VITLENS: A LIGHTWEIGHT INTERPRETABILITY TOOL FOR VISION TRANSFORMERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Do Vision Transformers (ViTs) represent human-interpretable semantic concepts, and if so, where do they emerge across modules and how do they influence downstream predictions? We introduce ViTLens, an efficient, non-invasive framework for arbitrary-location concept probing and attribution from forward activations using only **10 positive exemplars** per concept. We then estimate each concept's influence on the prediction using our adapted SVARM-LOO and attribute that influence across ViT's modules with MC-Shapley. ViTLens achieves competitive concept identification on CIFAR-100, CUB-200-2011, and ImageNet-1k while fitting up to 1490× faster than concept bottleneck model variants per probing site, and achieves a Pearson correlation of up to 0.95 with exact Shapley value at substantially lower cost. We further demonstrate ViTLens's capabilities in selective concept suppression with minimal collateral damage and reveal that old semantic concepts remain recoverable despite severe catastrophic forgetting.

1 INTRODUCTION

Vision Transformers (ViTs) (Dosovitskiy et al., 2021) form the backbone of many modern vision models (Oquab et al., 2024; Kirillov et al., 2023) and vision-language models (Liu et al., 2023; Wang et al., 2024), yet how their internal computations give rise to a prediction remains poorly understood. Consider a ViT that recognizes a Red-headed Woodpecker (*Melanerpes erythrocephalus*, see Fig. 1), a vibrant bird with a red head, black body and white rump. **Does the model represent these concepts? If so, where they emerge, how they evolve, and if they matter?**

Neural network interpretability has long been a central research focus to answer the questions raised above, yet existing methods exhibit several drawbacks:

- **Computational overhead.** Concept bottleneck model (CBM) variants (Koh et al., 2020; Srivastava et al., 2024) and sparse autoencoder variants (Bussmann et al., 2024; Zaigrajew et al., 2025) require extensive training with backpropagation, while circuit-tracing methods (Olah et al., 2020; Conmy et al., 2023) rely on extensive manual analysis or repeated interventions. This makes extracting interpretable semantic structure extremely costly to scale.
- **Invasive.** CBMs require training an additional concept bottleneck layer (CBL), thereby changing the model's architecture and degrading the task performance as observed by Oikarinen et al. (2023) and Tapli et al. (2026).
- **Restricted coverage.** CBMs expose interpretable concepts only at the CBL, while Logit-Lens variants (Nostalgebraist, 2020; Neo et al., 2025) interpret hidden states through the unembedding layer. Thus, neither supports faithful early-layer probing and concept-level attribution.

Thus, we propose ViTLens, a model interpretability method that is **efficient**, **non-invasive**, and supports **arbitrary-location probing and attribution**. On a high level, ViTLens identifies the existence of a concept with Singular Value Decomposition (SVD), and assigns each concept an attribution score based on a hierarchical estimation of Shapley value at different sites.

Inspired by the view that high-dimensional representations often concentrate near low-dimensional subspaces (Elhamifar & Vidal, 2013; Yu et al., 2020), we hypothesize that semantic concepts exhibit the same structure. This motivates **ViTLens**, which uses only **10 exemplars per concept** from the dataset annotated by Srivastava et al. (2024) to recover user-specified concept subspaces directly

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

from forward activations using SVD, without backpropagation. Across CIFAR-100 (Krizhevsky, 2009), CUB-200-2011 (Wah et al., 2011), and ImageNet-1k (Deng et al., 2009), ViTLens achieves competitive concept identification against learned concept baselines, while being up to $1490\times$ faster than VLG-CBM (Srivastava et al., 2024) to fit per probing site.

While SVD reveals whether a concept is present in the model's activations, it does not tell us whether a concept matters for the prediction, or where its influence arises. Shapley value (Shapley, 1953) is a natural choice for quantifying concept influence across ViT locations, but its exact computation is prohibitively expensive. We therefore propose hierarchical concept–module attribution, which first estimates each concept's overall influence on the prediction and then distributes that influence across modules. Because concept- and module-level attributions exhibit different interaction structures, we introduce **SVARM-LOO**, adapted from SVARM (Kolpaczki et al., 2024), to efficiently refine concept-level Shapley estimates, while using Monte Carlo (MC) Shapley (Castro et al., 2009) at the module level, where downstream propagation induces stronger interactions. Together, our framework achieves up to 0.95 Pearson correlation with exact Shapley while being $5.2\times$ and $180\times$ faster in the concept and module attributions.

Finally, we put ViTLens to work in two case studies. ViTLens enables selective concept suppression with minimal collateral damage by utilizing Rayleigh-quotient-based intervention and reveals what catastrophic forgetting changes ViT geometrically: relative geometry of concepts is largely preserved, and forgetting is induced by influence collapses of concept subspaces, ultimately leading to drop in output performance.

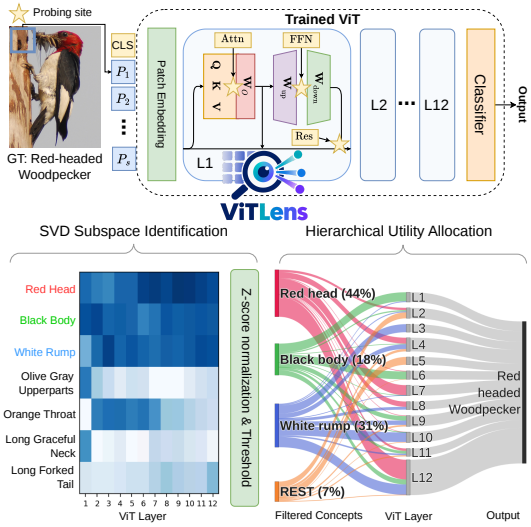


Figure 1: Overview of **ViTLens**. ViTLens probes a trained ViT in activation space by first identifying **which** concept subspaces exist across probing site defined in Eq. (1) with SVD and obtaining a per-probing site energy score $E_c^{(l)}(\boldsymbol{x}^{(l)})$ (Eq. (5)) as shown in the heatmap. After filtering based on the z-score normalized energy, the hierarchical Shapley-value approximation then assigns utilities to tell us **how the concept subspaces evolve** and **whether they matter** for the output decision. Each concept receives an attribution score $\widehat{\Phi}_c$ (§ 3.2), represented by the flow width at the concept end of the Sankey diagram and a concept-module attribution $\widehat{\phi}_{c,l}$ (Eq. (11)) represented by the flow width at the layer end.

2 Background & Related Work

Throughout the paper, we use lowercase letters for scalars a, bold lowercase letters for vectors $\boldsymbol{a}$, bold uppercase letters for matrices $\boldsymbol{A}$, calligraphic letters for sets $\mathcal{A}$, and $\boldsymbol{I}_d$ the $d \times d$ identity matrix. We use $|\mathcal{A}|$ for set cardinality and $\|\boldsymbol{a}\|_2$ for the Euclidean norm. Full details in Tab. A.1.

Problem Statement. Let $f_{\boldsymbol{\theta}}$ be a ViT with fixed parameters $\boldsymbol{\theta}$, ℓ for the Transformer layer index and $l \in \mathcal{L}$ for a probed module, where $\mathcal{L}$ is the set of probing locations. Given an input image I, the model produces a prediction $\hat{y} = f_{\boldsymbol{\theta}}(I)$. Our goal is to interpret the internal representations of $f_{\boldsymbol{\theta}}$ by identifying semantic concepts in its hidden activations. Specifically, let $\mathcal{C} = \{c_1, \ldots, c_M\}$ be a finite, user-specified set of semantic concepts, and let $\mathcal{D}_{\text{probe}} = \{(I_i, y_i, \mathcal{C}_i)\}_{i=1}^{N_{\text{probe}}}$ be a probing dataset, where y_i denotes the ground-truth class and $\mathcal{C}_i \subseteq \mathcal{C}$ the concepts associated with I_i. We use $\mathcal{D}_{\text{probe}}$ to study **what** concepts are encoded, **how** they affect the model's task decision, and **where** they are most influential in the ViT.

The Transformer Modeling. As we focus on dissecting the behavior of ViT (Dosovitskiy et al., 2021), we follow the common Transformer residual modeling from ROME (Meng et al., 2022),

$$\underbrace{\boldsymbol{h}_i^{(\ell)}}_{\text{Res}} = \boldsymbol{h}_i^{(\ell-1)} + \boldsymbol{W}_o \underbrace{\text{Attention}(\boldsymbol{h}_i^{(\ell-1)})}_{\text{Attn}} + \boldsymbol{W}_{\text{down}} \underbrace{\left(\sigma\left(\boldsymbol{W}_{\text{up}} \boldsymbol{h}_i^{(\ell-1)}\right)\right)}_{\text{FFN}}, \tag{1}$$

where $\boldsymbol{h}_i^{(\ell)}$ is the hidden state of token i at layer ℓ and σ denotes the non-linearity. For the rest of this paper, we denote the module Res (Residual) as $\boldsymbol{h}_i^{(\ell)}$, FFN (Feed-forward network) as hidden activation after $\boldsymbol{W}_{\text{up}}$ and non-linearity in the ViT's FFN, and Attn (Attention) as hidden activation before the output matrix of attention, $\boldsymbol{W}_o$ of the attention. A simple illustration is provided in Fig. 1. We will omit token position as we work primarily with the `[CLS]` token only.

2.1 Related Work

Semantic subspaces in weight space. In weight space, Millidge & Black (2022) show that singular vectors of GPT-2 (Radford et al., 2019) weights can encode human-interpretable concepts. Gentile et al. (2026) associate singular directions in CLIP's (Radford et al., 2021) visual attention weights with text-defined concepts, while Xue & Andrzejak (2026) interpret rank-one components of Transformer MLP weights as Detector-Effector Units and quantify their contribution. However, semantic representations need not align with the weight-space directions. Under superposition, concepts may be represented by distributed directions in activation space (Bricken et al., 2023).

Semantic subspaces in activation space. In activation space, CAV (Kim et al., 2018) learns concept directions from positive and negative examples and measures their influence through directional derivatives. Subsequent work extracts semantic structure through latent-space decomposition. Graziani et al. (2023) use singular directions of activations and gradients for unsupervised concept identification, CRAFT (Fel et al., 2023) combines nonnegative matrix factorization with Sobol indices for concept discovery and localization, and Bushnaq et al. (2024) analyze features using PCA-rotated activations together with downstream Jacobian directions. Meanwhile, Zhou et al. (2026) apply SVD to the weights of converged linear probes (Alain & Bengio, 2017) to isolate low-rank subspaces encoding geometric information in ViTs. ViTLens differs in both objective and construction: **(1)** ViTLens uses user-specified **positive** exemplars only (unlike CAV (Kim et al., 2018)) to recover a low-dimensional concept subspace, and **(2)** it operates solely on forward activations, avoiding both the costly gradient or Jacobian computation and the training of task-specific probes.

3 ViTLens

In this section, we introduce the two stages of ViTLens. **(1) Concept subspace identification:** we recover each user-specified concept as a low-dimensional activation subspace using SVD and detect its presence using normalized projection energy. **(2) Hierarchical concept attribution:** we quantify each concept's influence on the prediction and allocate that influence across probing modules using Shapley-value estimators. We obtain empirical results on CIFAR-100 (Krizhevsky, 2009), CUB-200-2011 (Wah et al., 2011) and ImageNet-1k (Deng et al., 2009) with 698, 608, and 4,167 valid concepts with ViT-B/16 and ViT-S/16 (Dosovitskiy et al., 2021). More details in Appendix H.

3.1 Concept Subspace Identification

For a fixed concept $c \in \mathcal{C}$, let $\mathcal{I}_c$ denote a set of $|\mathcal{I}_c| = N_c$ images containing c obtained from the probing set $\mathcal{D}_{\text{probe}}$. Each image in $\mathcal{I}_c$ may also contain other concepts $c' \neq c$. A forward pass with $\mathcal{I}_c$ through $f_{\boldsymbol{\theta}}$ generates the input activation $\boldsymbol{X}_c^{(l)} \in \mathbb{R}^{d_l \times N_c}$ at module l. We use $N_c = 10$ throughout the paper (ablation in Tab. F.1) for ViTLens.

Subspace fitting. Similar to Elhamifar & Vidal (2013); Yu et al. (2020), we hypothesize that the semantic concepts lie in low-dimensional subspaces in the model's feature space. A classic method to identify such a subspace is to perform SVD on the activation induced by $\mathcal{I}_c$ in the l-th module

$$\boldsymbol{X}_c^{(l)} = \tilde{\boldsymbol{U}}_c^{(l)} \boldsymbol{\Sigma}_c^{(l)} \boldsymbol{V}_c^{(l)\top}. \tag{2}$$

As we assume that semantic concepts lie on a low-dimensional subspace, we instead take $\boldsymbol{U}_c^{(l)} = \widetilde{\boldsymbol{U}}_{c,\,1:r}^{(l)}$, where we fix $r = 8$ for the rest of the paper (ablation in Tab. F.2). Define the projection matrix onto the concept subspace as

$$\boldsymbol{P}_c^{(l)} = \boldsymbol{U}_c^{(l)} \boldsymbol{U}_c^{(l)^\top}. \tag{3}$$

Thus, for an arbitrary input activation $\boldsymbol{x}^{(l)}$, the output activation can then be decomposed as

$$\boldsymbol{h}^{(l)} = \boldsymbol{W}^{(l)} \boldsymbol{x}^{(l)} = \boldsymbol{W}^{(l)} \boldsymbol{P}_c^{(l)} \boldsymbol{x}^{(l)} + \boldsymbol{W}^{(l)} (\boldsymbol{I}_{d_l} - \boldsymbol{P}_c^{(l)}) \boldsymbol{x}^{(l)}, \tag{4}$$

Table 1: Concept macro-F1 (%, ↑), evaluated with ViT-S/16 and ViT-B/16 (Dosovitskiy et al., 2021). Standard deviations are shown as subscripts. We use **bold** for best and underline for second best.

Method	Setting	CIFAR-100		CUB-200-2011		ImageNet-1k	
		S/16	B/16	S/16	B/16	S/16	B/16
CBM	Last Res	$\underline{58.11}_{\pm 0.30}$	$\mathbf{57.59}_{\pm 0.33}$	$16.55_{\pm 0.96}$	$11.15_{\pm 0.22}$	$41.14_{\pm 0.12}$	$46.46_{\pm 0.29}$
VLG-CBM	Last Res	$54.50_{\pm 0.18}$	$54.65_{\pm 0.05}$	$\underline{73.65}_{\pm 0.08}$	$\underline{69.76}_{\pm 0.08}$	$\mathbf{49.14}_{\pm 0.01}$	$\mathbf{52.41}_{\pm 0.03}$
Linear probe	FFN	$17.75_{\pm 0.46}$	$37.12_{\pm 0.30}$	$37.33_{\pm 0.56}$	$35.85_{\pm 0.40}$	$13.57_{\pm 0.20}$	$17.64_{\pm 0.39}$
	Attn	$28.20_{\pm 0.31}$	$32.17_{\pm 0.28}$	$43.77_{\pm 0.21}$	$40.82_{\pm 0.19}$	$17.87_{\pm 0.31}$	$13.38_{\pm 0.34}$
	Res	$11.46_{\pm 0.32}$	$40.50_{\pm 0.44}$	$38.05_{\pm 0.32}$	$43.51_{\pm 0.21}$	$14.83_{\pm 0.22}$	$16.35_{\pm 0.03}$
	Last Res	$\mathbf{58.28}_{\pm 0.13}$	$55.80_{\pm 0.08}$	$\mathbf{73.87}_{\pm 0.22}$	$61.83_{\pm 0.20}$	$33.83_{\pm 0.11}$	$11.55_{\pm 0.04}$
KMeans	FFN	$26.46_{\pm 0.68}$	$46.18_{\pm 0.11}$	$33.00_{\pm 0.47}$	$39.92_{\pm 0.03}$	$17.73_{\pm 0.10}$	$27.04_{\pm 0.25}$
	Attn	$34.31_{\pm 0.45}$	$45.85_{\pm 0.23}$	$39.32_{\pm 0.42}$	$41.83_{\pm 0.25}$	$24.31_{\pm 0.01}$	$29.87_{\pm 0.06}$
	Res	$31.87_{\pm 0.60}$	$41.02_{\pm 0.22}$	$40.81_{\pm 0.25}$	$44.58_{\pm 0.12}$	$18.81_{\pm 0.03}$	$29.99_{\pm 0.15}$
	Last Res	$55.21_{\pm 0.16}$	$52.43_{\pm 0.42}$	$65.46_{\pm 0.12}$	$48.80_{\pm 0.86}$	$44.39_{\pm 0.09}$	$46.81_{\pm 0.12}$
kNN	FFN	$29.54_{\pm 0.41}$	$56.19_{\pm 0.21}$	$51.61_{\pm 0.27}$	$56.51_{\pm 0.24}$	$27.48_{\pm 0.06}$	$43.67_{\pm 0.11}$
	Attn	$53.83_{\pm 0.16}$	$48.53_{\pm 0.08}$	$41.90_{\pm 0.39}$	$53.23_{\pm 0.22}$	$31.45_{\pm 0.04}$	$33.37_{\pm 0.04}$
	Res	$1.80_{\pm 0.01}$	$55.78_{\pm 0.09}$	$2.83_{\pm 0.02}$	$60.15_{\pm 0.16}$	$34.03_{\pm 0.17}$	$44.27_{\pm 0.03}$
	Last Res	$1.72_{\pm 0.00}$	$56.04_{\pm 0.15}$	$66.07_{\pm 0.02}$	$62.53_{\pm 0.23}$	$43.64_{\pm 0.13}$	$40.61_{\pm 0.06}$
CAV	FFN	$28.79_{\pm 0.14}$	$42.50_{\pm 0.12}$	$25.70_{\pm 0.14}$	$43.54_{\pm 0.74}$	$18.00_{\pm 0.23}$	$34.23_{\pm 0.09}$
	Attn	$35.05_{\pm 0.49}$	$43.19_{\pm 0.09}$	$31.44_{\pm 0.22}$	$43.28_{\pm 0.09}$	$22.48_{\pm 0.13}$	$28.66_{\pm 0.11}$
	Res	$27.46_{\pm 0.21}$	$45.13_{\pm 0.33}$	$28.34_{\pm 0.31}$	$45.77_{\pm 0.11}$	$15.30_{\pm 0.18}$	$34.58_{\pm 0.12}$
	Last Res	$56.07_{\pm 0.19}$	$56.47_{\pm 0.10}$	$70.26_{\pm 0.22}$	$67.98_{\pm 0.13}$	$39.91_{\pm 0.02}$	$45.76_{\pm 0.16}$
ViTLens	FFN	$27.54_{\pm 0.23}$	$\underline{57.14}_{\pm 0.17}$	$49.73_{\pm 0.30}$	$62.25_{\pm 0.19}$	$30.16_{\pm 0.06}$	$46.21_{\pm 0.10}$
	Attn	$50.80_{\pm 0.19}$	$50.48_{\pm 0.07}$	$59.13_{\pm 0.40}$	$61.53_{\pm 0.25}$	$35.99_{\pm 0.01}$	$38.97_{\pm 0.03}$
	Res	$30.97_{\pm 0.28}$	$55.84_{\pm 0.11}$	$45.12_{\pm 0.17}$	$67.71_{\pm 0.12}$	$42.68_{\pm 0.02}$	$\underline{48.20}_{\pm 0.02}$
	Last Res	$57.64_{\pm 0.18}$	$57.09_{\pm 0.14}$	$72.56_{\pm 0.11}$	$\mathbf{69.96}_{\pm 0.04}$	$\underline{46.57}_{\pm 0.05}$	$44.54_{\pm 0.05}$

where $\boldsymbol{W}^{(l)}\boldsymbol{P}_c^{(l)}\boldsymbol{x}^{(l)}$ is the component of output activation contributed by concept c, where $\boldsymbol{W} \in \mathbb{R}^{k_l \times d_l}$ is the weight of module l ($\boldsymbol{W}^{(l)} = \boldsymbol{I}_{d_l}$ for Res).

Concept Identification. Suppose that we have obtained $P_c^{(\ell)}$ for all c, l using $\mathcal{D}_{\text{probe}}$. To identify the concepts for an arbitrary test image I_{test} which have activation $\boldsymbol{x}^{(l)}$. We quantify the activation energy, E, in $\boldsymbol{x}^{(l)}$ captured by $\boldsymbol{P}_c$ according to Eq. (4), and the z-score normalized ViTLens score, $\hat{E}$ at module l of concept c as

$$E_c^{(l)}(\boldsymbol{x}^{(l)}) = \frac{\|\boldsymbol{P}_c^{(l)}\boldsymbol{x}^{(l)}\|_2^2}{\|\boldsymbol{x}^{(l)}\|_2^2}, \qquad \hat{E}_c^{(l)}(\boldsymbol{x}^{(l)}) = \frac{E_c^{(l)}(\boldsymbol{x}^{(l)}) - \mu_E^{(l)}}{\sigma_E^{(l)}}, \tag{5}$$

where $\mu_E^{(l)}$ and $\sigma_E^{(l)}$ are the mean and standard deviation of $\{E_{c'}^{(l)}(\boldsymbol{x}^{(l)})\}_{c' \in \mathcal{C}}$ for the same input image I and module ℓ. We define the concept to "exist" if $\hat{E}_c^{(l)}(\boldsymbol{x}^{(l)}) > \tau$ where τ is the threshold. Energy-module heatmap and Lucid (Olah et al., 2017) visualization are shown in Figs. I.1 and I.2.

Can ViTLens identify concepts faithfully? We evaluate concept identification using macro-F1 against the test-set concept annotations provided by Srivastava et al. (2024) for CIFAR-100, CUB-200-2011, and ImageNet-1k. In Tab. 1, ViTLens remains within 0.96 macro-F1 points of VLG-CBM (Srivastava et al., 2024) on average, while using only 10 positive exemplars per concept, requiring no backpropagation, and achieving up to $1490\times$ faster fitting per probing site (Tab. F.4). In contrast, CBM (Koh et al., 2020) and VLG-CBM require training with backpropagation on the full training set. By operating directly on the trained ViT's activations without modifying the model, ViTLens also preserves the backbone's classification accuracy, avoiding the degradation observed in CBM variants (Tab. H.2) and reported by Oikarinen et al. (2023) and Tapli et al. (2026).

