. Using the latent-concept matching defined previously, let $I_{\text{add}} \subset \{1, \ldots, L\}$ denote the set of latents matched to the attribute being added, and I_{rem} those matched to the attribute being removed (see Fig. 1). We aggregate over each matched set via the maximum (equivalently, a logical OR): the concept is considered active if at least one of its latents fires. The signed response for a single image pair is:

$$\delta_{\text{add}} = \max_{i \in I_{\text{add}}} \hat{\mathbf{z}}^i_{\text{bin}} - \max_{i \in I_{\text{add}}} \mathbf{z}^i_{\text{bin}}, \quad \text{and} \quad \delta_{\text{rem}} = \max_{i \in I_{\text{rem}}} \hat{\mathbf{z}}^i_{bin} - \max_{i \in I_{\text{rem}}} \mathbf{z}^i_{bin}. \tag{5}$$

Averaging over all P image pairs in the dataset yields:

$$\Delta_{\text{add}} = \frac{1}{P} \sum_p \delta^{(p)}_{\text{add}}, \quad \text{and} \quad \Delta_{\text{rem}} = \frac{1}{P} \sum_p \delta^{(p)}_{\text{rem}}. \tag{6}$$

Finally, we define the **T**argeted **A**ttribute **P**erturbation **A**lignment **S**core (TAPAScore) as:

$$\text{TAPAScore} = \Delta_{\text{add}} - \Delta_{\text{rem}}. \tag{7}$$

For datasets containing only removal perturbations (e.g., synCOCO), we set $\Delta_{\text{add}} = 0$. A positive TAPAScore indicates that matched latents respond selectively and in the correct direction under targeted attribute perturbations.

3.3 Synthetic Dataset generation

Evaluating TAPAScore requires image pairs that differ in exactly one semantic attribute while keeping all other factors constant. Since no such benchmark exists for naturalistic images, we construct two synthetic datasets tailored to this requirement: synCUB, which targets fine-grained attribute perturbations in single-object bird images, and synCOCO, which targets object removal in

complex multi-object scenes. To the best of our knowledge, this constitutes the first intervention-style benchmark for SAE evaluation in naturalistic images, as prior work has been largely restricted to simple controlled settings [14].

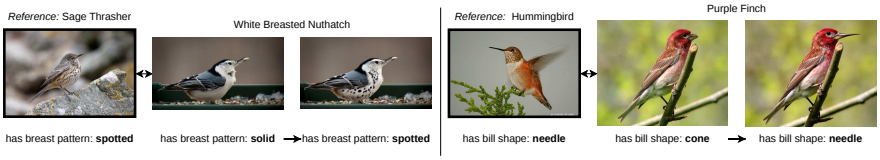

Fig. 2: synCUB pairs: a reference image guides the target attribute (e.g., breast pattern: solid → spotted), while the base image preserves identity, pose, and background.

Synthetic CUB (synCUB, attribute perturbations). CUB-200-2011 [52] provides dense, per-instance attribute annotations across 312 attributes covering color, shape, and pattern across bird parts, making it a natural starting point for attribute-level intervention evaluation. We construct synCUB following the SUB benchmark [1], which modifies CUB by generating prototypical class examples with uncommon attributes, introducing attribute variation across images. However, SUB pairs images across different class instances rather than providing the same bird before and after a single attribute change, making it unsuitable for image-level intervention evaluation. In contrast, we construct synCUB, where each pair $(\mathbf{x}, \hat{\mathbf{x}})$ is constructed such that their attribute annotation vectors differ in exactly one target attribute, while all other annotated attributes remain unchanged (Fig. 2). We restrict ourselves to the 33-class subset used in SUB and their curated set of 45 attribute concepts. The target attributes are selected based on frequency and annotation confidence. For each target attribute, a base image and a reference image exhibiting the desired attribute state are paired and passed to Flux2 [26], which generates an edited image conditioned on both: the base image preserves identity, pose, and background, while the reference guides the target attribute appearance. We validate the generated images using an attribute predictor that verifies the target attribute was correctly modified; samples where the predictor fails are manually curated (see Appendix Sec. S3.4).

Fig. 3: synCOCO pairs: the target object (e.g., umbrella, car) is removed while the remaining scene is preserved.