Beyond the final layer, ViTLens enables concept probing throughout the network, whereas CBM and VLG-CBM operate only at the bottleneck layer. We therefore compare against methods that support probing at different locations under similar data budgets, namely Linear Probe (Alain & Bengio, 2017), KMeans (Arthur & Vassilvitskii, 2007), kNN (Cover & Hart, 1967), and CAV (Kim et al., 2018). ViTLens remains robust across probing sites, outperforming the second-best method, kNN, by 7.46 macro-F1 points on average across FFN, Attn, and Res locations, while CBM and kNN occasionally suffer from instability. Implementation details are provided in Appendix H. Moreover, ViTLens supports incremental concept addition at the fitting stage, since each subspace is fitted independently with positive exemplars only without modifying the previously fitted subspaces.

Concept localization and semantic geometry. Beyond concept identification, we leverage ViTLens to study the spatial and geometric structure of ViT. First, we show that ViTLens can perform concept localization by applying `[CLS]`-fitted subspaces to `[PATCH]` tokens in Fig. B.1. We then visualize Res concept subspaces across layers using UMAP (McInnes et al., 2018) with chordal distance (Def. C.7) to show how concept subspaces become increasingly separated from the middle layers onward. We also construct a binary tree using chordal distance, with depth determined by the layer at which each subspace is fitted. GPT-4 then labels each internal node based on the concept annotations of its descendant leaves (Fig. I.3; Appendix I.2). The results show that even Kaiming-initialized (He et al., 2015) ViTs exhibit non-trivial geometry, while fine-tuning endows ViT with the capability to arrange the subspaces in a semantically meaningful way.

3.2 Hierarchical Attribution of Concept Influence

Now that we have established that Eq. (5) can effectively identify concepts, the next question is, for some given image I, how much does each concept c contribute to the model's output $\hat{y}$?

Intervention and coalition utility. At module l, intervening on concept c removes the component of the activation lying in the identified concept subspace:

$$\boldsymbol{h}^{(l)*} \leftarrow \boldsymbol{h}^{(l)} - \boldsymbol{W}^{(l)} \boldsymbol{P}_c^{(l)} \boldsymbol{x}^{(l)}. \tag{6}$$

Let $\mathcal{N} = \{(c, l) \mid c \in \mathcal{C}, l \in \mathcal{L}\}$ denote the set of concept-module pairs. A *coalition* $\mathcal{S} \subseteq \mathcal{N}$ specifies a set of concept-module interventions to be applied jointly, where $(c, l) \in \mathcal{S}$ means that concept c is intervened at module l. Let $\boldsymbol{p}_{\mathcal{S}}(\cdot \mid I)$ denote the intervened output distribution and $\boldsymbol{p}_{\varnothing}(\cdot \mid I)$ the clean output. We define the utility of the coalition as the Kullback-Leibler Divergence,

$$v_{\mathcal{S}}(I) = D_{\mathrm{KL}}(\boldsymbol{p}_{\varnothing}(\cdot \mid I) \,\|\, \boldsymbol{p}_{\mathcal{S}}(\cdot \mid I)), \tag{7}$$

with $v_{\varnothing}(I) = 0$ and $v_{\mathcal{S}}(I) \geq 0$ for every coalition S due to Gibbs' inequality.

Shapley value. Our goal is to determine how much each concept at each module contributes to the model's prediction. An intuitive and mathematically grounded method will be Shapley value (Shapley, 1953). Informally, Shapley value is defined as the average change in utility caused by adding one element to all possible subsets of the remaining elements (see Def. C.1 for formal definition). In our setting, each element is a concept-module pair (c, l), and a subset $\mathcal{S}$ specifies the pairs jointly intervened on according to Eq. (6), with utility $v_{\mathcal{S}}(I)$ defined by Eq. (7).

Exact Shapley value, however, requires evaluating exponentially many coalitions and in our case, requires $2^{|\mathcal{C}||\mathcal{L}|}$ evaluations. For example, a simple CIFAR-100 dataset and ViT-Small require $|\mathcal{C}| = 698$ and $|\mathcal{L}| = 12$, completely defeating the efficiency desiderata. Thus, given a test image, we retain concepts with summed ViTLens score $\sum_{l \in \mathcal{L}} \hat{E}_c^{(l)}(\boldsymbol{x}^{(l)}) > \tau_g$, yielding $\mathcal{C}_f$. For computational tractability, we further restrict to $3 \leq |\mathcal{C}_f| \leq 7$. Even so, $2^{|\mathcal{C}_f||\mathcal{L}|}$ is still computationally intractable. Hence, we take a step back and first compute the contribution of each concept c across all $l \in \mathcal{L}$.

Concept Attribution. As shown in the cross-layer subspace analysis Figs. I.2 and I.3, concept subspaces become highly distinguishable from one another starting from the middle layer of ViT. This is most likely due to neural collapse (Papyan et al., 2020), where semantic subspaces are largely orthogonal to each other in the later layers of neural networks. Such separation suggests that interactions between concept subspaces may be limited, thus motivating the approximation of each concept's true Shapley value with Leave-one-Out (LOO) (Cook, 1977) defined as

$$v_c(I) \coloneqq v_{\{(c,l):\, l \in \mathcal{L}\}}(I), \tag{8}$$

where the Shapley value is approximated by the influence of each concept evaluated independently.

Empirically, Fig. 2 shows that LOO achieves a competitive Pearson and Spearman correlation at around 0.8 with respect to the true Shapley value despite requiring only one intervention per concept, which is extremely cheap. To understand when such a simple approximation is accurate, we characterize its error via the Harsanyi interaction (Harsanyi, 1963; Rota, 1964).

Corollary 3.1 (Exact LOO–Shapley gap (Proof in Appendix C.3)). *Let $m_v(T)$ denote the Harsanyi dividend (Def. C.2), $\forall c \in \mathcal{C}$, the error of LOO score, $v_c(I)$ with respect to Shapley value $\phi_c(I)$ is*

$$\phi_c(I) - v_c(I) = \sum_{T \subseteq \mathcal{C},\ c \in T,\ |T| \geq 2} \frac{m_v(T)}{|T|}. \tag{9}$$

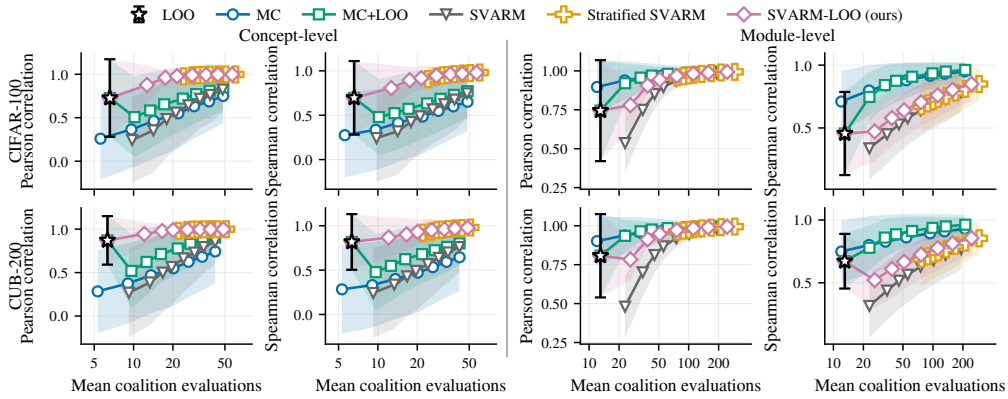


Figure 2: Pearson and Spearman correlations of LOO (Cook, 1977), MC-Shapley (Castro et al., 2009) (MC), MC-Shapley with LOO initialization (MC+LOO), SVARM, Stratified SVARM (Kolpaczki et al., 2024) and our adapted SVARM-LOO under different computation budgets for 400 samples selected from CUB-200-2011 and CIFAR-100 datasets.

Hence, the LOO-Shapley gap is exactly determined by the pairwise interactions involving c. When concepts do not interact with each other, LOO will give the exact Shapley value, and we further show that under regularity assumptions, the chordal distance between subspaces bounds the higher-order Harsanyi interaction in Prop. C.10.

LOO refinement. Even though LOO is cheap and accurate, its estimate cannot be refined once all size-1 coalitions have been evaluated. Let $\widehat{\Phi}_c$ denotes the concept attribution score ($\tilde{\Phi}_c = v_c$ for LOO), SVARM (Kolpaczki et al., 2024) decomposes Shapley value in terms of strata representing different coalition sizes (we provide details of SVARM in Appendix D.1 for a self-contained work). Let $n = |\mathcal{C}|$, and let $\mu^+_{c,k}$ denotes the mean value of size-k coalitions containing c, and let $\mu^-_{c,k}$ denotes the mean value of size-k coalitions excluding c, the SVARM estimate is given by

$$\tilde{\Phi}_c = \frac{1}{n}\left(\sum_{k=1}^{n} \mu^+_{c,k} - \sum_{k=0}^{n-1} \mu^-_{c,k}\right), \tag{10}$$

where $\mu^-_{c,0} = v_\emptyset(I) = 0$ and $\mu^+_{c,n} = v_\mathcal{C}(I)$. After all singleton ($k = 1$) coalitions are evaluated, LOO already determines $\mu^+_{c,1} = v_c(I)$ and $\mu^-_{c,1} = \frac{1}{n-1}\sum_{c' \neq c} v_{c'}(I)$ exactly. Evaluating the grand coalition $k = n$ additionally determines $\mu^+_{c,n}$, leaving only the remaining strata to be estimated.

Kolpaczki et al. (2024) further improves SVARM with Stratified SVARM, where it first evaluates all coalitions of sizes 1, $n-1$, and n, by a warm-up of every remaining player-specific stratum. However, this initialization is costly under a limited evaluation budget. We therefore adapt Stratified SVARM to reuse the evaluations already obtained by LOO. Our adapted **SVARM-LOO** instead initializes from the LOO evaluations, evaluates the grand coalition once, and then samples coalition sizes $2, \ldots, n-1$ approximately uniformly to estimate the remaining strata and progressively refine $\tilde{\Phi}_c$. SVARM-LOO achieves competitive attribution quality at approximately $1.9\times$ the LOO budget, while Stratified SVARM first produces estimates at approximately $3.7\times$ as shown in Fig. 2.

Module attribution. Having estimated the overall attribution of each concept, we next ask *where* a fixed concept c is most influential? Therefore, we fix c and compute the attribution over the module-wise interventions $\{(c, l) : l \in \mathcal{L}\}$, where each element corresponds to ablating concept c at one module. Under this formulation, a coalition is thus a subset $S \subseteq \mathcal{L}$ of modules at which the fixed concept c is ablated, with utility defined by Eq. (7). Unlike concept attribution, ablating a concept subspace at any module l affects all downstream activations, inducing interactions across subsets of $\mathcal{L}$, as illustrated by Eq. (1). Consequently, LOO, which evaluates each module independently, poorly captures these interactions, as shown empirically in Fig. 2 and by the high marginal-cancellation ratio (MCR; Eq. (30) in Appendix C.2). We therefore use MC-Shapley (Castro et al., 2009) to estimate the module-level attribution $\widetilde{\phi}_{c,l}$.

To standardize the attribution of modules across concepts, we obtain the normalized attribution score of concept c at module l, $\hat{\phi}_{c,l}(I)$ as

$$\widehat{\phi}_{c,l} \coloneqq \widehat{\Phi}_c \frac{\widetilde{\phi}_{c,l}}{\sum_{l' \in \mathcal{L}} \widetilde{\phi}_{c,l'}}, \tag{11}$$

where $\widehat{\Phi}_c \coloneqq v_{\mathcal{N}}(I) \frac{\widetilde{\Phi}_c}{\sum_{c' \in \mathcal{C}} \widetilde{\Phi}_{c'}}$, is the normalized concept attribution and $\mathcal{N} = \{(c,l) \mid c \in \mathcal{C}, l \in \mathcal{L}\}$. Consequently, $\sum_{l \in \mathcal{L}} \widehat{\phi}_{c,l} = \widehat{\Phi}_c$, and $\sum_{c \in \mathcal{C}} \sum_{l \in \mathcal{L}} \widehat{\phi}_{c,l} = v_{\mathcal{N}}(I)$. This conceptually follows prior structure-aware Shapley decompositions (Tarun et al., 2024). Under Assumption C.12, we show that theoretically the exact Shapley values at both levels recover the Shapley value of the full concept-module attribution exactly in Prop. C.13 and empirically, it allows up to 0.88 Pearson correlation with the exact Shapley value under a computationally tractable setting in Appendix D.3.

Putting everything together, at $1.9\times$ and $1.8\times$ the respective LOO budgets, our framework obtains average Pearson and Spearman correlations of 0.95 and 0.85 with the concept- and module-level Shapley values across CUB-200-2011 and CIFAR-100 as shown in Fig. 2. Average speedups over Shapley value are $5.2\times$ and $180\times$ for the concept- and module-level attribution, respectively. The resulting concept-module allocations are visualized in Figs. 1, D.1 and D.2 as Sankey diagrams, with details in Appendix D. Notably, different concepts within the same image can be most influential at different layers, revealing no single critical layer. This complements the layer-wise analysis in Dorszewski et al. (2025), while going beyond to quantify the concept's influence on the prediction.

4 Case studies

Now that we have established ViTLens as an efficient and non-invasive model interpretability method, we will utilize it to study some key phenomena and applications.

4.1 Can concepts be selectively removed?

Unlike linear probes (Alain & Bengio, 2017), ViTLens represents each concept as an explicit low-dimensional subspace of the model's activation space, enabling both concept identification and direct intervention. We exploit this structure for sample-dependent suppression at inference time, removing components associated with a target class while preserving other detected concepts. This is more challenging than standard unlearning because **(1)** non-target classes may share concepts with the target class, and **(2)** the concepts to suppress or retain vary across inputs.

Let $\mathcal{C}_s$ denote the concepts associated with target class y that are to be suppressed. For each test image I_{test}, ViTLens detects the present concepts $\mathcal{C}(I_{\text{test}})$ and partitions them into two disjoint sets: the concepts to be suppressed, $\mathcal{C}_{\text{sup}}(I_{\text{test}}) = \mathcal{C}(I_{\text{test}}) \cap \mathcal{C}_s$, and the concepts to be retained, $\mathcal{C}_{\text{ret}}(I_{\text{test}}) = \mathcal{C}(I_{\text{test}}) \setminus \mathcal{C}_s$. Inspired by the suppression-retention trade-off in unlearning evaluation (Fan et al., 2024), we define the suppressed-class accuracy (SA↓) and selectivity by accuracy on overlapping (OA↑) and non-overlapping (NOA↑) non-target classes. For example, when suppressing Red-headed Woodpecker, a non-target class such as Brewer's Blackbird that shares the *black body* concept is overlapping, whereas a class sharing none of the suppressed concepts is non-overlapping. Note that SA, OA and NOA are evaluated on the test-set; details iare provided Appendix E.1.

For each module l, let $\mathcal{C}_{\text{sup}}$ and $\mathcal{C}_{\text{ret}}$ denote the detected concepts to be suppressed and retained, respectively. We construct $\mathbf{A}_{\text{sup}}^{(l)} = \sum_{c \in \mathcal{C}_{\text{sup}}} \mathbf{P}_c^{(l)}$, and $\mathbf{A}_{\text{ret}}^{(l)} = \sum_{c \in \mathcal{C}_{\text{ret}}} \mathbf{P}_c^{(l)} + \epsilon \mathbf{I}$, where $\mathbf{P}_c^{(l)} = \mathbf{U}_c^{(l)} \mathbf{U}_c^{(l)\top}$ as defined in Eq. (3). We identify directions that favor the suppressed concepts while minimizing overlap with retained concepts by solving the generalized eigenproblem $\mathbf{A}_{\text{sup}}^{(l)} \mathbf{u}_i = \lambda_i \mathbf{A}_{\text{ret}}^{(l)} \mathbf{u}_i$, where λ_i measures the ratio of suppressed- to retained-concept energy along $\mathbf{u}_i$. We solve this problem within the joint concept span (details in Appendix E), set $\eta_i = \min(1, \lambda_i)$, collect the generalized eigenvectors as $\mathbf{D}^{(l)} = [\mathbf{u}_1, \ldots, \mathbf{u}_m]$, and define $\mathbf{M}^{(l)} = \mathbf{D}^{(l)} \operatorname{diag}(\sqrt{\eta_i})$. Suppression is then performed as

$$\mathbf{x}^{(l)*} = \mathbf{x}^{(l)} - \mathbf{M}^{(l)} \mathbf{M}^{(l)\top} \mathbf{x}^{(l)}. \tag{12}$$

Unlike direct projection, our sample-dependent Rayleigh-quotient construction selectively suppresses detected target concepts while preserving retained concepts by softly attenuating shared directions and strongly suppressing those specific to $\mathcal{C}_{\text{sup}}$ similar to Qian et al. (2026).

Table 2: Concept suppression results across CIFAR-100 (Krizhevsky, 2009), CUB-200-2011 (Wah et al., 2011), and ImageNet-100 (Tian et al., 2020), a 100-class subset of ImageNet-1k (Deng et al., 2009) used to keep the class-wise suppression evaluation computationally feasible. Diff-in-Mean+ denotes Diff-in-Mean (Arditi et al., 2024) incorporated with ViTLens's concept detection mechanism. Shapley-1 denotes the specific layer with highest utility determined with Eq. (11). We use **Bold** for best and underline for second best.

Method	Budget	SA ↓	NOA ↑	OA ↑
CIFAR-100	Base	88.40	88.72	88.19
Diff-in-Mean Arditi et al. (2024)	Last	$10.99_{\pm 2.34}$	$84.64_{\pm 1.30}$	$75.56_{\pm 2.04}$
	All	$13.25_{\pm 0.67}$	$73.11_{\pm 3.08}$	$66.15_{\pm 2.10}$
Diff-in-Mean+	Last	$20.08_{\pm 0.92}$	$88.69_{\pm 0.01}$	$87.77_{\pm 0.24}$
	All	$19.74_{\pm 0.89}$	$88.71_{\pm 0.01}$	$87.75_{\pm 0.24}$
ViTLens	Shapley-1	$\underline{4.81}_{\pm 0.28}$	$\underline{88.74}_{\pm 0.00}$	$\mathbf{88.29}_{\pm 0.01}$
	All	$\mathbf{0.68}_{\pm 0.05}$	$\mathbf{88.77}_{\pm 0.01}$	$\underline{88.10}_{\pm 0.03}$
CUB-200-2011	Base	83.88	83.31	83.94
Diff-in-Mean Arditi et al. (2024)	Last	$31.88_{\pm 1.07}$	$75.89_{\pm 0.42}$	$72.51_{\pm 0.33}$
	All	$26.14_{\pm 1.34}$	$71.44_{\pm 0.55}$	$66.61_{\pm 0.51}$
Diff-in-Mean+	Last	$38.96_{\pm 0.84}$	$83.29_{\pm 0.01}$	$\underline{83.38}_{\pm 0.13}$
	All	$34.95_{\pm 0.18}$	$83.31_{\pm 0.01}$	$83.25_{\pm 0.07}$
ViTLens	Shapley-1	$\underline{6.36}_{\pm 0.27}$	$\underline{83.33}_{\pm 0.00}$	$\mathbf{83.68}_{\pm 0.04}$
	All	$\mathbf{5.85}_{\pm 0.08}$	$\mathbf{83.34}_{\pm 0.00}$	$82.46_{\pm 0.12}$
IMAGENET-100	Base	94.20	94.40	93.95
Diff-in-Mean Arditi et al. (2024)	Last	$\underline{27.11}_{\pm 2.49}$	$94.37_{\pm 0.03}$	$93.92_{\pm 0.03}$
	All	$51.18_{\pm 0.89}$	$91.62_{\pm 0.39}$	$90.63_{\pm 0.57}$
Diff-in-Mean+	Last	$41.21_{\pm 0.66}$	$\underline{94.40}_{\pm 0.01}$	$\underline{93.98}_{\pm 0.01}$
	All	$56.64_{\pm 0.49}$	$\mathbf{94.40}_{\pm 0.01}$	$\mathbf{93.99}_{\pm 0.02}$
ViTLens	Shapley-1	$34.12_{\pm 1.36}$	$94.39_{\pm 0.00}$	$93.89_{\pm 0.03}$
	All	$\mathbf{24.27}_{\pm 1.07}$	$94.38_{\pm 0.00}$	$93.63_{\pm 0.14}$

As reported in Tab. 2, ViTLens achieves an SA as low as 4.81% with an intervention at only the single layer with highest utility identified with Eq. (11). More impressively, the NOA and OA metrics experience virtually no drop (maximum reduction of 0.02 and 1.48 respectively) from the base model even when all layers are intervened on. This is due to ViTLens correctly identifies the concept to be suppressed, thus, preventing unwanted interference.

4.2 ARE SEMANTIC CONCEPTS ACTUALLY FORGOTTEN?

In this subsection, we leverage ViTLens to study catastrophic forgetting (McCloskey & Cohen, 1989; McClelland et al., 1995), where a model's performance on previously learned tasks degrades after it is subsequently trained on new tasks. Because forgetting is typically quantified by the loss of old-task accuracy, it is often interpreted as evidence that previously acquired knowledge has been overwritten (French, 1999; Parisi et al., 2019; Luo et al., 2024). However, Davari et al. (2022) challenged this interpretation for ResNets (He et al., 2016), and our results reveal a similar discrepancy between performance loss and the forgetting of underlying knowledge for ViT. We study this distinction in ViT-B/16 and ViT-T/16 sequentially trained on five CUB-200-2011 tasks over three random permutations constructed by grouping species sharing a common-name category to reduce semantic overlap: Warbler, Sparrow, Gull, Tern, and Vireo. Full details in Appendix G.

First, Fig. 3 reproduces the observation by Davari et al. (2022) where a 10-shot refitted classifier recovers a substantial fraction of the lost accuracy and outperforms classic continual-learning baselines of ER (Riemer et al., 2019), EWC (Kirkpatrick et al., 2017) and LwF (Li & Hoiem, 2017). ViTLens then further reveals that the discrepancy extends to the semantic level in Fig. 3. When we refit the concept bank after each task, F1-score and the fraction of top-ranked concepts associated with the correct old class degrade substantially less than classification accuracy. Fig. G.1 further shows that concept F1 and top-5 concept share remain stable under ER, EWC, and LwF as well. This suggests that semantic information can remain recoverable even as its geometric realization drifts. Importantly, semantic retention does not imply that the original concept subspace remains fixed. A concept bank frozen immediately after an old task progressively degrades under subsequent models as shown (see Fig. G.2), indicating representational drift beyond a trivial coordinate change (e.g., a shared rotation), but this drift need not erase the semantic structure itself.