Synthetic COCO (synCOCO, object removal perturbations). While synCUB evaluates attribute-level alignment in a controlled single-object setting, real-world scenes are considerably more complex. MS-COCO [29] provides a natural complement: its images contain multiple objects, diverse backgrounds, and rich compositional structure that is absent in CUB. As COCO provides no attribute annotations, we treat object labels as concepts, since objects can be considered high-level semantic concepts. We therefore construct synCOCO,

where each pair $(\mathbf{x}, \hat{\mathbf{x}})$ differs in exactly one object label, with all remaining objects preserved (Fig. 3). For each pair, we select the target object to remove by first choosing the object with the lowest instance count in the scene, breaking ties by selecting the one with the largest area. All instances of the selected object are removed via image editing using Flux2 [26], conditioned on the original image and a fixed prompt that instructs the model to remove the target while preserving the remaining scene. To verify that only the target object was removed, we apply the same automatic classifier check as for synCUB, followed by a manual validation of every pair retained in the final dataset. Full details of the image selection procedure, prompts, and validation protocols for both datasets are provided in the Appendix Sec. S3.

4 Experimental Results

Experimental Settings. We evaluate SAE interpretability on two controlled benchmarks derived from CUB-200-2011 [52] and MS-COCO [29]. For both datasets, we construct synthetic intervention benchmarks (synCUB and synCOCO) as described in Sec. 3.3. Latent-concept matching is computed on the original CUB and COCO datasets, while perturbation alignment is evaluated on the synthetic benchmarks. For each dataset, we extract image embeddings from two pretrained vision models, CLIP (ViT-L-14 backbone) and DINOv2 (ViT-S-14 backbone), and train SAEs on these embeddings. We compare four SAE variants: JumpReLU, TopK, BatchTopK, and Matryoshka SAEs, across dictionary sizes {128, 256, 512, 1024, 2048, 4096}. Unless stated otherwise, all SAEs except JumpReLU use TopK sparsity with K=32, which is the average occurrence of attributes in CUB; a sweep over the sparsity K in the Appendix (Sec. S6.4) confirms that moderate sparsity levels provide the best trade-off between matching and perturbation alignment. Full training statistics, including reconstruction loss and sparsity metrics, are listed in the Appendix in Sec. S4. For state-of-the-art comparison, we compare functional proxy metrics commonly used in prior work, namely FMS [18], MS [36], and CKNNA [55]. We evaluate monosemanticity-based scores (FMS [18] and MS [36]), aggregated over all latents. While monosemanticity-based scores were not designed as a metric for evaluating a full SAE, we treat the average over latents as a proxy: if an SAE contains more monosemantic latents, its average score should indicate higher interpretability. For our selection criteria, we evaluate latent-concept matching using classical F_1 (with k=1) and F_β-based matching pursuit (FBMP, with k=3) across $\beta \in \{0.25, 0.5, 1\}$; an analysis of metric sensitivity over k is provided in the Appendix Sec. S6.1. As a supervised upper bound, we additionally train per-attribute logistic regression probes on the raw embeddings and evaluate their thresholded outputs within the same pipeline (Figs. 5 and 6).

Sanity Check Analysis. Evaluating interpretability metrics is inherently challenging due to the absence of reliable ground truth explanations [19,24]. Drawing on the disentanglement literature, where failure modes of common metrics are

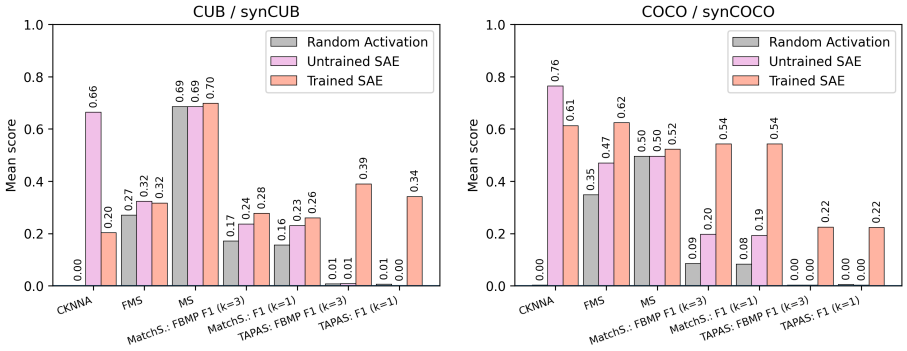

Fig. 4: Metrics failure mode comparison aggregated over all dictionary sizes for CUB (left) and COCO (right) with CLIP, across three conditions: trained SAEs, an untrained TopK SAE, and random activations. TAPAScore (computed on the synthetic datasets) and MATCHScore with FBMP clearly drop under untrained and random conditions, while FMS and MS show little sensitivity, and CKNNA inflates for the untrained SAE.

well-documented [9], we test whether a metric can distinguish meaningful concepts from spurious or random correspondence. Concretely, we compare three conditions: trained SAEs (with scores aggregated over all dictionary sizes and SAE variants), an untrained TopK SAE, and random activations. We report results for CKNNA [55], FMS [18], and MS [36], alongside our MATCHScore with two criteria (FBMP F_1 with k=3 and F_1 with k=1) and the TAPAScore for the respective matching. To compare with the untrained baseline, we evaluate the unnormalized matching score as described in Eq. 3. The metrics and MATCHScores are calculated on the respective dataset (CUB or COCO), whereas TAPAScore is calculated on the corresponding synthetic version.

From Fig. 4, one can observe that both matching variants exhibit a pronounced drop when replacing trained SAEs with either untrained weights or random activations. For both datasets, the MATCHScores for FBMP with selection criterion F_1 (k=3) and F_1 (k=1) show clear absolute decreases in matching F_1 score under the untrained and random conditions. In contrast, MS shows limited sensitivity to whether the representation is trained, untrained, or random across both datasets, while FMS shows this limitation only on CUB but not on COCO. CKNNA behaves inconsistently: while it shows a similar drop for random activations, it exhibits a strong absolute increase for the untrained SAE, indicating that the untrained SAE achieves higher CKNNA scores than the trained models. TAPAScore shows the strongest separation, dropping to near zero for both untrained and random conditions on both datasets. Overall, only our matching variants and TAPAScore pass the intended sanity checks, clearly distinguishing trained SAEs from both untrained and random conditions.

4.1 Latent to Concept Matching

We evaluate semantic alignment between SAE latents and human-annotated concepts across dictionary sizes and SAE variants for CLIP. The matching scores of untrained SAEs increase with dictionary size, as larger dictionaries offer more

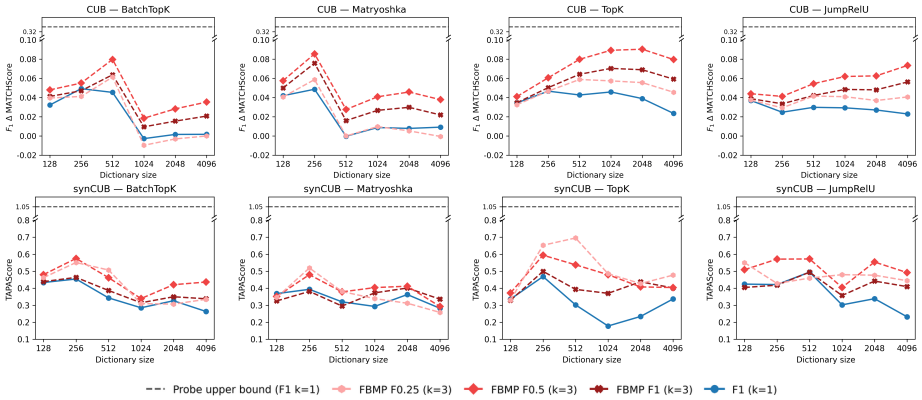

Fig. 5: F_1 ΔMATCHScore on CUB (*top row*) and TAPAScore on synCUB (*bottom row*) as a function of dictionary size for SAEs trained on CLIP embeddings of CUB, across BatchTopK, Matryoshka, TopK and JumpReLU SAE variants (*left to right*) and different matching criteria. The gray dashed line marks the supervised linear-probe upper bound.

latents and thus a greater chance of spurious correlations, a trend we validate in the Appendix (Fig. S27). To enable fair comparison across dictionary sizes, we therefore calculate the ΔMATCHScore (Eq. 4) throughout. The raw scores and results for different models are provided in the Appendix (Sec. S6.1).