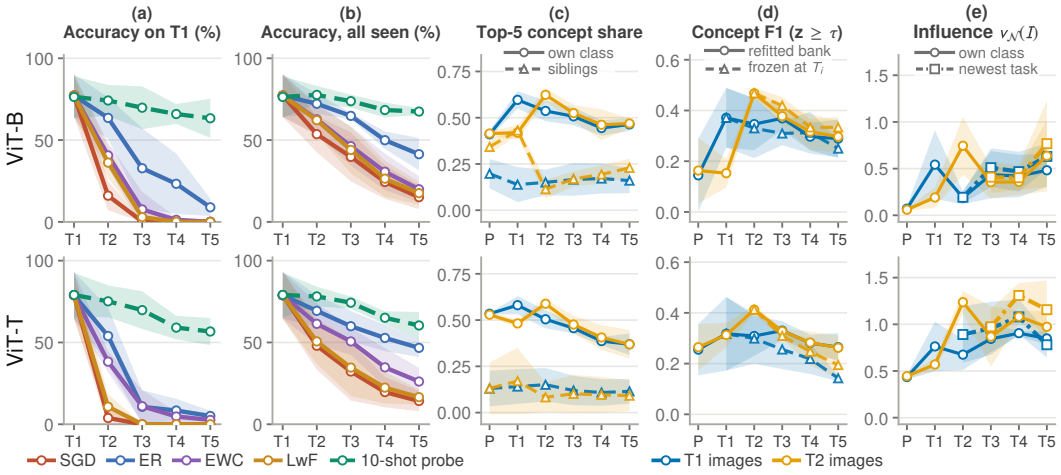


Figure 3: Behavioral versus semantic forgetting in ViT-B/16 and ViT-T/16. P: pretrained checkpoint; T_i: checkpoint after task i. Siblings are concepts from the same task but a different class. **(a)**, **(b)** A 10-shot classifier refit recovers old-task accuracy for the very first task (T_1) and also all previous task T_{i-1}. **(c)**, **(d)** By refitting ViTLens after training on Task T_i for T_1 and T_2 images, we observe that the fraction of correct concept within the concept with top-5 and the F1-score have minimal deterioration despite significant drop in prediction accuracy. **(e)** We then compare the influence, $v_{\mathcal{N}}(I)$ (Eq. (7)), of the correct concepts (own class) against concepts of the latest task (newest task). Immediately after training on the new task, the influence of the correct concepts becomes indistinguishable from the new tasks concepts, which leads to the decrease in accuracy.

As illustrated in Fig. 3, by analyzing the total influence based on the concept intervention defined in Eq. (7), the influence of the correct concepts from previous tasks drops sharply and approaches that of unrelated concepts. This implies that the representation can retain an old concept while its alignment with the model's current decision function deteriorates, leading to intervention with Eq. (7) inducing smaller changes in the output distribution. We also note that ViT-T/16 forgets substantially more than ViT-B/16 despite its higher initial performance from ImageNet-21k pretraining, likely due to its smaller dimensionality and greater subspace overlap (Fig. G.3). This is consistent with prior findings that larger models are more robust to forgetting (Ramasesh et al., 2022; Davari et al., 2022). Since mitigating catastrophic forgetting is beyond our scope, we leave this as an open question for future work.

5 Limitations & Conclusion

Limitations. ViTLens inherently relies on the representation quality of the underlying base model; suboptimal latent spaces can degrade ViTLens performance. Furthermore, this work analysis focuses primarily on `[CLS]` tokens. While we show that `[CLS]`-fitted subspaces transfer to `[PATCH]` tokens in Appendix B for concept localization, extending concept attribution across spatial tokens remains a future work.

Conclusion. We introduced ViTLens, an efficient and non-invasive framework that supports arbitrary-location probing and attribution to study what semantic concepts are represented in a ViT, where they emerge, and how they influence the model's output. ViTLens identifies these concepts as low-dimensional activation subspaces and combines them with a hierarchical concept-module attribution. Across three datasets, ViTLens achieves competitive concept identification with few exemplars and enables selective concept suppression and the analysis of catastrophic forgetting. Our results also raise several open questions: **(1)** whether ViTLens can provide an efficient probing framework for LLMs and VLMs; **(2)** whether its observations on semantic retention and influence can inform new approaches to mitigating catastrophic forgetting; and **(3)** whether ViTLens can be used as a probing tool to determine whether unlearning methods such as Thudi et al. (2022) and Fan et al. (2024), truly erase underlying semantic representations or merely reduce their influence.

AI USE STATEMENT

In this work, we used generative AI tools to provide critical ingredients for proving mathematical claims, assist in the writing of proofs, design or provide feedback on research methodology or experiments, implement methods, and support qualitative and thematic data analysis. We also use AI to create or modify scientific figures or images, suggest experimental parameters, create or edit software code, discover research topics or identify gaps and generate ViTLens's logo in Fig. 1.

We have not used generative AI tools to help develop theoretical models or conceptual frameworks, formulate mathematical claims, assist with translation, and interpret results. Moreover, we have not used generative AI tools to generate synthetic datasets, propose or refine hypotheses, or clean or reformat datasets. But do note that we use AI-annotated datasets from Srivastava et al. (2024).

We have reviewed all AI-assisted work. All equations and text are checked and for the LLM-generated code, a cross-check was performed between different LLM agents, and one author manually stepped through the core parts line-by-line. All citations in the paper are done manually with the help of Google Scholar and DBLP only. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Details necessary to reproduce our experiments, including implementation details, hyperparameters, dataset construction, baselines, and additional ablations, are provided in Appendix H. Proofs and derivations are provided in Appendices B, C, D, and E, with additional experimental details in Appendix G. We will release our code upon acceptance.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Workshop Track Proceedings*. OpenReview.net, 2017.

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. *Advances in Neural Information Processing Systems*, 37:136037–136083, 2024.

David Arthur and Sergei Vassilvitskii. k-means++: the advantages of careful seeding, p 1027–1035. *Society for Industrial and Applied Mathematics*, 2007.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2(5):6, 2023.

Lucius Bushnaq, Stefan Heimersheim, Nicholas Goldowsky-Dill, Dan Braun, Jake Mendel, Kaarel Hänni, Avery Griffin, Jörn Stöhler, Magdalena Wache, and Marius Hobbhahn. The local interaction basis: Identifying computationally-relevant and sparsely interacting features in neural networks. *arXiv preprint arXiv:2405.10928*, 2024.

Bart Bussmann, Patrick Leask, and Neel Nanda. Batchtopk sparse autoencoders. In *NeurIPS 2024 Workshop on Scientific Methods for Understanding Deep Learning*, 2024.

Javier Castro, Daniel Gómez, and Juan Tejada. Polynomial calculation of the shapley value based on sampling. *Computers & operations research*, 36(5):1726–1730, 2009.

Jianbo Chen, Le Song, Martin J. Wainwright, and Michael I. Jordan. L-shapley and c-shapley: Efficient model interpretation for structured data. In *International Conference on Learning Representations*, 2019.

Vikram S Chundawat, Ayush K Tarun, Murari Mandal, and Mohan Kankanhalli. Zero-shot machine unlearning. *IEEE Transactions on Information Forensics and Security*, 18:2345–2354, 2023.

Arthur Conmy, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. *Advances in Neural Information Processing Systems*, 36:16318–16352, 2023.

R. Dennis Cook. Detection of influential observation in linear regression. *Technometrics*, 19(1): 15–18, 1977.

Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27, 1967.

MohammadReza Davari, Nader Asadi, Sudhir Mudur, Rahaf Aljundi, and Eugene Belilovsky. Probing representation forgetting in supervised and unsupervised continual learning. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16691–16700. IEEE, 2022.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. IEEE, 2009.

Teresa Dorszewski, Lenka Tětková, Robert Jenssen, Lars Kai Hansen, and Kristoffer Knutsen Wickstrøm. From colors to classes: Emergence of concepts in vision transformers. In *World Conference on Explainable Artificial Intelligence*, pp. 28–47. Springer, 2025.

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

Ehsan Elhamifar and René Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE transactions on pattern analysis and machine intelligence*, 35(11):2765–2781, 2013.

Chongyu Fan, Jiancheng Liu, Yihua Zhang, Eric Wong, Dennis Wei, and Sijia Liu. Salun: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. In *International Conference on Learning Representations*, volume 2024, pp. 53643–53673, 2024.

Thomas Fel, Agustin Picard, Louis Bethune, Thibaut Boissin, David Vigouroux, Julien Colin, Rémi Cadène, and Thomas Serre. Craft: Concept recursive activation factorization for explainability. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2711–2721, 2023.

Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4):128–135, 1999.

Francesco Gentile, Nicola Dall'Asen, Francesco Tonini, Massimiliano Mancini, Lorenzo Vaquero, and Elisa Ricci. From weights to concepts: Data-free interpretability of clip via singular vector decomposition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2895–2906, 2026.

Mara Graziani, Laura O'Mahony, An phi Nguyen, Henning Müller, and Vincent Andrearczyk. Uncovering unique concept vectors through latent space decomposition. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856.

John C Harsanyi. A simplified bargaining model for the n-person cooperative game. *International Economic Review*, 4(2):194–220, 1963.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pp. 2668–2677. PMLR, 2018.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *2023 IEEE/CVF international conference on computer vision (ICCV)*, pp. 3992–4003. IEEE, 2023.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International conference on machine learning*, pp. 5338–5348. Pmlr, 2020.

Patrick Kolpaczki, Viktor Bengs, Maximilian Muschalik, and Eyke Hüllermeier. Approximating the shapley value without marginal contributions. In *Proceedings of the AAAI conference on Artificial Intelligence*, volume 38, pp. 13246–13255, 2024.

A Krizhevsky. Learning multiple layers of features from tiny images. *Master's thesis, University of Toronto*, 2009.

Isaac Ning Lee, Leila Mahmoodi, Trung Le, and Mehrtash Harandi. Exemplar-free continual learning for state space models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25077–25087, June 2026.

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

Paul Pu Liang, Yun Cheng, Xiang Fan, Chun Kai Ling, Suzanne Nie, Richard Chen, Zihao Deng, Nicholas Allen, Randy Auerbach, Faisal Mahmood, et al. Quantifying & modeling multimodal interactions: An information decomposition framework. *Advances in Neural Information Processing Systems*, 36:27351–27393, 2023.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

Yan Luo, Yongkang Wong, Mohan Kankanhalli, and Qi Zhao. Learning to predict gradients for semi-supervised continual learning. *IEEE Transactions on Neural Networks and Learning Systems*, 36(2):2593–2607, 2024.

James L McClelland, Bruce L McNaughton, and Randall C O'Reilly. Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. *Psychological review*, 102(3):419, 1995.

Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pp. 109–165. Elsevier, 1989.

Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

Kevin Meng, David Bau, Alex J Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. In *Advances in neural information processing systems*, 2022.

Beren Millidge and Sid Black. The singular value decompositions of transformer weight matrices are highly interpretable. In *AI Alignment Forum*, pp. 17, 2022.

Roger B Myerson. Graphs and cooperation in games. *Mathematics of operations research*, 2(3): 225–229, 1977.

Clement Neo, Luke Ong, Philip Torr, Mor Geva, David Krueger, and Fazl Barez. Towards interpreting visual information processing in vision-language models. In *International Conference on Learning Representations*, volume 2025, pp. 57172–57189, 2025.

Nostalgebraist. Interpreting GPT: The logit lens. AI Alignment Forum, August 2020. URL https://www.alignmentforum.org/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens. August 31, 2020.

Tuomas Oikarinen, Subhro Das, Lam M. Nguyen, and Tsui-Wei Weng. Label-free concept bottleneck models. In *The Eleventh International Conference on Learning Representations*, 2023.

Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2017. doi: 10.23915/distill.00007. https://distill.pub/2017/feature-visualization.

Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. *Distill*, 5(3):e00024–001, 2020.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856.

Vardan Papyan, Xiao Y Han, and David L Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40): 24652–24663, 2020.

German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural networks*, 113:54–71, 2019.

Chengyao Qian, Jing Wu, Trung Le, Dinh Phung, and Mehrtash Harandi. ph-strips for selective forgetting: A blunt but fast diagnostic baseline for machine unlearning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3306–3315, June 2026.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Vinay Venkatesh Ramasesh, Aitor Lewkowycz, and Ethan Dyer. Effect of scale on catastrophic forgetting in neural networks. In *International Conference on Learning Representations*, 2022.

Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, , and Gerald Tesauro. Learning to learn without forgetting by maximizing transfer and minimizing interference. In *International Conference on Learning Representations*, 2019.

Gian-Carlo Rota. On the foundations of combinatorial theory i. theory of möbius functions. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 2(4):340–368, 1964.

LS Shapley. A value for n-person games. In *Contributions to the Theory of Games, Volume II*. 1953.

Divyansh Srivastava, Ge Yan, and Tsui-Wei Weng. Vlg-cbm: Training concept bottleneck models with vision-language guidance. *Advances in Neural Information Processing Systems*, 37:79057–79094, 2024.

Merve Tapli, Quentin Bouniot, Wolfgang Stammer, Zeynep Akata, and Emre Akbas. Rethinking concept bottleneck models: From pitfalls to solutions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9901–9910, June 2026.

Ayush Tarun, Vikram Chundawat, Murari Mandal, Hong Ming Tan, Bowei Chen, and Mohan Kankanhalli. Ecoval: An efficient data valuation framework for machine learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2866–2875, 2024.

Anvith Thudi, Gabriel Deza, Varun Chandrasekaran, and Nicolas Papernot. Unrolling sgd: Understanding factors influencing machine unlearning. In *2022 IEEE 7th European symposium on security and privacy (EuroS&P)*, pp. 303–319. IEEE, 2022.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *European conference on computer vision*, pp. 776–794. Springer, 2020.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. Jul 2011.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Jing Wu and Mehrtash Harandi. Scissorhands: Scrub data influence via connection sensitivity in networks. In *European Conference on Computer Vision*, pp. 367–384. Springer, 2024.

Min Xue and Artur Andrzejak. SVD as a fast interpretability method for transformers. In *Forty-third International Conference on Machine Learning*, 2026.

Ke Ye and Lek-Heng Lim. Schubert varieties and distances between subspaces of different dimensions. *SIAM Journal on Matrix Analysis and Applications*, 37(3):1176–1197, 2016.

Yaodong Yu, Kwan Ho Ryan Chan, Chong You, Chaobing Song, and Yi Ma. Learning diverse and discriminative representations via the principle of maximal coding rate reduction. *Advances in neural information processing systems*, 33:9422–9434, 2020.

Vladimir Zaigrajew, Hubert Baniecki, and Przemyslaw Biecek. Interpreting CLIP with hierarchical sparse autoencoders. In *Forty-second International Conference on Machine Learning*, 2025.

Weichen Zhou, Yawen Zou, Chunzhi Gu, Ran Dong, Haoran Xie, and Chao Zhang. Understanding geometric representations in self-supervised vision transformers via subspace intervention. *arXiv preprint arXiv:2607.01987*, 2026.

A NOTATIONS

Table A.1: Summary of recurring notation used throughout the paper.

Symbol	Description	Introduced in
Model and activations		
f_θ, θ	Pre-trained/Fine-tuned model with parameter θ.	§ 2
ℓ	The layer index in the model.	§ 2
$\mathcal{C},\ c$	Set of semantic concepts and an individual concept.	§ 2
$l,\ L,\ \mathcal{L}$	Probed module index, total number of probed modules and set of modules to be probed.	§ 2
d_l	Activation dimension at module l.	§ 3.1
$\mathbf{X}^{(l)},\ \mathbf{x}^{(l)}$	Activation matrix and an individual activation at module l.	§ 3.1
Concept subspaces		
$\mathcal{I}_c,\ N_c$	Set of exemplar images containing concept c and its size, $N_c = \lvert\mathcal{I}_c\rvert$.	§ 3.1
$\mathbf{X}_c^{(l)}$	Activations at module l obtained from the exemplars of concept c.	§ 3.1
r	Rank of the retained concept subspace.	§ 3.1
$\mathbf{U}_c^{(l)}$	Orthonormal rank-r basis of concept c at module l, obtained from the retained left singular vectors.	§ 3.1
$\mathbf{P}_c^{(l)}$	Orthogonal projector onto the concept subspace, $\mathbf{P}_c^{(l)} = \mathbf{U}_c^{(l)}\mathbf{U}_c^{(l)\top}$.	§ 3.1
$E_c^{(l)}(\mathbf{x}^{(l)})$	Fraction of activation energy captured by concept c at module l.	§ 3.1
$\widehat{E}_c^{(l)},\ \tau$	Z-score normalized concept energy and threshold used for concept detection.	§ 3.1
Concept–module attribution		
$\mathcal{N}$	Set of concept-module players in the cooperative game.	§ 3.2
$\mathcal{G}_c$	Module-wise players associated with concept c.	§ 3.2
$\mathcal{S}$	Coalition of players.	§ 3.2
$v_{\mathcal{S}}(I)$	Intervention damage of coalition S for input I, measured by KL divergence from the clean output distribution.	§ 3.2
$\phi_i(v)$	Shapley value of player i under utility function v.	§ 3.2
$\widetilde{\Phi}_c$	Estimated concept-level attribution of concept c.	§ 3.2
$\widehat{\Phi}_c$	Normalized concept-level attribution of concept c.	§ 3.2
$\widetilde{\phi}_{c,l}$	Estimated module-level attribution of module l for concept c.	§ 3.2
$\widehat{\phi}_{c,l}$	Final concept–module attribution of concept c at module l.	§ 3.2
$m_\nu(T)$	Harsanyi dividend of concept coalition T.	§ 3.2
Concept geometry and intervention		
$\theta_{cc',j}^{(l)}$	j-th principal angle between the subspaces of concepts c and c' at module l.	Appendix C.3.2
$\delta_{cc'}^{\mathrm{ch},(l)}$	Normalized chordal distance between the subspaces of concepts c and c' at module l.	Appendix C.3.2
$\mathcal{C}_{\mathrm{sup}},\ \mathcal{C}_{\mathrm{ret}}$	Detected concepts selected for suppression and retention, respectively.	§ 4.1
$\mathbf{M}^{(l)}$	Sample-dependent suppression matrix applied at module l.	§ 4.1
$\mathrm{Overlap}^{(l)}(n)$	Fraction of the current-task concept subspace explained by concept directions associated with previous tasks.	Appendix G

B CONCEPT PATCH VISUALIZATION

In order to improve efficiency of ViTLens while reducing the probing set size, ViTLens works on the `[CLS]` token instead of the `[Patch]` token of ViT. This means that transferring the ViTLens score of Eq. (5) from `[CLS]` token to `[Patch]` tokens is non-trivial due to token mismatch and inter-concept entanglement. Below, we will show the modification onto § 3 in order to generate better visualization.

For concept c at module l, the identified concept subspace projection matrix is $\boldsymbol{P}_c^{(l)} = \boldsymbol{U}_c^{(l)} \boldsymbol{U}_c^{(l)\top}$. Then, for all $c \in \mathcal{C}$ we define the module-wise aggregate concept operator

$$\boldsymbol{G}^{(l)} \coloneqq \sum_{c \in \mathcal{C}} \boldsymbol{P}_c^{(l)} = \boldsymbol{A}^{(l)} \boldsymbol{A}^{(l)\top} \tag{13}$$

where

$$\boldsymbol{A}^{(l)} = [\boldsymbol{U}_1^{(l)} | \dots | \boldsymbol{U}_C^{(l)}]. \tag{14}$$

Since $\boldsymbol{G}^{(l)} = \boldsymbol{A}^{(l)} \boldsymbol{A}^{(l)\top}$, its dominant eigenspace captures directions with maximal aggregate overlap across concept subspaces: for any unit vector $\boldsymbol{u}$,

$$\boldsymbol{u}^\top \boldsymbol{G}^{(l)} \boldsymbol{u} = \sum_{c \in \mathcal{C}} \left\| \boldsymbol{U}_c^{(l)\top} \boldsymbol{u} \right\|_2^2. \tag{15}$$

Let $\mathbf{S}^{(l)} \in \mathbb{R}^{d_l \times k}$ contain the top-k eigenvectors of $\boldsymbol{G}^{(l)}$, with $\mathbf{S}^{(l)\top} \mathbf{S}^{(l)} = \mathbf{I}_k$. We remove these shared directions from each concept basis by

$$\overline{\boldsymbol{U}}_c^{(l)} = \left(\mathbf{I}_{d_l} - \mathbf{S}^{(l)} \mathbf{S}^{(l)\top} \right) \boldsymbol{U}_c^{(l)}. \tag{16}$$

Although $\boldsymbol{U}_c^{(l)}$ has orthonormal columns, $\overline{\boldsymbol{U}}_c^{(l)}$ in general does not. We thus ensure the orthonormality by using QR decomposition,

$$\overline{\boldsymbol{U}}_c^{(l)} \;=\; \boldsymbol{Q}_c^{(l)} \boldsymbol{R}_c^{(l)}, \qquad \boldsymbol{Q}_c^{(l)} \in \mathbb{R}^{d_l \times r}, \quad \boldsymbol{Q}_c^{(l)\top} \boldsymbol{Q}_c^{(l)} = \boldsymbol{I}_r, \tag{17}$$

with $\boldsymbol{R}_c^{(l)} \in \mathbb{R}^{r \times r}$ upper triangular, and define

$$\widetilde{\boldsymbol{P}}_c^{(l)} \;=\; \boldsymbol{Q}_c^{(l)} \boldsymbol{Q}_c^{(l)\top}. \tag{18}$$

Thus $\widetilde{\boldsymbol{P}}_c^{(l)}$ is the orthogonal projector onto the concept subspace after removing the shared component.

For `[Patch]` token $\mathbf{x}_p^{(l)} \in \mathbb{R}^{d_l}$, we define the patch-level concept energy analogously to Eq. (5) as

$$\widetilde{E}_{c,p}^{(l)} = \frac{\left\| \boldsymbol{Q}_c^{(l)\top} \mathbf{x}_p^{(l)} \right\|_2^2}{\left\| \mathbf{x}_p^{(l)} \right\|_2^2} = \frac{\left\| \widetilde{\boldsymbol{P}}_c^{(l)} \mathbf{x}_p^{(l)} \right\|_2^2}{\left\| \mathbf{x}_p^{(l)} \right\|_2^2}. \tag{19}$$

Whenever $\overline{\boldsymbol{U}}_c^{(l)}$ has full column rank, $\mathrm{range}(\boldsymbol{Q}_c^{(l)}) = \mathrm{range}(\overline{\boldsymbol{U}}_c^{(l)}) \perp \mathrm{range}(\boldsymbol{S}^{(l)})$ and hence $\boldsymbol{S}^{(l)\top} \boldsymbol{Q}_c^{(l)} = 0$, so the shared component of $\boldsymbol{x}_p^{(l)}$ contributes nothing to Eq. (19).