Fig. 5 (*top row*) shows the $F_1\Delta$ matching scores across dictionary sizes for BatchTopK, Matryoshka, TopK, and JumpReLU SAEs on CUB with a CLIP model. For each attribute, we find the best matching latent(s) under a given criterion, either one-to-one (F_1, k=1) or different FBMP variants with k=3 (FBMP F_1, FBMP $F_{0.5}$, FBMP $F_{0.25}$). The scores are aggregated over all attributes to obtain a mean F_1 matching score per dictionary size. We observe that FBMP consistently outperforms its one-to-one counterpart across all SAE variants, consistent with the motivation that attributes may be represented by multiple complementary latents rather than a single unit. Among the FBMP variants, FBMP $F_{0.5}$ tends to achieve the highest scores. Matching scores are not monotonically increasing with dictionary size: BatchTopK and Matryoshka peak at dictionary sizes 512 and 256, respectively, after which the score decreases. TopK, in contrast, exhibits a more monotonic increase, peaking at 2048, and therefore outperforms the other variants at large dictionary sizes, though a drop is observed at 4096. JumpReLU shows a dip at dictionary size 256 followed by a steady increase up to 4096, while its one-to-one matching slowly decreases with dictionary size. All SAE variants remain well below the linear-probe upper bound, indicating that only part of the attribute information present in the embeddings is recovered as individual latents.

Fig. 6 (*top row*) reports the F_1 matching scores for BatchTopK, Matryoshka, TopK, and JumpReLU SAEs trained on COCO with CLIP. Similarly to the CUB results, FBMP consistently outperforms the one-to-one baseline across all SAE variants, and FBMP $F_{0.5}$ achieves the highest scores overall. Matching scores are notably higher than on CUB, which we attribute to the smaller and higher-level

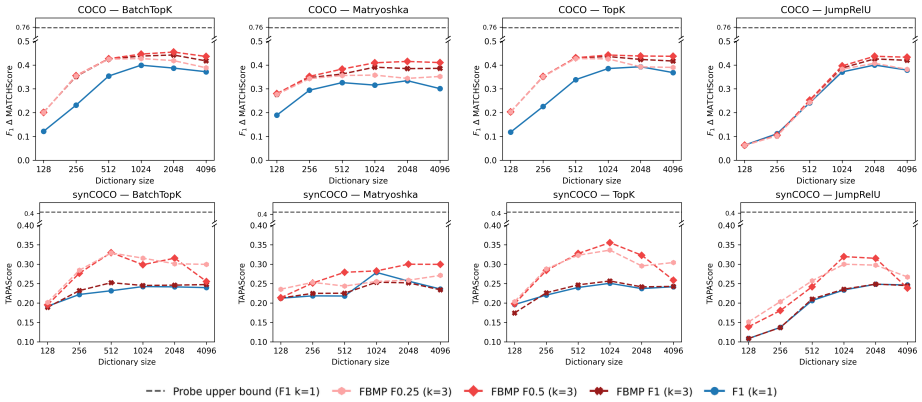

Fig. 6: F_1 ΔMATCHScore on COCO (*top row*) and TAPAScore on synCOCO (*bottom row*) as a function of dictionary size for SAEs trained on CLIP embeddings of COCO, across BatchTopK, Matryoshka, TopK and JumpReLU SAE variants (*left to right*) and different matching criteria. The gray dashed line marks the supervised linear-probe upper bound.

concept set whose categories are more semantically distinct from one another. Unlike in CUB, BatchTopK and Matryoshka do not exhibit a clear peak followed by a decline; instead, their performance remains stable or continues to improve at larger dictionary sizes, suggesting that the greater visual complexity of COCO requires more latents to fully encode its representations. JumpReLU follows the same pattern, increasing monotonically and saturating around dictionary size 2048, although the gap between FBMP and one-to-one matching is smaller than for the other variants.

4.2 Functional Validation of Concept Alignment

While a high matching score indicates statistical alignment between SAE latents and annotated attributes, correlation alone does not imply causal encoding. To validate that matched latents genuinely encode the corresponding semantic attributes, we compute TAPAScore on our synthetic datasets. The SAEs are trained on CLIP embeddings of CUB and COCO. The results for DINOv2 are provided in the Appendix (Sec. S6.2). From Fig. 5 (*bottom row*), we observe that BatchTopK and Matryoshka follow the same qualitative pattern in both metrics: TAPAScore rises to an early peak (around dictionary size 256), declines into a trough at 1024, and partially recovers, mirroring the shape of the matching score (*top row*) though less pronounced. This shared trend suggests that statistical and causal alignment move together for these variants. TopK, in contrast, shows a clear dissociation: while matching keeps improving up to dictionary size 2048, the FBMP $F_{0.5}$ TAPAScore peaks at 256 and then degrades, indicating that larger TopK dictionaries achieve better statistical alignment without encoding causally valid concepts. JumpReLU exhibits the mildest behavior: its FBMP TAPAScores remain comparatively stable, with a slight peak around 512

and only a gradual decline at the largest dictionary sizes, most visible for the one-to-one criterion.

From Fig. 6 (*bottom row*), we observe a larger divergence between matching and TAPAScore than in synCUB. F_1 ΔMATCHScore grows with dictionary size for all SAE variants, saturating around sizes 512–1024. TAPAScore follows a similar trend up to this point, but then declines for larger dictionary sizes in BatchTopK, TopK, and JumpReLU, suggesting that overcompleteness degrades perturbation alignment despite improved matching quality; for JumpReLU this decline only sets in at dictionary size 4096 and is the least pronounced. Matryoshka behaves differently: FBMP $F_{0.5}$ (k=3) continues to improve beyond 1024, reaching its maximum at dictionary size 2048 before plateauing, suggesting that the nested structure of Matryoshka SAEs requires larger dictionaries to fully capture the representations. However, Matryoshka scores are overall lower than those of the other SAE types. In the Appendix (Sec. S6.5) we further verify that TAPAScore is not inflated by leakage: latents matched to untouched attributes remain stable under perturbation.

Correlation analysis. To validate the patterns observed in Secs. 4.1 and 4.2, we examine whether latent-concept matching predicts targeted perturbation alignment. We compute the Pearson correlation between matching scores and TAPAScore across all SAE configurations (variant and dictionary size), with each point in Fig. 7 corresponding to one configuration. On synCUB, we observe a consistent positive correlation across all criteria, with FBMP variants outperforming one-to-one matching, and FBMP $F_{0.5}$ achieving the highest correlation. On synCOCO, the correlations are weaker overall, reflecting the divergence between matching and TAPAScore at larger dictionary sizes, where the MATCHScore continues to improve while TAPAScore declines. Here, FBMP variants yield near-zero correlations, while F_1 k=1 yields a negative correlation.

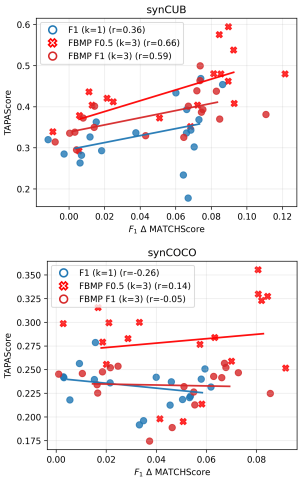

Fig. 7: Correlation between matching score and TAPAScore across SAE configurations on synCUB (*top*) and synCOCO (*bottom*).

Taken together, the results reveal a consistent picture: statistical and causal alignment largely agree, with higher-matching configurations tending to exhibit stronger perturbation alignment, but the correspondence is not universal. TopK on CUB shows that high matching scores do not guarantee causality, and the COCO results show that overcompleteness can reduce perturbation alignment despite higher matching quality. TAPAScore therefore provides a necessary complement that statistical alignment alone cannot replace, and moderate dictionary sizes achieve the best trade-off between the two. We recommend FBMP $F_{0.5}$ with k=3 as the default matching criterion, paired with TAPAScore for causal validation. Selecting by TAPAScore, the strongest configurations are

BatchTopK and TopK at dictionary sizes 512 and 1024, respectively, on COCO, and both variants at dictionary size 256 on CUB.