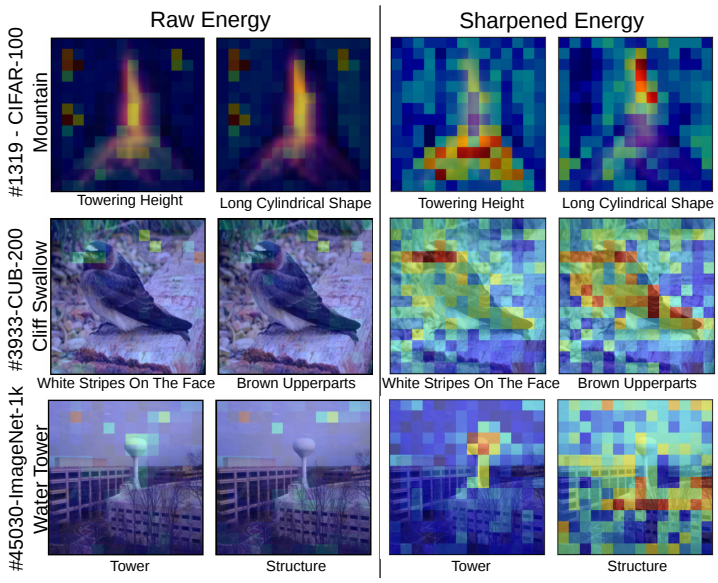


Figure B.1: CIFAR-100 Krizhevsky (2009), CUB-200-2011 Wah et al. (2011), and ImageNet-1k Deng et al. (2009) patch concept visualization. Applying Eq. (19) significantly improves the quality of concept segmentation.

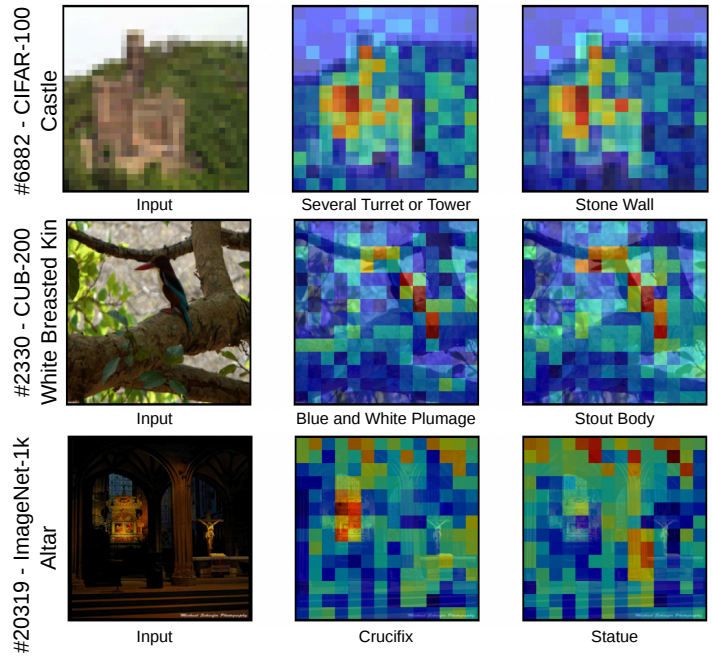


Figure B.2: CIFAR-100 (Krizhevsky, 2009), CUB-200-2011 (Wah et al., 2011) and ImageNet-1k (Deng et al., 2009) patch concept failure cases visualization.

C CONCEPT & MODULE ATTRIBUTION

Definition C.1 (Shapley Value). *Let $\mathcal{N} = \{1, \ldots, N\}$ be a set of players, and let $v : 2^N \to \mathbb{R}$ be a characteristic function assigning a value to each coalition. For any player $i \in \mathcal{N}$, the Shapley value of player i is*

$$\phi_i(v) = \frac{1}{N} \sum_{k=0}^{N-1} \binom{N-1}{k}^{-1} \sum_{S \subseteq \mathcal{N} \setminus i \ |S|=k} [v(S \cup i) - v(S)] \,. \tag{20}$$

It measures the expected marginal contribution of player i to the coalition value over all coalitions not containing i.

The Shapley value satisfies the following properties.

1. **Linearity**: $\phi_i(\alpha v + \beta w) = \alpha\phi_i(v) + \beta\phi_i(w)$ for $\alpha, \beta \in \mathbb{R}$ and some $v, w : 2^{\mathcal{N}} \to \mathbb{R}$.
2. **Efficiency**: $\sum_{i \in \mathcal{N}} \phi_i(v) = v(\mathcal{N})$
3. **Symmetry**: If $v(S \cup \{i\}) = v(S \cup j)$ for every subset S of $\mathcal{N}$ which contains neither i nor j, then $\phi_i(v) = \phi_j(v)$.

Concept shortlist and REST player. Since evaluating the concept game over the full concept bank is intractable, we construct a sample-dependent player set. For each image, we retain concepts whose aggregated z-scored concept energy exceeds $\tau_g = 2$, while clamping the shortlist size to $k_{\min} = 3$ and $k_{\max} = 7$ using the highest-energy concepts when necessary. All remaining concepts are grouped into a single REST player $\mathcal{R}$. At module l, we form

$$\boldsymbol{G}_{\mathcal{R}}^{(l)} = \sum_{c \notin \mathcal{C}_I} \boldsymbol{P}_c^{(l)}, \tag{21}$$

and define $\boldsymbol{P}_{\mathcal{R}}^{(l)}$ as the projector onto its top-r eigenspace. We use $r = 8$, matching the rank of each individual concept subspace. The resulting concept game therefore contains $\mathcal{P}_I = \mathcal{C}_I \cup \{\mathcal{R}\}$, with $4 \leq |\mathcal{P}_I| \leq 8$.

C.1 HIERARCHICAL ATTRIBUTION OF CONCEPT INFLUENCE

In this section, we set up the § 3.2 with a more generalized notation for the derivations that follow.

By treating each concept-module pair (c, l) as a player, let $\mathcal{N}$ denote the set of all such players and $S \subseteq \mathcal{N}$ a coalition of (c, l) pairs to be ablated, to measure the influence of concept c at module l, we remove its contribution by

$$\boldsymbol{h}^{(l)\,*} \leftarrow \boldsymbol{h}^{(l)} - \boldsymbol{W}^{(l)} \boldsymbol{P}_c^{(l)} \boldsymbol{x}^{(l)}. \tag{22}$$

Let $\boldsymbol{p}_{\varnothing}(\cdot \mid I)$ and $\boldsymbol{p}_S(\cdot \mid I)$ denote, respectively, the output distributions of the original and intervened models given input image I and coalition S. We measure the utility of coalition S by the Kullback-Leibler Divergence (KLD) due to the output distribution shift,

$$v_I(S) = D_{\mathrm{KL}} \left(\boldsymbol{p}_{\varnothing}(\cdot \mid I) \,\|\, \boldsymbol{p}_S(\cdot \mid I) \right), \tag{23}$$

and $v_I(\varnothing) = 0$ and $v_I(S) \geq 0$ for every coalition S due to Gibbs' inequality. We did not follow the approach of inserting clean activation into corrupted ones and measure the rescue utilized in Causal Tracing (Meng et al., 2022) and *inter-alia* as **(1)** It is often unstable and we observe instances where it generates huge negative value in our case and **(2)** it does not preserve the Gibb's inequality of KLD and thus, does not guarantee a positive utility. By repeatedly applying Eqs. (6) and (7) to one concept-module pair at a time, we obtain a Leave-One-Out (LOO) (Cook, 1977) intervention score relative to the clean model. However, such an intervention evaluates the marginal utility of each player only at the empty coalition and therefore ignores interactions among concept-module players.

A natural way to account for these coalition effects is through the Shapley value (see Def. C.1) (Shapley, 1953). However, it is well known that Shapley value is computationally expensive and in our case, requires $2^{|\mathcal{C}|L}$ evaluations, where L is the total number of modules probed, which is computationally intractable. Thus, some tricks are required.

Hierarchical utility allocation. Instead of solving for the utility of all elements of $\mathcal{N}$ directly, we will solve for the concept utility Φ_c and module utility corresponding to concept c, $\phi_{c,l}$ independently. First, by collapsing all module-level players associated with the same concept, this gives the quotient game

$$\bar{v}(A) \coloneqq v\left(\bigcup_{c\in A} \mathcal{G}_c\right), \qquad A \subseteq \mathcal{C}, \tag{24}$$

for which $\bar{v}(\mathcal{C}) = v(\mathcal{N})$. Let $\widetilde{\Phi}_c$ denote any raw estimate of the utility of concept c in $\bar{v}$ obtained from some estimator of Shapley value (more in the following section). Since an approximate estimator need not satisfy efficiency, we apply the global correction

$$\widehat{\Phi}_c \coloneqq v(\mathcal{N}) \frac{\widetilde{\Phi}_c}{\sum_{c'\in\mathcal{C}} \widetilde{\Phi}_{c'}}, \qquad \sum_{c\in\mathcal{C}} \widehat{\Phi}_c = v(\mathcal{N}), \tag{25}$$

assuming the denominator is non-zero. If the concept estimator is already efficient, this correction is the identity. We then estimate how the utility of concept c should be distributed across its modules. Let v_c denote the restriction of the original game to $\mathcal{G}_c$,

$$v_c(T) \coloneqq v(T) - v(\varnothing), \qquad T \subseteq \mathcal{G}_c, \tag{26}$$

and let $\widetilde{\phi}_{c,l}$ be any module-level utility estimate for player (c, l) in v_c and assume $v(\emptyset) = 0$ throughout. We use these estimates only to determine the relative allocation within concept c, defining

$$\widehat{\phi}_{c,l} \coloneqq \widehat{\Phi}_c \frac{\widetilde{\phi}_{c,l}}{\sum_{l'\in\mathcal{L}} \widetilde{\phi}_{c,l'}}, \tag{27}$$

again assuming a non-zero denominator. Consequently, $\sum_{l\in\mathcal{L}} \widehat{\phi}_{c,l} = \widehat{\Phi}_c$, and $\sum_{c\in\mathcal{C}} \sum_{l\in\mathcal{L}} \widehat{\phi}_{c,l} = v(\mathcal{N})$. This follows prior structure-aware Shapley decompositions (Chen et al., 2019; Tarun et al., 2024) and it is estimator-agnostic, as the concept- and module-level utilities may be computed using different estimators and computational budgets. Without additional assumptions, the hierarchical construction guarantees global efficiency and consistency with the chosen concept-level allocation. Under Assumption C.12, we show that theoretically the exact Shapley values at both levels recover the Shapley value of the original concept-module game exactly in Prop. C.13 and practically it allow a decent approximation of the exact Shapley value empirically in Appendix D.3.

C.2 Marginal-cancellation ratio

We introduce the *marginal-cancellation ratio* (MCR) to quantify how strongly coalition-level variation cancels in the paired marginal contributions used by permutation MC-Shapley.

Consider a cooperative game $(\mathcal{P}, \nu)$ where $\mathcal{P}$ is the set of players and $\nu : 2^{\mathcal{P}} \to \mathbb{R}$ is the characteristic function assigning a scalar utility $\nu(S)$ to each coalition $S \subseteq \mathcal{P}$. Let S_i denote the set of players preceding i in a uniformly random permutation, equivalently

$$K \sim \mathrm{Unif}\{0, \dots, n-1\}, \qquad S_i \mid K = k \sim \mathrm{Unif}\{A \subseteq \mathcal{P} \setminus \{i\} : |A| = k\}. \tag{28}$$

Then

$$\phi_i(\nu) = \mathbb{E}[\nu(S_i \cup \{i\}) - \nu(S_i)]\,. \tag{29}$$

Whenever the denominator is non-zero, define

$$\mathcal{R}_i(\nu) := \frac{\mathrm{Var}[\nu(S_i \cup \{i\})] + \mathrm{Var}[\nu(S_i)]}{\mathrm{Var}[\nu(S_i \cup \{i\}) - \nu(S_i)]}. \tag{30}$$

Using

$$\begin{aligned}\mathrm{Var}[\nu(S_i \cup \{i\}) - \nu(S_i)] = \mathrm{Var}[\nu(S_i \cup \{i\})] + \mathrm{Var}[\nu(S_i)] \\ - 2\,\mathrm{Cov}(\nu(S_i \cup \{i\}), \nu(S_i))\,,\end{aligned} \tag{31}$$

A large MCR therefore indicates that $\nu(S_i \cup \{i\})$ and $\nu(S_i)$ contain substantial shared coalition-dependent variation that cancels when forming the marginal contribution. Importantly, a large MCR should not be interpreted as greater variation of the marginal contribution itself. Rather, it indicates greater variance cancellation in the paired difference.

Empirically, this effect is much stronger in the module-level game. The concept- versus module-level MCR is 1.56 versus 1794.23 on CUB-200-2011 and 1.59 versus 1193.01 on CIFAR-100, respectively, motivating our use of MC-Shapley for module-level attribution.

C.3 LOO APPROXIMATION BOUND

After establishing how we distribute the utility across concepts then across modules, a natural question is: Under what conditions does LOO provide an accurate approximation?

We follow the concept-level game defined in Eq. (24). Since the characteristic function v measures the output-distribution change induced by an ablation, the induced concept-level game is itself an ablation-damage game.

$$\nu(A) \coloneqq \bar{v}(A) - \bar{v}(\varnothing) = v\left(\bigcup_{c\in A} \mathcal{G}_c\right). \tag{32}$$

Moreover, let $\boldsymbol{x} \in \mathbb{R}^d$ denote the clean activation and let $\boldsymbol{z}_c \in \mathbb{R}^d$ denote the activation component removed when concept c is ablated. We define the scalar ablation-damage functional $F : \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ such that

$$\nu(A) = F\left(\boldsymbol{x} - \sum_{c\in A} \boldsymbol{z}_c\right), \qquad A \subseteq \mathcal{C}. \tag{33}$$

Here, F represents the KL divergence between the clean output distribution and the output distribution produced by the intervened activation. In particular,

$$F(\boldsymbol{x}) = \nu(\varnothing) = 0. \tag{34}$$

Consequently, the clean-baseline LOO attribution of concept c is

$$L_c^{\text{clean}} \coloneqq \nu(\{c\}) = F(\boldsymbol{x} - \boldsymbol{z}_c). \tag{35}$$

Harsanyi dividend Interactions are often analyzed from an information-theoretic perspective through redundancy, uniqueness, and synergy (Liang et al., 2023). In order to characterize when LOO approximates Shapley well, we instead begin with the Harsanyi dividend (Harsanyi, 1963).

Definition C.2 (Harsanyi dividend)**.** *Let $\mathcal{C}$ be the set of concepts, and let $\nu : 2^{\mathcal{C}} \to \mathbb{R}$ be the concept-level ablation-damage game. For each coalition $T \subseteq \mathcal{C}$, the Harsanyi dividend of T is defined as*

$$m_\nu(T) \coloneqq \sum_{A\subseteq T} (-1)^{|T|-|A|} \nu(A). \tag{36}$$

Equivalently, $m_\nu(T)$ is the Möbius coefficient (Rota, 1964) of ν at T on the Boolean lattice $\left(2^{\mathcal{C}}, \subseteq\right)$. It represents the component of the coalition value $\nu(T)$ that cannot be attributed to any proper subcoalition of T.

Lemma C.3 (Shapley value as a distribution of Harsanyi dividends)**.** *Let $\nu : 2^{\mathcal{C}} \to \mathbb{R}$ be a concept-level ablation-damage game, and $m_\nu(T)$ denote the Harsanyi dividend of coalition $T \subseteq \mathcal{C}$. Then, for every concept $i \in \mathcal{C}$, its Shapley value satisfies*

$$\phi_i(\nu) = \sum_{T\subseteq\mathcal{C}\ i\in T} \frac{m_\nu(T)}{|T|}. \tag{37}$$

Proof. For each coalition $T \subseteq \mathcal{C}$, define the unanimity game $u_T : 2^{\mathcal{C}} \to \mathbb{R}$ by

$$u_T(A) \coloneqq \begin{cases} 1, & T \subseteq A, \\ 0, & T \nsubseteq A. \end{cases} \tag{38}$$

By Möbius inversion, the game ν admits the unique decomposition

$$\nu = \sum_{T\subseteq\mathcal{C}} m_\nu(T) u_T. \tag{39}$$

Indeed, for every coalition $A \subseteq \mathcal{C}$,

$$\sum_{T\subseteq\mathcal{C}} m_\nu(T) u_T(A) = \sum_{T\subseteq A} m_\nu(T) = \nu(A), \tag{40}$$

where the final equality follows from the Möbius reconstruction formula.

By the linearity of the Shapley value,

$$\phi_i(\nu) = \phi_i\left(\sum_{T\subseteq\mathcal{C}} m_\nu(T)u_T\right) = \sum_{T\subseteq\mathcal{C}} m_\nu(T)\phi_i(u_T). \tag{41}$$

It therefore remains to compute the Shapley value of a unanimity game.

If $i \notin T$, then adding i to any coalition does not affect whether that coalition contains T. Hence, i is a null player in u_T, and

$$\phi_i(u_T) = 0. \tag{42}$$

If $i \in T$, then only the players in T are essential to the unanimity game. Moreover, all players in T are symmetric, while every player in $\mathcal{C} \setminus T$ is a null player in the unanimity game u_T. By the efficiency property of the Shapley value,

$$\sum_{i\in\mathcal{C}} \phi_i(u_T) = u_T(\mathcal{C}) - u_T(\varnothing) = 1, \tag{43}$$

where the final equality follows because $T \subseteq \mathcal{C}$ and $T \neq \varnothing$. Since

$$\phi_i(u_T) = 0, \qquad i \in \mathcal{C} \setminus T, \tag{44}$$

the efficiency condition reduces to

$$\sum_{i\in T} \phi_i(u_T) = 1. \tag{45}$$

By symmetry, all players in T receive the same Shapley value. Therefore, for every $i \in T$,

$$|T|\phi_i(u_T) = 1, \tag{46}$$

Therefore,

$$\phi_i(u_T) = \begin{cases} \dfrac{1}{|T|}, & i \in T, \\ 0, & i \notin T. \end{cases} \tag{47}$$

Substituting Eq. (47) into Eq. (41) yields

$$\phi_i(\nu) = \sum_{T\subseteq\mathcal{C}} m_\nu(T)\phi_i(u_T) = \sum_{T\subseteq\mathcal{C}\ i\in T} \frac{m_\nu(T)}{|T|}, \tag{48}$$

which proves the result. □

Corollary 3.1 (Exact LOO–Shapley gap). *For every concept $c \in \mathcal{C}$,*

$$\phi_c(\nu) - L_c^{\text{clean}} = \sum_{\substack{T\subseteq\mathcal{C} \\ c\in T,\ |T|\geq 2}} \frac{m_\nu(T)}{|T|}. \tag{49}$$

Consequently,

$$\left|\phi_c(\nu) - L_c^{\text{clean}}\right| \leq \frac{1}{2}\sum_{\substack{c'\in\mathcal{C} \\ c'\neq c}} |m_\nu(\{c,c'\})| + \mathcal{R}_c^{(\geq 3)}, \tag{50}$$

where

$$\mathcal{R}_c^{(\geq 3)} := \sum_{\substack{T\subseteq\mathcal{C} \\ c\in T,\ |T|\geq 3}} \frac{|m_\nu(T)|}{|T|}. \tag{51}$$

Proof. By Lem. C.3,

$$\phi_c(\nu) = \sum_{\substack{T\subseteq\mathcal{C} \\ c\in T}} \frac{m_\nu(T)}{|T|}. \tag{52}$$

Because $\nu(\varnothing) = 0$, Möbius reconstruction on the singleton coalition gives

$$m_\nu(\{c\}) = \nu(\{c\}) = L_c^{\text{clean}}. \tag{53}$$

Separating the singleton term $T = \{c\}$ proves Eq. (49). Separating the pairwise coalitions from the coalitions of order at least three and applying the triangle inequality proves Eq. (50). □

Remark C.4 (Interpretation). *The positive empirical association between interaction mass and LOO error is predicted directly by Cor. 3.1. Pairwise interaction mass is the leading observable term in the error bound. Higher-order interactions and cancellation among signed dividends determine how tight the bound is.*

C.3.1 Exact pairwise interaction as mixed curvature

In order to write the pairwise Harsanyi dividend as a finite difference of the same function over a two-dimensional ablation rectangular region

$$\mathcal{B}_{cc'} = \{\boldsymbol{x} - s\boldsymbol{z}_c - t\boldsymbol{z}_{c'} \mid s, t \in [0,1]\}, \tag{54}$$

we will first make an assumption.

Assumption C.5 (Fixed additive intervention). *For every coalition $S \subseteq \mathcal{N}$, the intervened activation is*

$$\boldsymbol{x}_S^* = \boldsymbol{x} - \sum_{j \in S} \boldsymbol{z}_j, \tag{55}$$

where each intervention direction $\boldsymbol{z}_j$ is independent of the coalition in which it is used.

Now, with the assumption made, we can analytically define the pairwise interaction local-curvature.