5 Conclusion

We present a human-grounded framework for evaluating SAE interpretability in vision models, operationalizing interpretability as alignment between learned sparse latents and human-annotated semantic concepts. The framework comprises latent-concept matching criteria, including FBMP, a coalition-based formulation robust to common SAE failure modes; two synthetic intervention benchmarks (synCUB and synCOCO), which provide controlled single-attribute perturbations; and TAPAScore, a functional metric that tests whether matched latents respond selectively and in the correct direction under targeted perturbations. The matching score and TAPAScore are positively correlated on CUB, indicating that stronger statistical alignment predicts more selective latent responses under intervention, though this correspondence weakens on COCO where overcompleteness drives a divergence between the two metrics. Increasing overcompleteness tends to degrade perturbation alignment, and our evaluation consistently favors moderate dictionary sizes as the best trade-off between semantic coverage and functional selectivity. Across variants, BatchTopK and TopK at moderate dictionary sizes provide the best causal alignment, at sizes 512 and 1024 on COCO and 256 on CUB, making them our recommended default; TopK in particular attains strong matching but loses causal alignment once dictionaries grow overcomplete. Our sanity check analysis further shows that both FBMP matching and TAPAScore reliably distinguish trained SAEs from untrained and random baselines, whereas existing metrics such as FMS, MS, and CKNNA do not. As a practical protocol, we recommend FBMP $F_{0.5}$ with coalition size k=3 as the primary matching criterion, paired with TAPAScore for causal validation: FBMP typically returns small coalitions; this criterion correlates most strongly with TAPAScore. Since the matching layer only requires vector-valued concept activations, any concept extractor (linear probes, concept bottleneck models) can be evaluated within the same framework.

The framework has a limitation that suggests a direction for future work. The matching quality is directly bounded by annotation quality and granularity, and since TAPAScore relies on matched latent sets, poor annotations lead to unreliable perturbation alignment estimates. Future work could therefore explore approaches that reduce the dependence on manually curated attribute annotations. Beyond evaluation, a natural extension is to use TAPAScore as a steering tool. Instead of measuring whether matched latents respond to attribute perturbations, they could be actively manipulated to control the predictions of an attribute trained classifier, turning the framework into a test for targeted model intervention.

References

1. Bader, J., Girrbach, L., Alaniz, S., Akata, Z.: Sub: Benchmarking cbm generalization via synthetic attribute substitutions. Proceedings of the IEEE International Conference on Computer Vision (ICCV) (2025)
2. Bau, D., Zhou, B., Khosla, A., Oliva, A., Torralba, A.: Network dissection: Quantifying interpretability of deep visual representations. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
3. Bhalla, U., Oesterling, A., Srinivas, S., Calmon, F.P., Lakkaraju, H.: Interpreting CLIP with sparse linear concept embeddings (SpLiCE). Advances in Neural Information Processing Systems (NeurIPS) (2024)
4. Bhalla, U., Srinivas, S., Ghandeharioun, A., Lakkaraju, H.: Towards unifying interpretability and control: Evaluation via intervention. ArXiv e-print (2024)
5. Bricken, T., Templeton, A., Batson, J., Chen, B., Jermyn, A., Conerly, T., Turner, N., Anil, C., Denison, C., Askell, A., et al.: Towards monosemanticity: Decomposing language models with dictionary learning. Transformer Circuits Thread (2023)
6. Bruckstein, A.M., Elad, M., Zibulevsky, M.: Sparse non-negative solution of a linear system of equations is unique. In: 2008 3rd International Symposium on Communications, Control and Signal Processing. pp. 762–767. IEEE (2008)
7. Bussmann, B., Leask, P., Nanda, N.: BatchTopK sparse autoencoders. NeurIPS 2024 Workshop on Scientific Methods for Understanding Deep Learning (2024)
8. Bussmann, B., Nabeshima, N., Karvonen, A., Nanda, N.: Learning multi-level features with matryoshka sparse autoencoders. Proceedings of the International Conference on Machine Learning (ICML) (2025)
9. Carbonneau, M.A., Zaidi, J., Boilard, J., Gagnon, G.: Measuring disentanglement: A review of metrics. IEEE transactions on neural networks and learning systems **35**(7), 8747–8761 (2022)
10. Chanin, D., Wilken-Smith, J., Dulka, T., Bhatnagar, H., Golechha, S., Bloom, J.: A is for absorption: Studying feature splitting and absorption in sparse autoencoders. Advances in Neural Information Processing Systems (NeurIPS) (2025)
11. Doshi-Velez, F., Kim, B.: Towards a rigorous science of interpretable machine learning. stat (2017)
12. Elhage, N., Hume, T., Olsson, C., Schiefer, N., Henighan, T., Kravec, S., Hatfield-Dodds, Z., Lasenby, R., Drain, D., Chen, C., et al.: Toy models of superposition. ArXiv e-print (2022)
13. Fel, T., Boutin, V., Béthune, L., Cadène, R., Moayeri, M., Andéol, L., Chalvidal, M., Serre, T.: A holistic approach to unifying automatic concept extraction and concept importance estimation. Advances in Neural Information Processing Systems (NeurIPS) (2023)
14. Fel, T., Lubana, E.S., Prince, J.S., Kowal, M., Boutin, V., Papadimitriou, I., Wang, B., Wattenberg, M., Ba, D.E., Konkle, T.: Archetypal SAE: Adaptive and stable dictionary learning for concept extraction in large vision models. Proceedings of the International Conference on Machine Learning (ICML) (2025)
15. Fel, T., Wang, B., Lepori, M.A., Kowal, M., Lee, A., Balestriero, R., Joseph, S., Lubana, E.S., Konkle, T., Ba, D., et al.: Into the rabbit hull: From task-relevant concepts in DINO to minkowski geometry. Proceedings of the International Conference on Learning Representations (ICLR) (2026)
16. Gao, L., la Tour, T.D., Tillman, H., Goh, G., Troll, R., Radford, A., Sutskever, I., Leike, J., Wu, J.: Scaling and evaluating sparse autoencoders. Proceedings of the International Conference on Learning Representations (ICLR) (2025)

17. Ghandeharioun, A., Yuan, A., Guerard, M., Reif, E., Lepori, M., Dixon, L.: Who's asking? user personas and the mechanics of latent misalignment. Advances in Neural Information Processing Systems (NeurIPS) (2024)
18. Härle, R., Friedrich, F., Brack, M., Wäldchen, S., Deiseroth, B., Schramowski, P., Kersting, K.: Measuring and guiding monosemanticity. Advances in Neural Information Processing Systems (NeurIPS) (2025)
19. Hedström, A., Bommer, P., Wickstrøm, K.K., Samek, W., Lapuschkin, S., Höhne, M.M.C.: The meta-evaluation problem in explainable AI: identifying reliable estimators with metaquantus. The Journal of Transactions on Machine Learning Research (TMLR) (2023)
20. Hernandez, E., Sharma, A.S., Haklay, T., Meng, K., Wattenberg, M., Andreas, J., Belinkov, Y., Bau, D.: Linearity of relation decoding in transformer language models. Proceedings of the International Conference on Learning Representations (ICLR) (2024)
21. Huben, R., Cunningham, H., Smith, L.R., Ewart, A., Sharkey, L.: Sparse autoencoders find highly interpretable features in language models. Proceedings of the International Conference on Learning Representations (ICLR) (2024)
22. Joseph, S., Suresh, P., Goldfarb, E., Hufe, L., Gandelsman, Y., Graham, R., Bzdok, D., Samek, W., Richards, B.A.: Steering clip's vision transformer with sparse autoencoders. Mechanistic Interpretability for Vision at CVPR 2025 (Non-proceedings Track) (2025)
23. Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., et al.: Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). Proceedings of the International Conference on Machine Learning (ICML) (2018)
24. Klotz, J., Burgert, T., Demir, B.: On the effectiveness of methods and metrics for explainable AI in remote sensing image scene classification. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (JSTARS) (2025)
25. Kopf, L., Feldhus, N., Bykov, K., Bommer, P.L., Hedström, A., Höhne, M., Eberle, O.: Capturing polysemanticity with prism: A multi-concept feature description framework. Advances in Neural Information Processing Systems (NeurIPS) (2025)
26. Labs, B.F.: FLUX.2: Frontier Visual Intelligence. `https://bfl.ai/blog/flux-2` (2025)
27. Li, H., Ying, H., Liu, X.: Binary generalized orthogonal matching pursuit. Japan Journal of Industrial and Applied Mathematics **41**(1), 1–12 (2024)
28. Lim, H., Choi, J., Choo, J., Schneider, S.: Sparse autoencoders reveal selective remapping of visual concepts during adaptation. ArXiv e-print (2024)
29. Lin, T.Y., Maire, M., Belongie, S.J., Bourdev, L.D., Girshick, R.B., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. CoRR (2014)
30. Makelov, A., Lange, G., Nanda, N.: Towards principled evaluations of sparse autoencoders for interpretability and control. Proceedings of the International Conference on Learning Representations (ICLR) (2025)
31. Mallat, S.G., Zhang, Z.: Matching pursuits with time-frequency dictionaries. IEEE Transactions on signal processing **41**(12), 3397–3415 (1993)
32. Mueller, A., Brinkmann, J., Li, M., Marks, S., Pal, K., Prakash, N., Rager, C., Sankaranarayanan, A., Sharma, A.S., Sun, J., Todd, E., Bau, D., Belinkov, Y.: The quest for the right mediator: Surveying mechanistic interpretability for nlp through the lens of causal mediation analysis. Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) pp. 1–48 (2026)