Lemma C.6 (Exact pairwise mixed-curvature identity). *Suppose F is twice continuously differentiable on $\mathcal{B}_{cc'}$. Then*

$$m_\nu(\{c, c'\}) = \int_0^1 \int_0^1 \boldsymbol{z}_c^\top \nabla^2 F(\boldsymbol{x} - s\boldsymbol{z}_c - t\boldsymbol{z}_{c'}) \boldsymbol{z}_{c'} \, ds \, dt. \tag{56}$$

Proof. Let

$$g(s,t) = F(\boldsymbol{x} - s\boldsymbol{z}_c - t\boldsymbol{z}_{c'}). \tag{57}$$

By the definition of the Harsanyi dividend, Eq. (33), and Assumption C.5,

$$m_\nu(\{c, c'\}) = g(1,1) - g(1,0) - g(0,1) + g(0,0). \tag{58}$$

Applying the fundamental theorem of calculus in both coordinates gives

$$g(1,1) - g(1,0) - g(0,1) + g(0,0) = \int_0^1 \int_0^1 \frac{\partial^2 g}{\partial s \, \partial t}(s,t) \, ds \, dt. \tag{59}$$

The chain rule yields

$$\frac{\partial^2 g}{\partial s \, \partial t}(s,t) = \boldsymbol{z}_c^\top \nabla^2 F(\boldsymbol{x} - s\boldsymbol{z}_c - t\boldsymbol{z}_{c'}) \boldsymbol{z}_{c'}. \tag{60}$$

Substitution proves Eq. (56). □

C.3.2 Grassmannian and Harsanyi Interaction

The Grassmannian Following the notation of Lee et al. (2026), the Grassmannian or Grassmann manifold denoted as $\mathrm{Gr}(r,d)$ represents the set of all $r-$dimensional subspaces in $\mathbb{R}^d$,

$$\mathrm{Gr}(r,d) = \left\{ \boldsymbol{X} \in \mathbb{R}^{d \times r} \mid \boldsymbol{X}^\top \boldsymbol{X} = \boldsymbol{I}_r \right\} / O(r), \tag{61}$$

where $O(r)$ is the orthogonal group of $r \times r$ matrices.

At module $l \in \mathcal{L}$, let

$$\boldsymbol{Q}_c^{(l)} \in \mathbb{R}^{d_l \times r_c^{(l)}} \tag{62}$$

contain an orthonormal basis for the concept-c subspace. In ViTLens, $\boldsymbol{Q}_c^{(l)}$ may be identified with the retained left-singular-vector matrix $\boldsymbol{U}_c^{(l)}$.

For concepts $c \neq c'$, define

$$q_{cc'}^{(l)} = \min\{r_c^{(l)}, r_{c'}^{(l)}\}. \tag{63}$$

Let

$$\cos\theta_{cc',1}^{(l)}, \ldots, \cos\theta_{cc',q_{cc'}^{(l)}}^{(l)} \tag{64}$$

be the singular values of the projection matrix

$$\boldsymbol{P}_{cc'}^{(l)} = \boldsymbol{Q}_c^{(l)\top} \boldsymbol{Q}_{c'}^{(l)}, \tag{65}$$

where each principal angle satisfies

$$0 \le \theta_{cc',i}^{(l)} \le \frac{\pi}{2}. \tag{66}$$

In our case we fix $r = 8$ for all modules and concepts.

Definition C.7 (Normalized chordal distance)**.** *At module l, let $\theta_{cc',1}^{(l)}, \dots, \theta_{cc',r}^{(l)}$ denote the principal angles between the rank-r concept subspaces spanned by $\boldsymbol{U}_c^{(l)}$ and $\boldsymbol{U}_{c'}^{(l)}$. Following Ye & Lim (2016), we define their normalized chordal distance as*

$$\delta_{cc'}^{\mathrm{ch},(l)} := \left(\frac{1}{r} \sum_{j=1}^{r} \sin^2 \theta_{cc',j}^{(l)} \right)^{1/2} = \frac{1}{\sqrt{2r}} \left\| \boldsymbol{P}_c^{(l)} - \boldsymbol{P}_{c'}^{(l)} \right\|_F, \tag{67}$$

where $\boldsymbol{P}_c^{(l)} = \boldsymbol{U}_c^{(l)} \boldsymbol{U}_c^{(l)\top}$ and $\boldsymbol{P}_{c'}^{(l)} = \boldsymbol{U}_{c'}^{(l)} \boldsymbol{U}_{c'}^{(l)\top}$. The normalization by r ensures that $\delta_{cc'}^{\mathrm{ch},(l)} \in [0, 1]$.

Because the singular values of $\boldsymbol{U}_c^{(l)\top} \boldsymbol{U}_{c'}^{(l)}$ are $\cos \theta_{cc',1}^{(l)}, \dots, \cos \theta_{cc',r}^{(l)}$, we have

$$\left\| \boldsymbol{U}_c^{(l)\top} \boldsymbol{U}_{c'}^{(l)} \right\|_F^2 = \sum_{j=1}^{r} \cos^2 \theta_{cc',j}^{(l)} = r \left[1 - \left(\delta_{cc'}^{\mathrm{ch},(l)} \right)^2 \right]. \tag{68}$$

Thus, $1 - (\delta_{cc'}^{\mathrm{ch},(l)})^2$ is the average squared alignment between the two concept subspaces: larger chordal distance corresponds to smaller aggregate subspace alignment.

To connect this geometric quantity to the pairwise Harsanyi interaction, we next introduce an average-case model for the intervention directions within each concept subspace. Unless stated otherwise, all expectations and L^2 norms below are taken over $\boldsymbol{X} \sim \mathcal{D}_{\mathrm{eval}}^{c,c'}$.

Assumption C.8 (Isotropic intervention within each concept subspace)**.** *Let $\boldsymbol{X} \sim \mathcal{D}_{\mathrm{eval}}^{c,c'}$ denote an input drawn from the evaluation distribution used to measure the interaction between concepts c and c'. For each input $\boldsymbol{X}$, let $\boldsymbol{a}_c(\boldsymbol{X})$ and $\boldsymbol{a}_{c'}(\boldsymbol{X})$ denote the corresponding coordinates of the concept-intervention directions in their respective subspaces. At module l, the intervention directions admit the representations*

$$\boldsymbol{z}_c = \boldsymbol{U}_c^{(l)} \boldsymbol{a}_c, \qquad \boldsymbol{z}_{c'} = \boldsymbol{U}_{c'}^{(l)} \boldsymbol{a}_{c'}, \tag{69}$$

where, over the evaluation distribution,

$$\mathbb{E}[\boldsymbol{a}_c \boldsymbol{a}_c^\top] = \sigma_c^2 \boldsymbol{I}_r, \qquad \mathbb{E}[\boldsymbol{a}_{c'} \boldsymbol{a}_{c'}^\top] = \sigma_{c'}^2 \boldsymbol{I}_r, \tag{70}$$

and $\boldsymbol{a}_c, \boldsymbol{a}_{c'}$ are independent. $\sigma_c, \sigma_{c'}$ is the average intervention energy per retained basis direction for concept c, c' respectively.

This assumption states that, across the evaluation distribution, intervention energy is not concentrated in one preferred basis direction. It does not require the two subspaces to be orthogonal.

Assumption C.9 (Approximately isotropic downstream curvature)**.** *Let D denote the activation dimension. For every point $\boldsymbol{y} \in \mathcal{B}_{cc'}$, the Hessian of the scalar damage functional $F : \mathbb{R}^D \to \mathbb{R}$ admits the decomposition*

$$\nabla^2 F(\boldsymbol{y}) = \lambda(\boldsymbol{y}) \boldsymbol{I}_D + \boldsymbol{E}(\boldsymbol{y}), \tag{71}$$

where $\lambda(\boldsymbol{y}) \in \mathbb{R}$ is a scalar, and $\boldsymbol{E}(\boldsymbol{y}) \in \mathbb{R}^{D \times D}$ is the residual anisotropic curvature. We assume that there exist constants $\Lambda, \eta \ge 0$ such that

$$|\lambda(\boldsymbol{y})| \le \Lambda, \qquad \|\boldsymbol{E}(\boldsymbol{y})\|_2 \le \eta, \qquad \forall\, \boldsymbol{y} \in \mathcal{B}_{cc'}. \tag{72}$$

Here, $\|\cdot\|_2$ denotes the matrix spectral norm.

The isotropic component $\lambda(\boldsymbol{y})\boldsymbol{I}_D$ contributes $\boldsymbol{z}_c^\top \lambda(\boldsymbol{y})\boldsymbol{I}_D \boldsymbol{z}_{c'} = \lambda(\boldsymbol{y})\boldsymbol{z}_c^\top \boldsymbol{z}_{c'}$, and is therefore determined by the ordinary alignment between the two intervention directions. In contrast, the residual satisfies $\left|\boldsymbol{z}_c^\top \boldsymbol{E}(\boldsymbol{y})\boldsymbol{z}_{c'}\right| \le \eta \left\|\boldsymbol{z}_c\right\|_2 \left\|\boldsymbol{z}_{c'}\right\|_2$, and captures downstream coupling that is not explained by chordal separation. Thus, η controls the scale of the residual interaction that may remain even when the two concept subspaces are well separated.

Because the intervention directions vary across evaluation inputs, the resulting interaction is also treated as a random variable. For a scalar quantity $Y(\boldsymbol{X})$ defined for $\boldsymbol{X} \sim \mathcal{D}_{\text{eval}}^{c,c'}$, we define

$$\|Y\|_{L^2} \coloneqq \left(\mathbb{E}_{\boldsymbol{X}\sim\mathcal{D}_{\text{eval}}^{c,c'}}\left[Y(\boldsymbol{X})^2\right]\right)^{1/2}. \tag{73}$$

This is the root-mean-square magnitude of Y over the evaluation distribution.

Proposition C.10 (Chordal bound on pairwise Harsanyi interaction)**.** *Under Assumptions C.5, C.8 and C.9,*

$$\|m_\nu(\{c,c'\})\|_{L^2} \le \sigma_c\sigma_{c'}\left[\Lambda\sqrt{r\left(1-\left(\delta_{cc'}^{\text{ch},(l)}\right)^2\right)} + \eta r\right]. \tag{74}$$

Proof. For a fixed evaluation input $\boldsymbol{X}$, define

$$\boldsymbol{y}_{s,t}(\boldsymbol{X}) \coloneqq \boldsymbol{x}(\boldsymbol{X}) - s\boldsymbol{z}_c(\boldsymbol{X}) - t\boldsymbol{z}_{c'}(\boldsymbol{X}). \tag{75}$$

By Lem. C.6,

$$m_\nu(\{c,c'\};\boldsymbol{X}) = \int_0^1\int_0^1 \boldsymbol{z}_c(\boldsymbol{X})^\top \nabla^2 F\left(\boldsymbol{y}_{s,t}(\boldsymbol{X})\right)\boldsymbol{z}_{c'}(\boldsymbol{X})\,ds\,dt. \tag{76}$$

Using the decomposition from Assumption C.9,

$$\boldsymbol{z}_c^\top \nabla^2 F(\boldsymbol{y}_{s,t})\boldsymbol{z}_{c'} = \lambda(\boldsymbol{y}_{s,t})\boldsymbol{z}_c^\top \boldsymbol{z}_{c'} + \boldsymbol{z}_c^\top \boldsymbol{E}(\boldsymbol{y}_{s,t})\boldsymbol{z}_{c'}. \tag{77}$$

The bounds $|\lambda(\boldsymbol{y}_{s,t})| \le \Lambda$ and $\|\boldsymbol{E}(\boldsymbol{y}_{s,t})\|_2 \le \eta$ imply

$$\left|\boldsymbol{z}_c^\top \nabla^2 F(\boldsymbol{y}_{s,t})\boldsymbol{z}_{c'}\right| \le \Lambda\left|\boldsymbol{z}_c^\top \boldsymbol{z}_{c'}\right| + \eta\left\|\boldsymbol{z}_c\right\|_2\left\|\boldsymbol{z}_{c'}\right\|_2. \tag{78}$$

Substituting this inequality into Eq. (76) gives

$$|m_\nu(\{c,c'\};\boldsymbol{X})| \le \int_0^1\int_0^1 \left[\Lambda\left|\boldsymbol{z}_c^\top \boldsymbol{z}_{c'}\right| + \eta\left\|\boldsymbol{z}_c\right\|_2\left\|\boldsymbol{z}_{c'}\right\|_2\right] ds\,dt \tag{79}$$

$$= \Lambda\left|\boldsymbol{z}_c^\top \boldsymbol{z}_{c'}\right| + \eta\left\|\boldsymbol{z}_c\right\|_2\left\|\boldsymbol{z}_{c'}\right\|_2. \tag{80}$$

The final equality holds because, under the fixed additive intervention assumption, $\boldsymbol{z}_c$ and $\boldsymbol{z}_{c'}$ do not depend on s or t, and the area of $[0,1]^2$ is one.

Taking the L^2 norm of Eq. (80) and applying the triangle inequality yields

$$\|m_\nu(\{c,c'\})\|_{L^2} \le \Lambda\left\|\boldsymbol{z}_c^\top \boldsymbol{z}_{c'}\right\|_{L^2} + \eta\left\|\left\|\boldsymbol{z}_c\right\|_2\left\|\boldsymbol{z}_{c'}\right\|_2\right\|_{L^2}. \tag{81}$$

It remains to evaluate the two L^2 terms. Let $\boldsymbol{P}_{cc'}^{(l)} \coloneqq {\boldsymbol{U}_c^{(l)}}^\top \boldsymbol{U}_{c'}^{(l)}$. and by Eq. (69), $\boldsymbol{z}_c^\top \boldsymbol{z}_{c'} = \boldsymbol{a}_c^\top \boldsymbol{P}_{cc'}^{(l)} \boldsymbol{a}_{c'}$. Using the independence of $\boldsymbol{a}_c$ and $\boldsymbol{a}_{c'}$,

$$\mathbb{E}\left[\left(\boldsymbol{z}_c^\top \boldsymbol{z}_{c'}\right)^2\right] = \operatorname{tr}\left(\boldsymbol{P}_{cc'}^{(l)}\mathbb{E}[\boldsymbol{a}_{c'}\boldsymbol{a}_{c'}^\top]{\boldsymbol{P}_{cc'}^{(l)}}^\top \mathbb{E}[\boldsymbol{a}_c\boldsymbol{a}_c^\top]\right) = \sigma_c^2\sigma_{c'}^2\left\|\boldsymbol{P}_{cc'}^{(l)}\right\|_F^2$$

$$= r\sigma_c^2\sigma_{c'}^2\left[1-\left(\delta_{cc'}^{\text{ch},(l)}\right)^2\right], \tag{82}$$

where the final equality follows from Eq. (68). Consequently,

$$\left\|\boldsymbol{z}_c^\top \boldsymbol{z}_{c'}\right\|_{L^2} = \sigma_c\sigma_{c'}\sqrt{r\left[1-\left(\delta_{cc'}^{\text{ch},(l)}\right)^2\right]}. \tag{83}$$

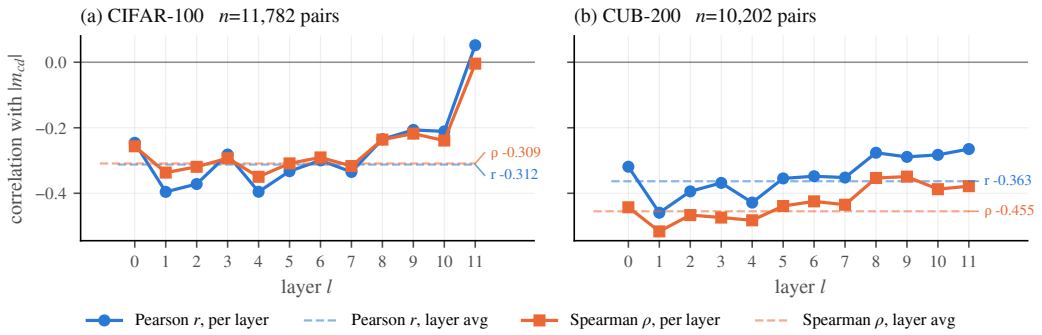


Figure C.1: Per-layer and average Pearson and Spearman correlation of chordal distance with respect to the pairwise interaction mass.

Because $\boldsymbol{U}_c^{(l)}$ and $\boldsymbol{U}_{c'}^{(l)}$ have orthonormal columns,

$$\|\boldsymbol{z}_c\|_2 = \|\boldsymbol{a}_c\|_2\,, \qquad \|\boldsymbol{z}_{c'}\|_2 = \|\boldsymbol{a}_{c'}\|_2\,. \tag{84}$$

Moreover,

$$\mathbb{E}\left[\|\boldsymbol{z}_c\|_2^2\right] = \operatorname{tr}\left(\mathbb{E}[\boldsymbol{a}_c\boldsymbol{a}_c^\top]\right) = r\sigma_c^2, \quad \mathbb{E}\left[\|\boldsymbol{z}_{c'}\|_2^2\right] = r\sigma_{c'}^2. \tag{85}$$

Independence therefore gives

$$\left\|\|\boldsymbol{z}_c\|_2\,\|\boldsymbol{z}_{c'}\|_2\right\|_{L^2}^2 = \mathbb{E}\left[\|\boldsymbol{z}_c\|_2^2\,\|\boldsymbol{z}_{c'}\|_2^2\right] = r^2\sigma_c^2\sigma_{c'}^2. \tag{86}$$

Substituting Eqs. (83) and (86) into Eq. (81) gives

$$\|m_\nu(\{c,c'\})\|_{L^2} \le \sigma_c\sigma_{c'}\left[\Lambda\sqrt{r\left(1-\left(\delta_{cc'}^{\mathrm{ch},(l)}\right)^2\right)} + \eta r\right].$$

□

Combining Prop. C.10 with the exact LOO-Shapley decomposition yields a bound on the approximation error in terms of pairwise chordal separation, intervention scale, downstream curvature, and the higher-order Harsanyi remainder.

Remark C.11 (Interpretation and layer averaging)**.** *For fixed intervention scales and controlled higher-order interactions, Eq.* (74) *decreases as* $\delta_{cc'}^{\mathrm{ch},(l)}$ *increases. This predicts a negative association between chordal distance and LOO approximation error. The conclusion is conditional: a large anisotropic curvature term* η *or a large higher-order remainder can weaken the relationship.*

The theorem is stated at one module. In the experiments, the layer-averaged summary

$$\overline{\delta}_{cc'}^{\mathrm{ch}} := \frac{1}{|\mathcal{L}|}\sum_{l\in\mathcal{L}}\delta_{cc'}^{\mathrm{ch},(l)} \tag{87}$$

is used as an empirical measure of overall concept separation. This average summarizes the per-module geometry; it is not itself a single Grassmann distance unless an additional global construction across modules is imposed.

C.4 Exactness of Hierarchical Shapley Allocation

Unlike the concept- and module-level game, evaluating the overall Shapley value (Shapley, 1953) is computationally intractable. For example, for $|\mathcal{C}| = 5, |\mathcal{L}| = 12$ in ViT-B/16, it will take $2^{60} \approx 1.15 \times 10^{18}$ iterations per image. We thus provide an analytical explanation on why such formulation could be approximately faithful.

Recall the concept-level quotient game in Eq. (24),

$$\bar{v}(A) \coloneqq v\left(\bigcup_{c \in A} \mathcal{G}_c\right), \qquad A \subseteq \mathcal{C}, \tag{88}$$

and the module-level game in Eq. (26), which is a concept-restricted game, both with $v(\varnothing) = 0$ in the ablation-damage game setting.

Let

$$\Phi_c := \phi_c(\bar{v}) \tag{89}$$

denote the exact Shapley value of concept c in the quotient game, and let

$$\psi_{c,l} := \phi_{(c,l)}(v_c) \tag{90}$$

denote the exact Shapley value of module l in the restricted game for concept c. The corresponding exact version of our hierarchical allocation is

$$\widehat{\phi}^{\text{hier}}_{c,l} = \Phi_c \frac{\psi_{c,l}}{\sum_{l' \in \mathcal{L}} \psi_{c,l'}}. \tag{91}$$

We now characterize a sufficient condition under which Eq. (91) coincides with the Shapley value of the original game.

Assumption C.12 (Concept separability)**.** *The concept–module characteristic function is additively separable across concept groups. That is, for every $S \subseteq \mathcal{N}$,*

$$v(S) = \sum_{c \in \mathcal{C}} v(S \cap G_c). \tag{92}$$

Equivalently, there are no irreducible cooperative interactions spanning multiple concept groups.

Assumption C.12 does not require additivity among modules within the same concept; arbitrary intra-concept interactions are permitted. It only excludes cross-concept interactions.

While Assumption C.12 is an idealized sufficient condition, our results support its approximate validity. At the concept level, LOO achieves around 0.8 Pearson and Spearman correlation with exact Shapley in Fig. 2, while Cor. 3.1 shows that the LOO-Shapley discrepancy is exactly determined by non-singleton Harsanyi interactions. Moreover, Prop. C.10 bounds the leading pairwise interaction through concept-subspace geometry, consistent with the negative association between chordal distance and pairwise interaction magnitude observed in Fig. C.1. Together, these results suggest that cross-concept interactions are sufficiently limited for concept separability to provide a useful approximation.

Proposition C.13 (Exactness of hierarchical Shapley)**.** *Suppose Assumption C.12 holds and the concept-level and within-concept quantities in Eqs.* (89) *and* (90) *are exact Shapley values. Then, for every $(c, l) \in \mathcal{N}$ for which the denominator of Eq.* (91) *is nonzero,*

$$\widehat{\phi}^{\text{hier}}_{c,l} = \phi_{(c,l)}(v). \tag{93}$$

Hence, under concept separability, exact Shapley attribution at both levels of the hierarchy exactly recovers the Shapley value of the original concept–module game.