33. Natarajan, B.K.: Sparse approximate solutions to linear systems. SIAM Journal on Computing **24**(2), 227–234 (1995)
34. Olson, M.L., Hinck, M., Ratzlaff, N., Li, C., Howard, P., Lal, V., Tseng, S.Y.: Analyzing hierarchical structure in vision models with sparse autoencoders. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2025)
35. Oquab, M., Darcet, T., Moutakanni, T., Vo, H.V., Szafraniec, M., Khalidov, V., Fernandez, P., HAZIZA, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P.Y., Li, S.W., Misra, I., Rabbat, M., Sharma, V., Synnaeve, G., Xu, H., Jegou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P.: DINOv2: Learning robust visual features without supervision. The Journal of Transactions on Machine Learning Research (TMLR) (2024)
36. Pach, M., Karthik, S., Bouniot, Q., Belongie, S., Akata, Z.: Sparse autoencoders learn monosemantic features in vision-language models. Advances in Neural Information Processing Systems (NeurIPS) (2025)
37. Pati, Y.C., Rezaiifar, R., Krishnaprasad, P.S.: Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers (1993)
38. Paulo, G.S., Mallen, A.T., Juang, C., Belrose, N.: Automatically interpreting millions of features in large language models. In: Proceedings of the International Conference on Machine Learning (ICML) (2025)
39. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. Proceedings of the International Conference on Machine Learning (ICML) (2021)
40. Rajamanoharan, S., Lieberum, T., Sonnerat, N., Conmy, A., Varma, V., Kramár, J., Nanda, N.: Jumping ahead: Improving reconstruction fidelity with jumprelu sparse autoencoders. ArXiv e-print (2024)
41. Rao, S., Mahajan, S., Böhle, M., Schiele, B.: Discover-then-name: Task-agnostic concept bottlenecks via automated concept discovery. Proceedings of the IEEE European Conference on Computer Vision (ECCV) (2024)
42. Rimsky, N., Gabrieli, N., Schulz, J., Tong, M., Hubinger, E., Turner, A.: Steering llama 2 via contrastive activation addition. Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) (2024)
43. Rubinstein, R., Bruckstein, A.M., Elad, M.: Dictionaries for sparse representation modeling. Proceedings of the IEEE **98**(6), 1045–1057 (2010)
44. Saeed, W., Omlin, C.: Explainable AI (XAI): A systematic meta-survey of current challenges and future opportunities. Knowledge-Based Systems **263**, 110273 (2023)
45. Saphra, N., Wiegreffe, S.: Mechanistic? In: Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP (2024)
46. Shu, D., Wu, X., Zhao, H., Rai, D., Yao, Z., Liu, N., Du, M.: A survey on sparse autoencoders: Interpreting the internal mechanisms of large language models. Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL) (2025)
47. Templeton, A.: Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. Anthropic (2024)
48. Tošić, I., Frossard, P.: Dictionary learning. IEEE Signal Processing Magazine **28**(2), 27–38 (2011)
49. Tropp, J.A.: Greed is good: Algorithmic results for sparse approximation. IEEE Transactions on Information Theory **50**(10), 2231–2242 (2004)

50. Tropp, J.A., Gilbert, A.C.: Signal recovery from random measurements via orthogonal matching pursuit. IEEE Transactions on Information Theory **53**(12), 4655–4666 (2007)
51. Vielhaben, J., Bareeva, D., Berend, J., Samek, W., Strodthoff, N.: Beyond scalars: Concept-based alignment analysis in vision transformers. Advances in Neural Information Processing Systems (NeurIPS) (2025)
52. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset. California Institute of Technology Technical Report (2011)
53. Wattenberg, M., Viégas, F.: Relational composition in neural networks: A survey and call to action. ICML 2024 Workshop on Mechanistic Interpretability (2024)
54. Wen, J., Li, H.: Binary sparse signal recovery with binary matching pursuit. Inverse Problems **37**(6), 065014 (2021)
55. Zaigrajew, V., Baniecki, H., Biecek, P.: Interpreting CLIP with hierarchical sparse autoencoders. Proceedings of the International Conference on Machine Learning (ICML) (2025)