Proof. For each concept c, define a game on the full player set by

$$u_c(S) := v(S \cap G_c). \tag{94}$$

By Assumption C.12,

$$v(S) = \sum_{c \in \mathcal{C}} u_c(S), \tag{95}$$

or equivalently,

$$v = \sum_{c \in \mathcal{C}} u_c. \tag{96}$$

Fix a player $i = (c, l) \in G_c$. By linearity of the Shapley value,

$$\phi_i(v) = \sum_{c' \in \mathcal{C}} \phi_i(u_{c'}). \tag{97}$$

For every $c' \neq c$, the game $u_{c'}$ depends only on players belonging to $G_{c'}$. Since $i \notin G_{c'}$, player i is a null player in $u_{c'}$, and therefore

$$\phi_i(u_{c'}) = 0, \qquad c' \neq c. \tag{98}$$

Consequently,

$$\phi_i(v) = \phi_i(u_c). \tag{99}$$

On coalitions $T \subseteq G_c$, we have

$$u_c(T) = v(T) = v_c(T),$$

where the last equality follows from $v(\varnothing) = 0$. Therefore,

$$\phi_i(v) = \phi_i(v_c) = \psi_{c,l}. \tag{100}$$

It remains to evaluate the concept-level Shapley value Φ_c. For any $A \subseteq \mathcal{C}$, concept separability gives

$$\bar{v}(A) = v\left(\bigcup_{c' \in A} G_{c'}\right) = \sum_{c' \in \mathcal{C}} v\left(G_{c'} \cap \bigcup_{d \in A} G_d\right) = \sum_{c' \in A} v(G_{c'}). \tag{101}$$

Hence, the quotient game is additive over concepts. In particular, for every $A \subseteq \mathcal{C} \setminus \{c\}$,

$$\bar{v}(A \cup \{c\}) - \bar{v}(A) = v(G_c). \tag{102}$$

Since every Shapley marginal contribution of concept c is identical,

$$\Phi_c = \phi_c(\bar{v}) = v(G_c). \tag{103}$$

Moreover, by efficiency of the Shapley value in the restricted game,

$$\sum_{l' \in \mathcal{L}} \psi_{c,l'} = v_c(G_c) - v_c(\varnothing) = v(G_c). \tag{104}$$

Substituting Eqs. (103) and (104) into Eq. (91) gives

$$\widehat{\phi}^{\text{hier}}_{c,l} = v(G_c)\frac{\psi_{c,l}}{v(G_c)} = \psi_{c,l} = \phi_{(c,l)}(v),$$

where the final equality follows from Eq. (100). This proves the result. □

Exploiting structural or hierarchical organization to reduce the cost of Shapley valuation has precedent in prior work. Chen et al. (2019) exploit graphical structure to restrict Shapley computation to relevant feature neighborhoods which is related to the Myerson value (Myerson, 1977), while Tarun et al. (2024) employ a two-stage valuation scheme that first values groups and subsequently allocates utility to their individual members. These works establish hierarchical decomposition as a principled approach to reducing the computational cost of Shapley value, rather than requiring the full game to be evaluated directly. Our result follows this line of work while providing an explicit sufficient condition in Prop. C.13 under which the proposed two-level construction recovers the Shapley value of the original game exactly.

D A BETTER SHAPLEY APPROXIMATOR

The preceding analysis provides a general rule of thumb for deciding when LOO is expected to approximate the Shapley value well. Nevertheless, in settings where a modest amount of additional computation is available, it is desirable to refine the inexpensive LOO estimate rather than discard it and restart a Monte-Carlo estimator from an uninformative initialization.

A standard permutation Monte-Carlo estimator is asymptotically consistent, but its variance can dominate the deterministic LOO error under a small evaluation budget. Consequently, a low-budget Monte-Carlo estimate is not guaranteed to be more accurate than LOO. We therefore integrate LOO evaluations into Stratified SVARM Kolpaczki et al. (2024). The resulting implementation, SVARM-LOO, reuses the singleton evaluations and estimates the remaining strata through additional coalition sampling.

D.1 REVIEW OF SVARM

Let C denote the player set and let $n = |C|$. For a fixed player $c \in C$, write $C_{-c} = C \setminus \{c\}$. Applying the distributive law, the Shapley value decomposes as

$$\phi_c(\nu) = \phi_c^+(\nu) - \phi_c^-(\nu), \qquad \phi_c^+(\nu) = \sum_{S \subseteq C_{-c}} w_S \nu(S \cup \{c\}), \qquad \phi_c^-(\nu) = \sum_{S \subseteq C_{-c}} w_S \nu(S), \tag{105}$$

where

$$w_S = \frac{1}{n\binom{n-1}{|S|}}. \tag{106}$$

Thus ϕ_c^+ and ϕ_c^- are expectations of coalition values rather than expectations of explicitly evaluated marginal contributions. One evaluation of $\nu(A)$ can therefore update the positive estimate of every $c \in A$ and the negative estimate of every $c \notin A$ simultaneously.

Size-stratified form. Because w_S depends on S only through $|S|$, the two signed means in Eq. (105) can be grouped by coalition size. Define

$$\mu_{c,k}^+ = \frac{1}{\binom{n-1}{k-1}} \sum_{\substack{S \subseteq C,\ |S|=k \\ c \in S}} \nu(S),$$
$$\mu_{c,k}^- = \frac{1}{\binom{n-1}{k}} \sum_{\substack{S \subseteq C,\ |S|=k \\ c \notin S}} \nu(S). \tag{107}$$

Grouping Eq. (105) by coalition size gives the exact identity

$$\phi_c(\nu) = \frac{1}{n}\left[\sum_{k=1}^{n} \mu_{c,k}^+ - \sum_{k=0}^{n-1} \mu_{c,k}^-\right]. \tag{108}$$

In this form, one evaluation of $\nu(A)$ updates the positive size-$|A|$ stratum of every $c \in A$ and the negative size-$|A|$ stratum of every $c \notin A$ simultaneously. The boundary strata satisfy $\mu_{c,0}^- = \nu(\emptyset) = 0$ and $\mu_{c,n}^+ = \nu(C)$.

Our Stratified SVARM baseline follows Kolpaczki et al. (2024), where it evaluates all coalitions of sizes 1, $n-1$, and n, then warms up every remaining player-specific positive and negative stratum. For $n \geq 4$, this requires $W = 2n + 1 + 2\sum_{s=2}^{n-2} \lceil n/s \rceil$ value-function accesses before memoization. Subsequent draws use the published symmetric coalition-size distribution. For $n \leq 5$, our implementation instead uses a uniform distribution over the interior sizes because the published distribution degenerates at $n = 4$. Our SVARM-LOO variant, described below, reuses the singleton evaluations and samples the remaining sizes without this full warm-up.

D.2 SVARM-LOO: LEAVE-ONE-OUT SCORES ARE EXACT STRATA

Our implementation evaluates every singleton $\nu(\{c\})$ as well as the grand coalition $\nu(C)$ during initialization. The singleton evaluations determine the size-one signed strata exactly, while the grand-coalition evaluation determines the positive size-n stratum.

Proposition D.1 (LOO scores as exact SVARM strata)**.** *Let*

$$b_c := \nu(\{c\}) \tag{109}$$

denote the clean-baseline leave-one-out score. Then, for every $c \in C$,

$$\mu^+_{c,1} = b_c, \qquad \mu^-_{c,1} = \frac{1}{n-1}\sum_{c' \neq c} b_{c'}. \tag{110}$$

Consequently,

$$\phi_c(\nu) = \frac{1}{n}\left[b_c - \frac{1}{n-1}\sum_{c' \neq c} b_{c'}\right] + \frac{\nu(C)}{n} + \frac{1}{n}\sum_{k=2}^{n-1}\left(\mu^+_{c,k} - \mu^-_{c,k}\right). \tag{111}$$

Proof. There is exactly one size-one coalition containing c, namely $\{c\}$, so $\mu^+_{c,1} = \nu(\{c\}) = b_c$. There are exactly $n-1$ size-one coalitions that do not contain c, namely $\{c'\}$ for $c' \neq c$, hence $\mu^-_{c,1} = (n-1)^{-1}\sum_{c' \neq c} b_{c'}$. Substituting these identities, together with $\mu^-_{c,0} = 0$ and $\mu^+_{c,n} = \nu(C)$, into Eq. (108) leaves exactly the indices $k = 2, \ldots, n-1$ in the unknown part and proves Eq. (111). □

Prop. D.1 shows how the singleton evaluations computed by LOO enter the stratified estimator directly. The LOO scores occur directly inside an exact Shapley identity in centered form, with no additional interpolation parameter. Only the intermediate size strata remain to be estimated. In an additive game, Eq. (111) reduces exactly to $\phi_c = b_c$, recovering the consistency of LOO.

The estimator. Let T denote the number of *additional* coalition draws after the singleton and grand-coalition values have been cached. We spread these draws approximately uniformly over the interior sizes $k = 2, \ldots, n-1$. Each sampled coalition updates the positive stratum of every member and the negative stratum of every non-member at its coalition size. We form the corresponding sampled stratum means $\widehat{\mu}^{\pm}_{c,k}$ and substitute them into Eq. (111):

$$\widehat{\phi}_c^{\mathrm{SVARM-LOO}} = \frac{1}{n}\left[b_c - \frac{1}{n-1}\sum_{c' \neq c} b_{c'} + \nu(C) + \sum_{k=2}^{n-1}\left(\widehat{\mu}^+_{c,k} - \widehat{\mu}^-_{c,k}\right)\right]. \tag{112}$$

When a player-specific positive or negative stratum receives no sample, our implementation imputes it using the overall sampled mean at the same coalition size. This avoids requiring the full warm-up of Stratified SVARM.

The total evaluation cost is $n+1$ plus the number of distinct sampled coalitions not already cached: the n singleton evaluations and the grand-coalition evaluation are computed during initialization and reused by SVARM-LOO. With T additional coalition draws, the total cost is therefore at most $n+1+T$ distinct coalition evaluations.

Shapley correlation experiment details. We evaluate 400 test images per dataset using FFN, Attn, and Res activations across all 12 Transformer layers. The candidate concept banks contain 608 concepts for CUB-200-2011 and 698 for CIFAR-100. For the concept-level benchmark, we compute exact Shapley values by enumerating all 2^{K_I} coalitions of the shortlisted concepts plus REST. Concepts are shortlisted based on ViTLens's energy in Eq. (5), where only concepts with energy greater than $\tau_g = 2$ are considered. To ensure computational feasibility, we limit them at $K_I \in [4, 8]$. For the module-level game, exact Shapley values are computed over all $2^{12} = 4096$ layer coalitions. The coalition values are cached during exact enumeration, and the estimator benchmark replays sampling procedures against these cached tables. Thus, the replay benchmark requires no new model forward passes. The reported evaluation counts measure the number of distinct coalition values queried by each estimator.

D.3 HIERARCHICAL GAME VALIDATION

Table D.1: Pooled correlation results across datasets.

Dataset	Pearson	Spearman
CUB-200-2011	0.881	0.745
CIFAR-100	0.857	0.666

To empirically verify our hierarchical game framework and the theoretical derivation in Prop. C.13, we have set up a small scale experiment. Full-scale experiment is not possible as for $|\mathcal{C}| = 5, |\mathcal{L}| = 12$ in ViT-B/16, we require around $2^{60} \approx 1.15 \times 10^{18}$ iterations per image.

Instead, we restrict to the last 3 layers of ViT-B and limit the maximum concepts to 5, while keeping the REST concept player to absorb the remaining subspaces. Similar to Fig. 2, we evaluated over 400 images from the test set of CIFAR-100 and CUB-200-2011 with SVARM-LOO for concept-level game and MC-Shapley for module level game at $1.92\times$ and $1.93\times$ the budget of LOO. As shown in Tab. D.1 after pooling across Res , FFN and Attn, we achieved up to 0.881 and 0.745 Pearson and Spearman correlation. These results provide empirical evidence that the hierarchical approximation preserves much of the joint Shapley structure in tractable games where exact enumeration is possible.

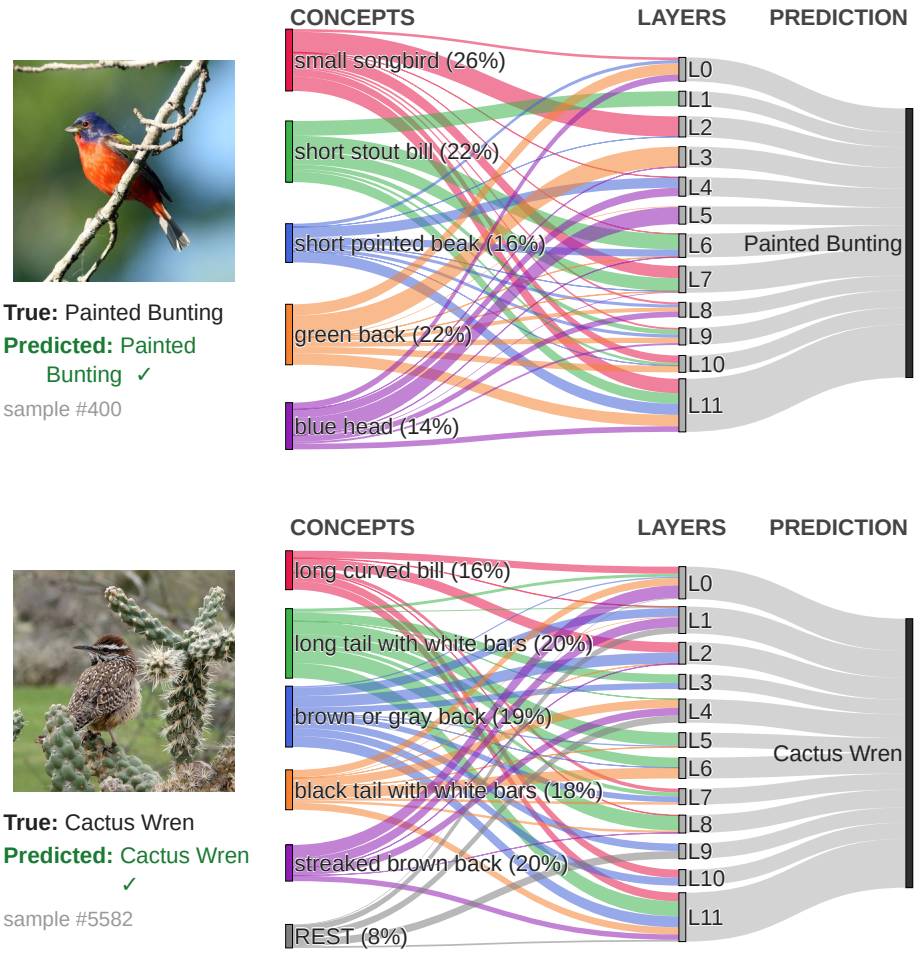


Figure D.1: Sankey diagrams for concept and layer utility computed with SVARM-LOO and MC-Shapley, respectively for CUB-200-2011 dataset.

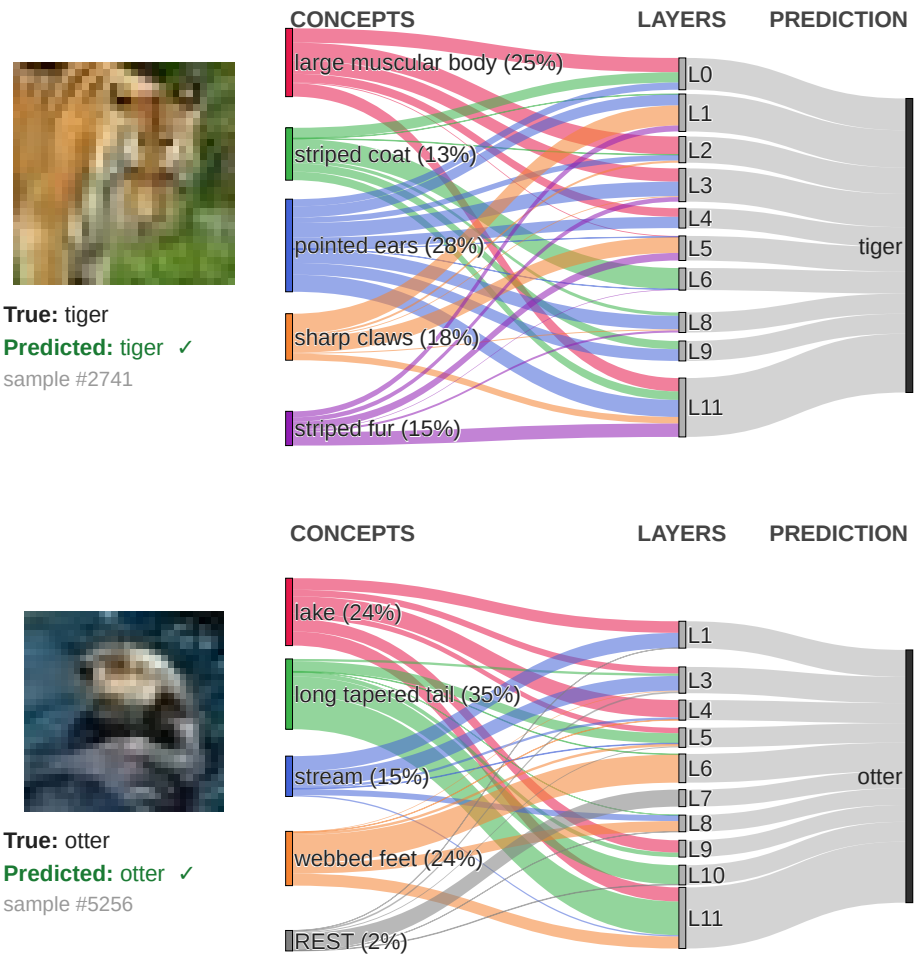


Figure D.2: Sankey diagrams for concept and layer utility computed with SVARM-LOO and MC-Shapley, respectively for CIFAR-100 dataset.

E CONCEPT SUPPRESSION

In order to achieve disentangled selective suppression, instead of blindly removing all components lying in the subspaces of concepts to be suppressed, we identify directions that maximally capture the suppressed concepts relative to the retained concepts through a generalized Rayleigh problem, similar to Qian et al. (2026).

Let $\mathcal{C}_s$ be the set of concepts belonging to class y that are to be suppressed. For every test image I_{test}, ViTLens first detects the set of concepts present in the image, denoted by $\mathcal{C}(I_{\text{test}})$. We then partition the detected concepts into two disjoint sets: the concepts to be suppressed,

$$\mathcal{C}_{\text{sup}}(I_{\text{test}}) = \mathcal{C}(I_{\text{test}}) \cap \mathcal{C}_s, \tag{113}$$

and the concepts to be retained,

$$\mathcal{C}_{\text{ret}}(I_{\text{test}}) = \mathcal{C}(I_{\text{test}}) \setminus \mathcal{C}_s. \tag{114}$$

For each module l, let $\mathcal{C}_{\text{sup}}$ and $\mathcal{C}_{\text{ret}}$ denote the detected concepts to be suppressed and retained, respectively. We construct

$$\mathbf{A}_{\text{sup}}^{(l)} = \sum_{c \in \mathcal{C}_{\text{sup}}} \mathbf{P}_c^{(l)}, \qquad \mathbf{A}_{\text{ret}}^{(l)} = \sum_{c \in \mathcal{C}_{\text{ret}}} \mathbf{P}_c^{(l)} + \epsilon \mathbf{I}, \tag{115}$$

where $\mathbf{P}_c^{(l)} = \boldsymbol{U}_c^{(l)} \boldsymbol{U}_c^{(l)\top}$ is the concept projector defined in Eq. (3), and $\epsilon > 0$ is a ridge term that makes $\mathbf{A}_{\text{ret}}^{(l)}$ positive definite.

To minimize interference with retained concepts, we seek a direction $\mathbf{u} \neq \mathbf{0}$ that has large energy under $\mathbf{A}_{\text{sup}}^{(l)}$ while having small energy under $\mathbf{A}_{\text{ret}}^{(l)}$. This is quantified by the generalized Rayleigh quotient

$$\mathcal{R}^{(l)}(\mathbf{u}) = \frac{\mathbf{u}^\top \mathbf{A}_{\text{sup}}^{(l)} \mathbf{u}}{\mathbf{u}^\top \mathbf{A}_{\text{ret}}^{(l)} \mathbf{u}}. \tag{116}$$

Since

$$\mathbf{u}^\top \mathbf{A}_{\text{sup}}^{(l)} \mathbf{u} = \sum_{c \in \mathcal{C}_{\text{sup}}} \left\| \boldsymbol{U}_c^{(l)\top} \mathbf{u} \right\|_2^2, \tag{117}$$

the numerator measures the amount of suppressed-concept energy captured by $\mathbf{u}$. Likewise,

$$\mathbf{u}^\top \mathbf{A}_{\text{ret}}^{(l)} \mathbf{u} = \sum_{c \in \mathcal{C}_{\text{ret}}} \left\| \boldsymbol{U}_c^{(l)\top} \mathbf{u} \right\|_2^2 + \epsilon \|\mathbf{u}\|_2^2, \tag{118}$$

so the denominator penalizes directions that overlap with retained concepts, while the ridge term regularizes directions outside their span.

Maximizing Eq. (116) is equivalent to solving

$$\max_{\mathbf{u}} \quad \mathbf{u}^\top \mathbf{A}_{\text{sup}}^{(l)} \mathbf{u} \qquad \text{s.t.} \qquad \mathbf{u}^\top \mathbf{A}_{\text{ret}}^{(l)} \mathbf{u} = 1. \tag{119}$$

Introducing a Lagrange multiplier λ gives

$$\mathcal{L}(\mathbf{u}, \lambda) = \mathbf{u}^\top \mathbf{A}_{\text{sup}}^{(l)} \mathbf{u} - \lambda \left(\mathbf{u}^\top \mathbf{A}_{\text{ret}}^{(l)} \mathbf{u} - 1 \right). \tag{120}$$

Setting the derivative with respect to $\mathbf{u}$ to zero yields

$$\mathbf{A}_{\text{sup}}^{(l)} \mathbf{u} = \lambda \mathbf{A}_{\text{ret}}^{(l)} \mathbf{u}. \tag{121}$$

Thus, the generalized eigenvalue λ directly measures the ratio of suppressed-concept energy to retained-concept energy along the corresponding direction. Large values of λ indicate directions that are highly specific to the concepts being suppressed, whereas small values indicate directions that are strongly shared with retained concepts.

For numerical stability, we solve Eq. (121) by whitening the retained-concept matrix. Since $\mathbf{A}_{\text{ret}}^{(l)} \succ 0$, let

$$\mathbf{A}_{\text{ret}}^{(l)} = \mathbf{L}^{(l)} \mathbf{L}^{(l)\top} \tag{122}$$

be its Cholesky factorization. Defining

$$\mathbf{q} = \mathbf{L}^{(l)\top}\mathbf{u}, \qquad \mathbf{u} = \mathbf{L}^{(l)-\top}\mathbf{q}, \tag{123}$$

and substituting into Eq. (121) gives

$$\mathbf{L}^{(l)-1}\mathbf{A}_{\text{sup}}^{(l)}\mathbf{L}^{(l)-\top}\mathbf{q} = \widetilde{\mathbf{A}}_{\text{sup}}^{(l)}\mathbf{q} = \lambda\mathbf{q}. \tag{124}$$

The whitened matrix $\widetilde{\mathbf{A}}_{\text{sup}}^{(l)}$ is symmetric positive semidefinite, so it admits the eigendecomposition

$$\widetilde{\mathbf{A}}_{\text{sup}}^{(l)} = \mathbf{Q}^{(l)}\mathbf{\Lambda}^{(l)}\mathbf{Q}^{(l)\top}, \qquad \mathbf{\Lambda}^{(l)} = \text{diag}(\lambda_1, \ldots, \lambda_m). \tag{125}$$

The corresponding generalized eigenvectors in the original activation space are therefore

$$\boldsymbol{D}^{(l)} = \mathbf{L}^{(l)-\top}\mathbf{Q}^{(l)}. \tag{126}$$

They satisfy

$$\boldsymbol{D}^{(l)\top}\mathbf{A}_{\text{ret}}^{(l)}\boldsymbol{D}^{(l)} = \mathbf{I}, \qquad \boldsymbol{D}^{(l)\top}\mathbf{A}_{\text{sup}}^{(l)}\boldsymbol{D}^{(l)} = \mathbf{\Lambda}^{(l)}. \tag{127}$$

Note that the generalized eigenvectors are orthonormal with respect to the $\mathbf{A}_{\text{ret}}^{(l)}$ metric, while $\mathbf{A}_{\text{sup}}^{(l)}$ is diagonalized in this basis with eigenvalues $\mathbf{\Lambda}^{(l)}$; in general, $\boldsymbol{D}^{(l)\top}\boldsymbol{D}^{(l)} \neq \mathbf{I}$.

In practice, both $\mathbf{A}_{\text{sup}}^{(l)}$ and $\mathbf{A}_{\text{ret}}^{(l)} - \epsilon\mathbf{I}$ are low-rank because they are sums of concept projectors. We therefore solve the generalized eigenproblem only inside the joint concept span

$$\mathcal{S}^{(l)} = \text{range}\left(\left[\boldsymbol{U}_c^{(l)}\right]_{c\in\mathcal{C}_{\text{sup}}\cup\mathcal{C}_{\text{ret}}}\right). \tag{128}$$

Let $\mathbf{F}^{(l)}$ be an orthonormal basis of this span, obtained by an SVD of the concatenated concept bases. We form the reduced matrices

$$\widehat{\mathbf{A}}_{\text{sup}}^{(l)} = \mathbf{F}^{(l)\top}\mathbf{A}_{\text{sup}}^{(l)}\mathbf{F}^{(l)}, \qquad \widehat{\mathbf{A}}_{\text{ret}}^{(l)} = \mathbf{F}^{(l)\top}\mathbf{A}_{\text{ret}}^{(l)}\mathbf{F}^{(l)}, \tag{129}$$

solve the whitened eigenproblem in this reduced space, and map the resulting generalized eigenvectors back through $\mathbf{F}^{(l)}$. This yields the same nontrivial suppression directions while avoiding a full $d_l \times d_l$ generalized eigendecomposition.

Finally, we convert each generalized eigenvalue into a bounded suppression weight

$$\eta_i = \min(1, \lambda_i), \tag{130}$$

and define

$$\mathbf{M}^{(l)} = \boldsymbol{D}^{(l)}\,\text{diag}\left(\sqrt{\eta_1}, \ldots, \sqrt{\eta_m}\right). \tag{131}$$

The resulting positive-semidefinite suppression operator is

$$\mathbf{M}^{(l)}\mathbf{M}^{(l)\top} = \boldsymbol{D}^{(l)}\,\text{diag}(\eta_1, \ldots, \eta_m)\boldsymbol{D}^{(l)\top}. \tag{132}$$

For a column-vector activation $\mathbf{x}^{(l)}$, suppression is performed as

$$\mathbf{x}^{(l)*} = \mathbf{x}^{(l)} - \mathbf{M}^{(l)}\mathbf{M}^{(l)\top}\mathbf{x}^{(l)}. \tag{133}$$

Although $\boldsymbol{D}^{(l)}$ is not generally Euclidean-orthonormal, the magnitude of the implemented update is controlled by the ridge term. Since $\mathbf{A}_{\text{ret}}^{(l)} \succeq \epsilon\mathbf{I}$ and $\boldsymbol{D}^{(l)\top}\mathbf{A}_{\text{ret}}^{(l)}\boldsymbol{D}^{(l)} = \mathbf{I}$, we have

$$\epsilon\,\boldsymbol{D}^{(l)\top}\boldsymbol{D}^{(l)} \preceq \mathbf{I}. \tag{134}$$

Because $0 \leq \eta_i \leq 1$, it follows that

$$0 \preceq \mathbf{M}^{(l)}\mathbf{M}^{(l)\top} \preceq \frac{1}{\epsilon}\mathbf{I}. \tag{135}$$

We use $\epsilon = 1$ in all reported experiments; hence

$$0 \preceq \mathbf{M}^{(l)}\mathbf{M}^{(l)\top} \preceq \mathbf{I}, \tag{136}$$

so $\mathbf{I} - \mathbf{M}^{(l)}\mathbf{M}^{(l)\top}$ is positive semidefinite and non-expansive. Thus, the activation amplification or sign inversion that can arise for smaller ridge values does not occur under our experimental setting.

This construction interpolates between weaker and stronger demotion according to the relative evidence carried by each direction. A direction that strongly overlaps with retained concepts has a larger denominator in Eq. (116), resulting in a smaller λ_i and hence weaker suppression. Conversely, a direction that is predominantly associated with the suppressed concepts has a large generalized eigenvalue and reaches the maximum suppression weight once $\lambda_i \geq 1$.

E.1 Concept Suppression Metrics

Our suppression setting is analogous to class unlearning in that the objective is to remove information associated with a designated class while preserving performance on the remaining classes. We therefore adapt the evaluation metrics of Chundawat et al. (2023); Fan et al. (2024); Wu & Harandi (2024). Importantly, our evaluation differs from the standard unlearning setting. This is where *all* metrics below are computed exclusively on the test set. In particular, our suppressed accuracy (**SA**) and retained-class metrics do not use the training forget/retain sets used to compute UA and RA in class unlearning.

Let $\mathcal{Y}$ denote the set of classes, and let $y \in \mathcal{Y}$ be the class being suppressed, with associated concept set $\mathcal{C}_s$ as defined above. For every other class $y' \in \mathcal{Y} \setminus \{y\}$, let $\mathcal{C}_{y'}$ denote its associated concept set. We partition the non-suppressed classes according to whether they share at least one concept with $\mathcal{C}_s$:

$$\mathcal{Y}_{\mathrm{O}}(y) = \{y' \in \mathcal{Y} \setminus \{y\} : \mathcal{C}_{y'} \cap \mathcal{C}_s \neq \varnothing\}, \tag{137}$$

$$\mathcal{Y}_{\mathrm{NO}}(y) = \{y' \in \mathcal{Y} \setminus \{y\} : \mathcal{C}_{y'} \cap \mathcal{C}_s = \varnothing\}. \tag{138}$$

Let $\mathcal{D}_{\mathrm{test}}$ denote the test set and define the test examples belonging to class y' as

$$\mathcal{D}_{y'} = \{(I_{\mathrm{test}}, t) \in \mathcal{D}_{\mathrm{test}} : t = y'\}. \tag{139}$$

The test subsets corresponding to classes that overlap and do not overlap with the suppressed class are then

$$\mathcal{D}_{\mathrm{O}}(y) = \bigcup_{y' \in \mathcal{Y}_{\mathrm{O}}(y)} \mathcal{D}_{y'}, \tag{140}$$

$$\mathcal{D}_{\mathrm{NO}}(y) = \bigcup_{y' \in \mathcal{Y}_{\mathrm{NO}}(y)} \mathcal{D}_{y'}. \tag{141}$$

Denote by $\widehat{y}^{\,\mathrm{sup}}(I_{\mathrm{test}}; y)$ the model prediction after applying the suppression intervention in Eq. (133) for the target class y. For any $\mathcal{A} \subseteq \mathcal{D}_{\mathrm{test}}$, define the post-intervention accuracy

$$\mathrm{Acc}_{\mathrm{sup}}(\mathcal{A}; y) = \frac{1}{|\mathcal{A}|} \sum_{(I_{\mathrm{test}}, t) \in \mathcal{A}} \mathbf{1}\left\{\widehat{y}^{\,\mathrm{sup}}(I_{\mathrm{test}}; y) = t\right\}. \tag{142}$$

We report four test-set metrics:

$$\mathrm{SA}(y) = \mathrm{Acc}_{\mathrm{sup}}(\mathcal{D}_y; y), \tag{143}$$

$$\mathrm{OA}(y) = \mathrm{Acc}_{\mathrm{sup}}(\mathcal{D}_{\mathrm{O}}(y); y), \tag{144}$$

$$\mathrm{NOA}(y) = \mathrm{Acc}_{\mathrm{sup}}(\mathcal{D}_{\mathrm{NO}}(y); y), \tag{145}$$

$$\mathrm{TA}(y) = \mathrm{Acc}_{\mathrm{sup}}(\mathcal{D}_{\mathrm{test}}; y). \tag{146}$$

SA (suppressed accuracy) is the post-intervention accuracy on the suppressed class. **OA** (overlapped accuracy) measures retained test accuracy on non-suppressed classes that share at least one concept with $\mathcal{C}_s$, and therefore directly measures whether suppression remains selective when semantic content is shared. **NOA** (non-overlapping accuracy) measures retained test accuracy on classes whose concept sets are disjoint from $\mathcal{C}_s$. Finally, **TA** (test accuracy) is the overall post-intervention accuracy on the complete test set. All four metrics are computed on $\mathcal{D}_{\mathrm{test}}$, are reported as percentages.

F Additional Experiments

In this section, we include additional experiments and ablation studies for § 3.1.

For ablation studies, we have included the concept macro F1 score across different images per concept in Tab. F.1 and different subspace rank in Tab. F.2.

Table F.1: Concept macro-F1 (%, ↑) as a function of the number of images per concept used to fit each subspace, evaluated with ViT-B/16 (Dosovitskiy et al., 2021). Standard deviations over three seeds are shown as subscripts.

Probe	Img./concept	CIFAR-100	CUB-200	ImageNet
FFN	1	$51.24_{\pm 0.30}$	$47.19_{\pm 0.34}$	$35.10_{\pm 0.20}$
	5	$56.65_{\pm 0.17}$	$59.68_{\pm 0.20}$	$43.98_{\pm 0.09}$
	10	$57.14_{\pm 0.17}$	$62.25_{\pm 0.19}$	$46.21_{\pm 0.10}$
	20	$57.30_{\pm 0.03}$	$57.05_{\pm 0.07}$	$47.15_{\pm 0.08}$
Attn	1	$32.44_{\pm 0.22}$	$40.39_{\pm 0.61}$	$20.32_{\pm 0.26}$
	5	$47.42_{\pm 0.19}$	$56.92_{\pm 0.21}$	$35.09_{\pm 0.03}$
	10	$50.48_{\pm 0.07}$	$61.53_{\pm 0.25}$	$38.97_{\pm 0.03}$
	20	$53.28_{\pm 0.04}$	$64.62_{\pm 0.03}$	$41.85_{\pm 0.12}$
Res	1	$47.72_{\pm 0.20}$	$53.25_{\pm 0.62}$	$33.27_{\pm 0.22}$
	5	$55.05_{\pm 0.08}$	$65.42_{\pm 0.23}$	$46.19_{\pm 0.04}$
	10	$55.84_{\pm 0.11}$	$67.71_{\pm 0.12}$	$48.20_{\pm 0.02}$
	20	$56.04_{\pm 0.07}$	$68.75_{\pm 0.06}$	$49.16_{\pm 0.01}$
Last Res	1	$52.70_{\pm 0.21}$	$56.20_{\pm 0.49}$	$31.83_{\pm 0.19}$
	5	$56.73_{\pm 0.19}$	$67.37_{\pm 0.12}$	$43.70_{\pm 0.17}$
	10	$57.09_{\pm 0.14}$	$69.96_{\pm 0.04}$	$44.54_{\pm 0.05}$
	20	$57.09_{\pm 0.01}$	$70.78_{\pm 0.05}$	$43.83_{\pm 0.12}$

Table F.2: Concept macro-F1 (%, ↑) as a function of the subspace rank fitted per (concept, layer, probe point), evaluated with ViT-B/16 (Dosovitskiy et al., 2021) at 10 images per concept. Standard deviations over three seeds are shown as subscripts.

Probe	Rank	CIFAR-100	CUB-200	ImageNet
FFN	1	$55.29_{\pm 0.08}$	$56.04_{\pm 0.17}$	$46.33_{\pm 0.12}$
	2	$56.69_{\pm 0.14}$	$60.50_{\pm 0.23}$	$47.07_{\pm 0.06}$
	4	$57.11_{\pm 0.18}$	$62.19_{\pm 0.21}$	$45.58_{\pm 0.06}$
	8	$57.14_{\pm 0.17}$	$62.25_{\pm 0.19}$	$46.21_{\pm 0.10}$
Attn	1	$48.67_{\pm 0.24}$	$52.98_{\pm 0.24}$	$37.17_{\pm 0.07}$
	2	$50.65_{\pm 0.03}$	$57.92_{\pm 0.25}$	$38.03_{\pm 0.16}$
	4	$51.17_{\pm 0.07}$	$60.54_{\pm 0.27}$	$39.12_{\pm 0.03}$
	8	$50.48_{\pm 0.07}$	$61.53_{\pm 0.25}$	$38.97_{\pm 0.03}$
Res	1	$54.01_{\pm 0.07}$	$59.89_{\pm 0.20}$	$47.96_{\pm 0.08}$
	2	$55.62_{\pm 0.13}$	$64.80_{\pm 0.25}$	$48.87_{\pm 0.04}$
	4	$55.87_{\pm 0.12}$	$67.28_{\pm 0.05}$	$48.75_{\pm 0.02}$
	8	$55.84_{\pm 0.11}$	$67.71_{\pm 0.12}$	$48.20_{\pm 0.02}$
Last Res	1	$55.75_{\pm 0.13}$	$62.49_{\pm 0.04}$	$42.86_{\pm 0.13}$
	2	$55.87_{\pm 0.14}$	$63.34_{\pm 0.19}$	$43.75_{\pm 0.06}$
	4	$55.72_{\pm 0.11}$	$68.81_{\pm 0.02}$	$44.09_{\pm 0.02}$
	8	$57.09_{\pm 0.14}$	$69.96_{\pm 0.04}$	$44.54_{\pm 0.05}$

Table F.3: Unique images utilized by ViTLens per seed, deduplicated across concepts. Averages and standard deviations over three seeds are shown in the **Average** column.

Dataset	Seed 0	Seed 1	Seed 2	Average	Train set utilized
CUB-200 Wah et al. (2011)	3,966	3,954	3,954	$3{,}958.0_{\pm 6.9}$	66.0%
CIFAR-100 Krizhevsky (2009)	6,629	6,681	6,696	$6{,}668.7_{\pm 35.2}$	13.3%
ImageNet Deng et al. (2009)	41,321	41,227	41,217	$41{,}255.0_{\pm 57.4}$	3.2%

On top of that, the deduplicated counts of images utilized are reported in Tab. F.3. The computation efficiency test, including fitting speed and inference cost are reported in Tabs. F.4 and F.5, respectively. We report the fitting and inference speed on a single NVIDIA H100-96GB GPU.

Table F.4: Concept-layer fitting time (s, ↓) for each method and fitting setting. Standard deviations over three seeds are shown as subscripts.

Method	**Setting**	**CUB-200**	**CIFAR-100**	**ImageNet**
ViTLens	Full (36 sites)	$80.9_{\pm 0.5}$	$71.7_{\pm 1.9}$	$689.7_{\pm 30.7}$
	1-site est. (last layer)	$2.25_{\pm 0.01}$	$1.99_{\pm 0.05}$	$19.16_{\pm 0.85}$
Linear Probe	Full (36 sites)	$50.4_{\pm 1.9}$	$61.1_{\pm 1.4}$	$704.9_{\pm 7.3}$
	1-site est. (last layer)	$1.40_{\pm 0.05}$	$1.70_{\pm 0.04}$	$19.58_{\pm 0.20}$
CAV	Full (36 sites)	$1{,}226.3_{\pm 0.7}$	$1{,}236.4_{\pm 8.2}$	$7{,}559.2_{\pm 129.5}$
	1-site est. (last layer)	$34.06_{\pm 0.02}$	$34.34_{\pm 0.23}$	$209.98_{\pm 3.60}$
Concept Bottleneck	VLG-CBM (1 site)	$856.8_{\pm 8.3}$	$2{,}965.2_{\pm 27.7}$	$13{,}924.7_{\pm 1{,}290.8}$
	CBM (1 site)	$1{,}701.3_{\pm 26.1}$	$7{,}566.5_{\pm 99.6}$	$31{,}367.5_{\pm 666.8}$

Table F.5: Test-time inference cost (ms/sample, ↓) for each method. Standard deviations over three seeds are shown as subscripts.

	ViTLens	**Linear Probe**	**CAV**	**Concept Bottleneck**	
Dataset	**Full** 36 sites	**Full** 36 sites	**Full** 36 sites	**VLG-CBM** 1 site	**CBM** 1 site
CUB-200-2011	$21.23_{\pm 0.06}$	$23.24_{\pm 4.01}$	$20.97_{\pm 0.11}$	$1.40_{\pm 0.00}$	$1.43_{\pm 0.01}$
CIFAR-100	$20.81_{\pm 3.31}$	$20.80_{\pm 3.10}$	$21.06_{\pm 3.03}$	$1.37_{\pm 0.01}$	$1.43_{\pm 0.01}$
ImageNet	$23.50_{\pm 4.12}$	$21.60_{\pm 0.17}$	$23.96_{\pm 2.62}$	$1.55_{\pm 0.14}$	$1.34_{\pm 0.04}$

G FORGETTING

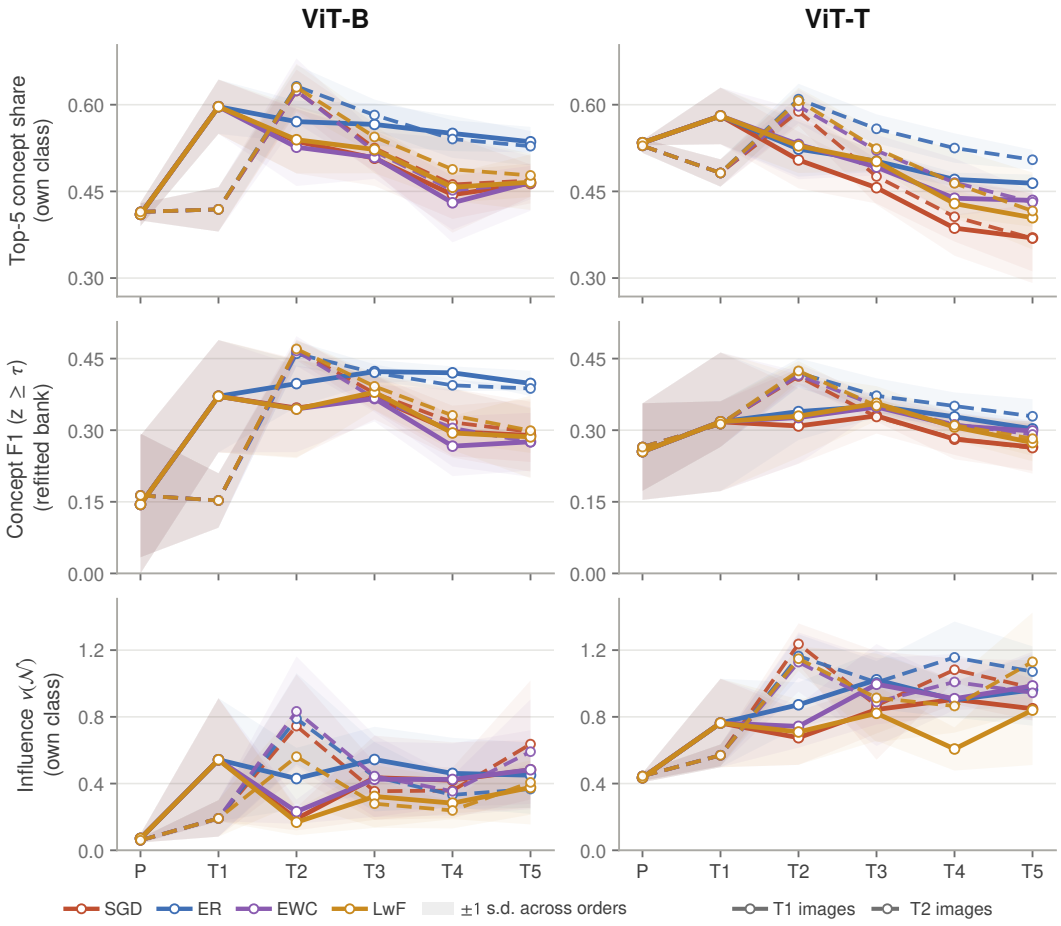


Figure G.1: ViTLens analysis on ViT-Base and ViT-Tiny on 5 sequential metaclass of CUB-200-2011 (Wah et al., 2011) for classic continual learning methods of ER (Riemer et al., 2019), EWC (Kirkpatrick et al., 2017), and LwF (Li & Hoiem, 2017). P denotes after pre-training before fine-tuning.

When a neural network is trained sequentially on changing tasks, learning a new task can sharply degrade its performance on earlier ones. This phenomenon is commonly known as catastrophic forgetting (McCloskey & Cohen, 1989; McClelland et al., 1995). Catastrophic forgetting is commonly understood as new information overwriting previously learned knowledge in the model's representation (French, 1999; Parisi et al., 2019; Luo et al., 2024). However, Davari et al. (2022) challenged this view by showing that, with ResNets, simply re-fitting the classifier using previous-task data can recover a substantial fraction of the lost performance.

We investigate this phenomenon in pre-trained ViT-B/16 and ViT-T/16 using five sequential meta-classes from CUB-200-2011. We construct each meta-class by grouping species that share a common-name category to reduce concept overlap across tasks. For example, California Gull and Western Gull are grouped under *Gull*. The experiments are conducted over three permutations of the task sequences.

In Fig. 3, we first show that, similar to ResNet behavior shown in (Davari et al., 2022), re-training a classifier with just 10 images per class from previous tasks recovers most of the lost accuracy and outperforms the continual-learning baselines we evaluate, including ER (Riemer et al., 2019), EWC (Kirkpatrick et al., 2017), and LwF (Li & Hoiem, 2017). We then refit ViTLens at the end of each task to examine what has actually been forgotten. Both concept F1 and the fraction of

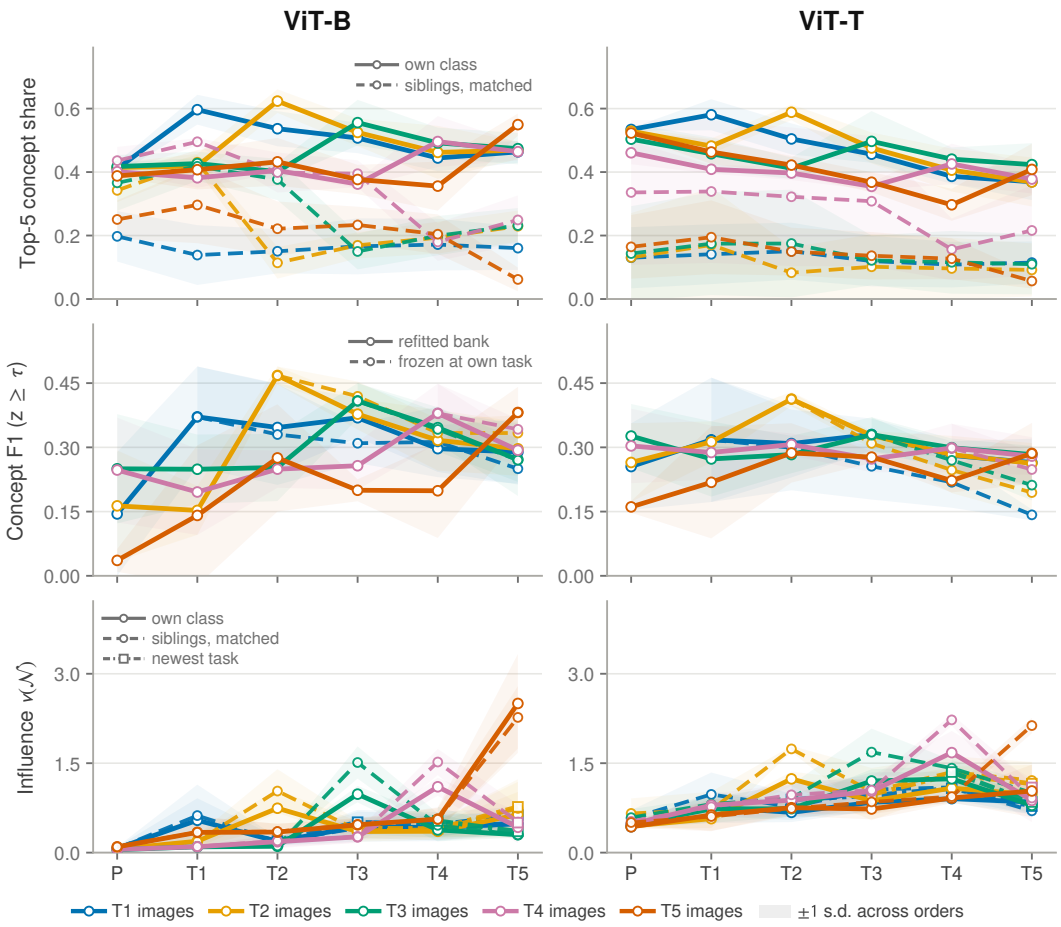


Figure G.2: ViTLens analysis on ViT-Base and ViT-Tiny on 5 sequential metaclass of CUB-200-2011 (Wah et al., 2011) for all 5 metaclasses' concepts. P denotes after pre-training before fine-tuning.

top-5 concepts belonging to the correct class degrade substantially less than classification accuracy, particularly for ViT-B. Thus, even when the model can no longer correctly classify an old task, much of its semantic structure remains detectable.

This is consistent with Ramasesh et al. (2022); Davari et al. (2022), where larger pre-trained models are shown to be more robust to catastrophic forgetting. We further find that this difference is reflected in the overlap of the geometry of the concept subspaces as defined below.

Recall that for concept c at module l, ViTLens identifies a rank-r orthonormal basis $\boldsymbol{U}_c^{(l)} \in \mathbb{R}^{d_l \times r}$ with projector $\boldsymbol{P}_c^{(l)} = \boldsymbol{U}_c^{(l)} \boldsymbol{U}_c^{(l)\top}$. For a set of concepts $\mathcal{C}$, we aggregate their subspaces through

$$\boldsymbol{M}_{\mathcal{C}}^{(l)} := \sum_{c \in \mathcal{C}} \boldsymbol{P}_c^{(l)} = \sum_{c \in \mathcal{C}} \boldsymbol{U}_c^{(l)} \boldsymbol{U}_c^{(l)\top}. \tag{147}$$

Since $M_{\mathcal{C}}^{(l)}$ is positive semidefinite, let $\lambda_1 \geq \lambda_2 \geq \cdots \geq 0$ and $u_1, u_2, \ldots$ denote its eigenvalues and eigenvectors. We retain the smallest $k_{\mathcal{C}}^{(l)}$ satisfying $\frac{\sum_{i=1}^{k_{\mathcal{C}}^{(l)}} \lambda_i}{\sum_i \lambda_i} \geq 0.95$. We then define the corresponding energy-truncated task-level subspace

$$\boldsymbol{S}_{\mathcal{C}}^{(l)} = \left[u_1, \ldots, u_{k_{\mathcal{C}}^{(l)}}, \right] \tag{148}$$

as exact span of all $\{\boldsymbol{U}_c^{(l)}\}_{c \in \mathcal{C}}$ can quickly saturate the ambient dimension and make different task sets trivially span the same space.

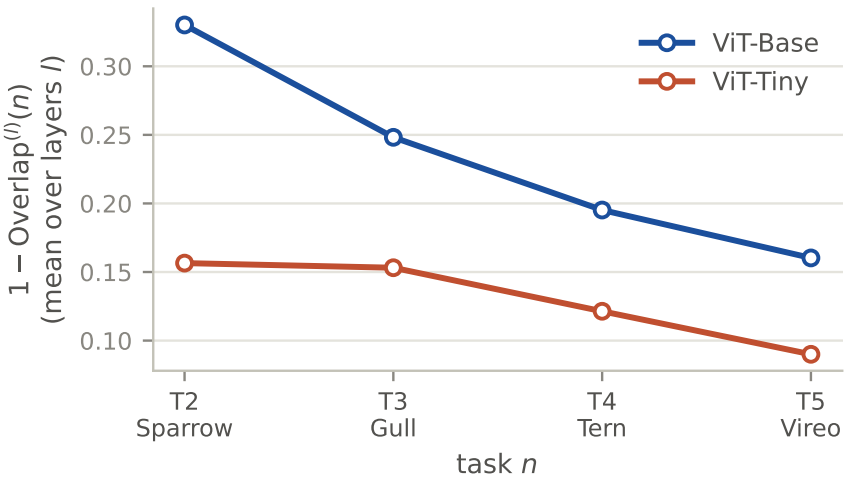


Figure G.3: Subspace overlap analysis based on Eq. (150) on ViT-Base and ViT-Tiny.

Let $\mathcal{C}_n$ denote the concepts associated with task n, and define the concepts appearing only in previous tasks as $\mathcal{C}_{<n} := \left(\bigcup_{m<n} \mathcal{C}_m\right) \setminus \mathcal{C}_n$. We write

$$\boldsymbol{S}_n^{(l)} := \boldsymbol{S}_{\mathcal{C}_n}^{(l)}, \qquad \boldsymbol{S}_{<n}^{(l)} := \boldsymbol{S}_{\mathcal{C}_{<n}}^{(l)}, \tag{149}$$

with $k_n^{(l)}$ denoting the rank of $\boldsymbol{S}_n^{(l)}$. We then measure the fraction of the current-task subspace explained by the previous-task subspace as

$$\mathrm{Overlap}^{(l)}(n) := \frac{1}{k_n^{(l)}} \left\| \boldsymbol{S}_{<n}^{(l)\top} \boldsymbol{S}_n^{(l)} \right\|_F^2 . \tag{150}$$

Thus, $\mathrm{Overlap}^{(l)}(n) \in [0, 1]$, where a larger value indicates that a greater fraction of the current-task concept subspace lies within directions already occupied by previous tasks.

As shown in Fig. G.3, ViT-Tiny has substantially greater subspace overlap between the current and previous tasks compared to ViT-B, providing one possible explanation for its greater forgetting.

ViTLens further reveals what changes when old-task performance collapses. As shown in Fig. 3, after the model is trained on a new task, the correct old-task concept can remain detectable while its influence on the output based on the concept-level game defined in Eq. (7) drops sharply, eventually becoming comparable to that of unrelated concepts. This suggests that the semantic representation can survive, while its alignment with the model's current readout deteriorates. In this sense, the observed behavior is consistent with classifier drift rather than complete erasure of the underlying concept. We provide additional results across continual-learning methods in Fig. G.1 and across the remaining meta-classes in Fig. G.2 to show that this phenomenon holds as well. However, as shown in Fig. G.2, a ViTLens fitted immediately after an earlier task also degrades when evaluated on the model after later tasks. This suggests that classifier drift is accompanied by non-trivial movement of the concept subspaces in the ambient activation space, while their relative inter-concept geometry remains largely preserved.

As providing a solution to catastrophic forgetting is beyond the scope of this work, we leave it as an interesting open question to address continual learning by preventing concept subspaces from losing their influence on the output space, rather than only minimizing representational drift.

H IMPLEMENTATION DETAILS

For tuning the threshold τ, we use a validation split drawn from the training set. Specifically, the validation sets contain 1,200 images for CUB-200-2011 (Wah et al., 2011) (20% of the training set), 5,000 images for CIFAR-100 (Krizhevsky, 2009) (10%), and 50,000 images for ImageNet-1k (Deng et al., 2009) (3.9%). We also ensure that at least two images per concepts are included in the validation set.

Table H.1: Experimental details and hyperparameter utilized.

Setup	
Backbones	ViT-B/16 and ViT-S/16 frozen. CUB-200-2011 and CIFAR-100 use fine-tuned checkpoints, ImageNet-1k the public pretrained weights
Concept labels	Open-vocabulary detections (GroundingDINO); concept c positive for image x iff its detection logit ≥ 0.15, from VLG-CBM (Srivastava et al., 2024)
Concepts evaluated	Those with ≥ 10 positive training images: 608 (CUB-200), 698 (CIFAR-100), 4167 (ImageNet-1k) — the identical set for every method
Hardware	We mainly utilized NVIDIA H100 96GB GPU. We also use NVIDIA A500, H100-47GB and A100-80GB GPUs.
Subspace fitting	
Probe points	Input to the FFN `down_proj`, input to Attn, and Res output, at all 12 layers
Token pooling	CLS
Images per concept	≤ 10, sampled uniformly
Subspace rank	$r = 8$
Estimator	Top-r left singular vectors of the $[d \times N_c]$ matrix of pooled positive activations; SVD through the origin, no mean-centering
Seeds	$\{0, 1, 2\}$. The seed selects which ≤ 10 images each concept is fitted on
Evaluation	
Aggregation	Uniform mean over the 12 layers; Last Res uses Res at the final layer, unless otherwise stated.
Threshold τ	$\{0, 0.1, 0.2, 0.25, 0.5, 0.75, 1, 1.5, 2, 2.5, 3, 3.5, 4, 5, 6, 7, 8, 9, 10\}$ are swept after z-score normalization.
Metric	Per-sample multi-label macro-F1 over the evaluated concepts; mean $\pm$ standard deviation over the 3 seeds
Baselines	
kNN	Mean cosine similarity to the $k = 3$ nearest of the same ≤ 10 exemplars
KMeans	Negative squared distance to the nearest of $\min(3, N_c)$ per-concept centroids
Linear probe	One shared classifier per layer over all concepts, multi-hot targets, 200 full-batch Adam steps, lr 10^{-2}, weight decay 10^{-4}
CAV	We utilized 10 positive and 10 negative exemplars per concept. For fairness negative exemplars are randomly sampled from the overall probing set defined by 10 positive exemplars per concept.
CBM & VLG-CBM	Thresholds applied to z-score normalized logits, so τ is selected within each family. CBM (Koh et al., 2020) is implemented with the joint training setup (concept bottleneck layer and classifier jointly trained). More details in Tab. H.3.

Table H.2: Classification accuracy of backbone ViT-B/16 and ViT-S/16 (Dosovitskiy et al., 2021) and respective CBM (Koh et al., 2020) and VLG-CBM (Srivastava et al., 2024).

Backbone	Model	CIFAR-100	CUB-200-2011	ImageNet-1k
ViT-B/16	Base ViT	88.40	83.77	81.07
	CBM	$86.91_{\pm 0.16}$	$77.75_{\pm 0.92}$	$67.90_{\pm 0.25}$
	VLG-CBM	$88.21_{\pm 0.01}$	$83.09_{\pm 0.10}$	$79.52_{\pm 0.04}$
ViT-S/16	Base ViT	88.77	86.58	81.38
	CBM	$87.30_{\pm 0.12}$	$83.32_{\pm 1.04}$	$64.04_{\pm 0.32}$
	VLG-CBM	$88.81_{\pm 0.05}$	$86.23_{\pm 0.06}$	$77.76_{\pm 0.10}$

Table H.3: Hyperparameters for the CBM (Koh et al., 2020) and VLG-CBM (Srivastava et al., 2024) baselines. Settings are identical for the ViT-B/16 and ViT-S/16 backbones.

	Hyperparameter	CIFAR-100	CUB-200-2011	ImageNet-1k
Shared	Optimizer	Adam	Adam	SGD ($m = 0.9$)
	Concept-layer lr η_{head}	5×10^{-4}	5×10^{-4}	10^{-2}
	Epochs	40	75	7
	Batch size	32	32	64
	Weight decay		10^{-5}	
	Concept loss		binary cross-entropy (`BCEWithLogits`)	
	BCE positive weight	$0.2 \cdot N_c^- / N_c^+$	1.0	8000
	Crop-to-concept prob.	0.0	0.5	0.5
VLG-CBM	Backbone		frozen	
	Bottleneck hidden layers	1	0	0
	Final layer		GLM-SAGA, elastic net $\alpha = 0.99$, step size 0.1, batch 512	
	sparsity λ	7×10^{-4}	2×10^{-4}	5×10^{-4}
	iterations	2000	4000	80
CBM	Backbone		trained jointly with bottleneck and classifier	
	Bottleneck hidden layers		0 (single linear layer)	
	Class-loss weight λ_{cls}		1.0	
	Backbone lr ratio ρ	0.02	0.02	1.0
	Backbone lr η_{bb}	10^{-5}	10^{-5}	10^{-2}
	Learning-rate schedule	cosine	constant	constant

Table H.4: Backbone fine-tuning hyperparameters for CIFAR-100 and CUB-200-2011. ImageNet-1k uses the pretrained weights directly and therefore requires no fine-tuning.

Hyperparameter	**CIFAR-100**	**CUB-200-2011**
Shared fine-tuning settings		
Optimizer	AdamW	AdamW
Head learning rate	10^{-3}	10^{-3}
Backbone learning rate	10^{-4}	10^{-4}
Weight decay	0.05	0.05
Full fine-tuning epochs	40	40
LR schedule	Cosine, $\eta_{\min} = 10^{-6}$, $T_{\max} = 40$	
Loss	Cross-entropy, label smoothing $= 0.1$	
Gradient clipping	$\|g\|_2 \leq 1.0$	
Training augmentation	RandomResizedCrop$(224,\ 0.6\text{–}1.0)$, horizontal flip ColorJitter$(0.3, 0.3, 0.3, 0.05)$	
Normalization	mean $=$ std $= 0.5$	
Seed	42	
ViT-B/16		
Pretrained weights	`torchvision ViT_B_16_Weights.IMAGENET1K_V1`	
Head-only warmup epochs	5	5
Batch size	128	32
Evaluation transform	Resize$(224, 224)$	Resize(256) CenterCrop(224)
ViT-S/16		
Pretrained weights	`timm vit_small_patch16_224` (AugReg IN-1k)	
Head-only warmup epochs	0	0
Batch size	32	32

Table H.5: Best thresholds selected for each method and setting using ViT-S/16 and ViT-B/16.

Method	**Setting**	**CIFAR-100**		**CUB-200**		**ImageNet**	
		S/16	B/16	S/16	B/16	S/16	B/16
CBM	Best Thr.	4	5	2	2	7	8
VLG-CBM	Best Thr.	2.5	3	4	4	5	7
Linear probe	FFN	0.5	1	0.5	0.1	0.5	0
	Attn	1	1	0.5	0.5	0.5	0.1
	Res	1	1.5	0.5	0.5	0.5	0.1
	Last Res	1.5	2.5	4	2.5	5	1
KMeans	FFN	1.5	1	1.5	1	2	1.5
	Attn	1.5	1	1.5	1.5	2	1.5
	Res	1.5	1.5	1.5	1.5	2	2
	Last Res	2	1.5	2	1.5	3	3
kNN	FFN	0.5	1	1	1	2	1.5
	Attn	1	2	2	2	2.5	3
	Res	0.1	1	0.1	1	1	2
	Last Res	0.1	0.5	1	1	7	4
CAV	FFN	1.5	1.5	2	1	2	1.5
	Attn	1.5	2	2	2	2.5	2.5
	Res	1.5	2	2	2	2	2.5
	Last Res	2.5	2.5	3	3	4	4
ViTLens	FFN	0.75	1	1.5	1	2.5	2
	Attn	1	2	2	2.5	3.5	4
	Res	0.25	1	0.25	1.5	1	2.5
	Last Res	0.5	1	1.5	1.5	10	6

I SUBSPACE VISUALIZATION

I.1 SUBSPACE SEMANTIC

In this subsection, we will first visualize the ViTLens energy over Res, FFN and Attn probing sites across layers of ViT-B/16 (Fig. I.2) and the subspace visualization based on Lucid (Olah et al., 2017), in which the subspace energy is utilized as loss function and the gradient is backpropagated onto the pixel space as shown in Fig. I.1.

In both Figs. I.1 and I.2 we can clearly observe that human-interpretable concepts begin to emerge in the middle layers of the ViT. Interestingly, the quality of the Lucid visualizations at the final layer is consistently poorer than that at Layers 5 and 7. Human-interpretable patterns are clearly visible at Layers 5 and 7 but become substantially less discernible at Layer 11.

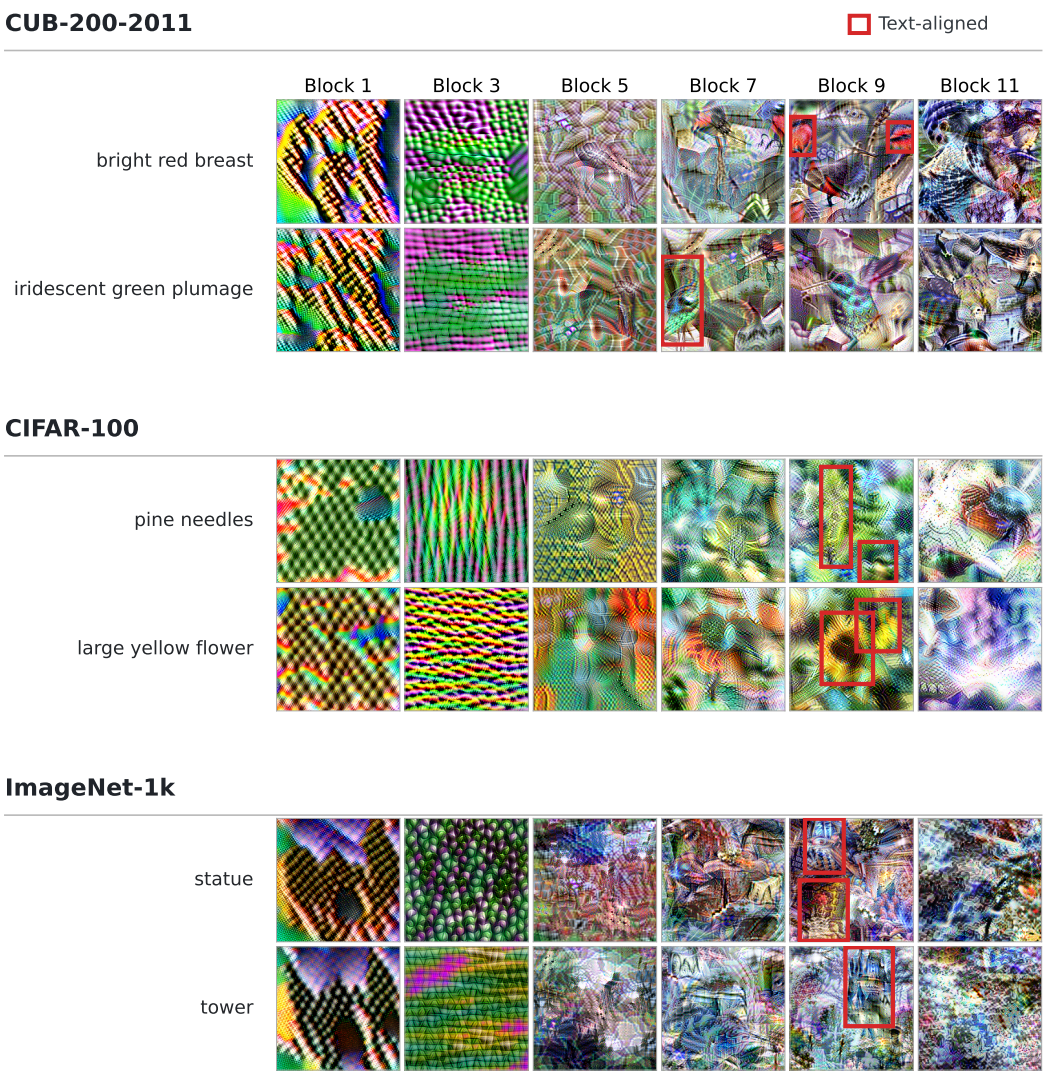


Figure I.1: Lucid Olah et al. (2017) visualization of ViTLens fitted subspace for CIFAR-100 Krizhevsky (2009), CUB-200-2011 Wah et al. (2011) and ImageNet-1k Deng et al. (2009) on ViT-B/16.



Figure I.2: Visualization of $\hat{E}_c^{(l)}(\boldsymbol{x}^{(l)})$ for CIFAR-100 Krizhevsky (2009), CUB-200-2011 Wah et al. (2011) and Imagenet-1k Deng et al. (2009).

I.2 SUBSPACE GEOMETRY

In this subsection, we include the cross-layers ViTLens fitted subspace UMAP (McInnes et al., 2018) visualization (Fig. I.3) and the binary tree formed based on the chordal distance between subspace and tree-depth based on the layer of which the subspace is fitted on (Figs. I.4 to I.6). All subspace analyzed are at the Res probing site.

The text in red for Figs. I.4 to I.6 denotes the GPT-4 annotated parent-node concept labels obtained by prompting GPT-4 with the concept annotation of the descendant leaf nodes from Srivastava et al. (2024).

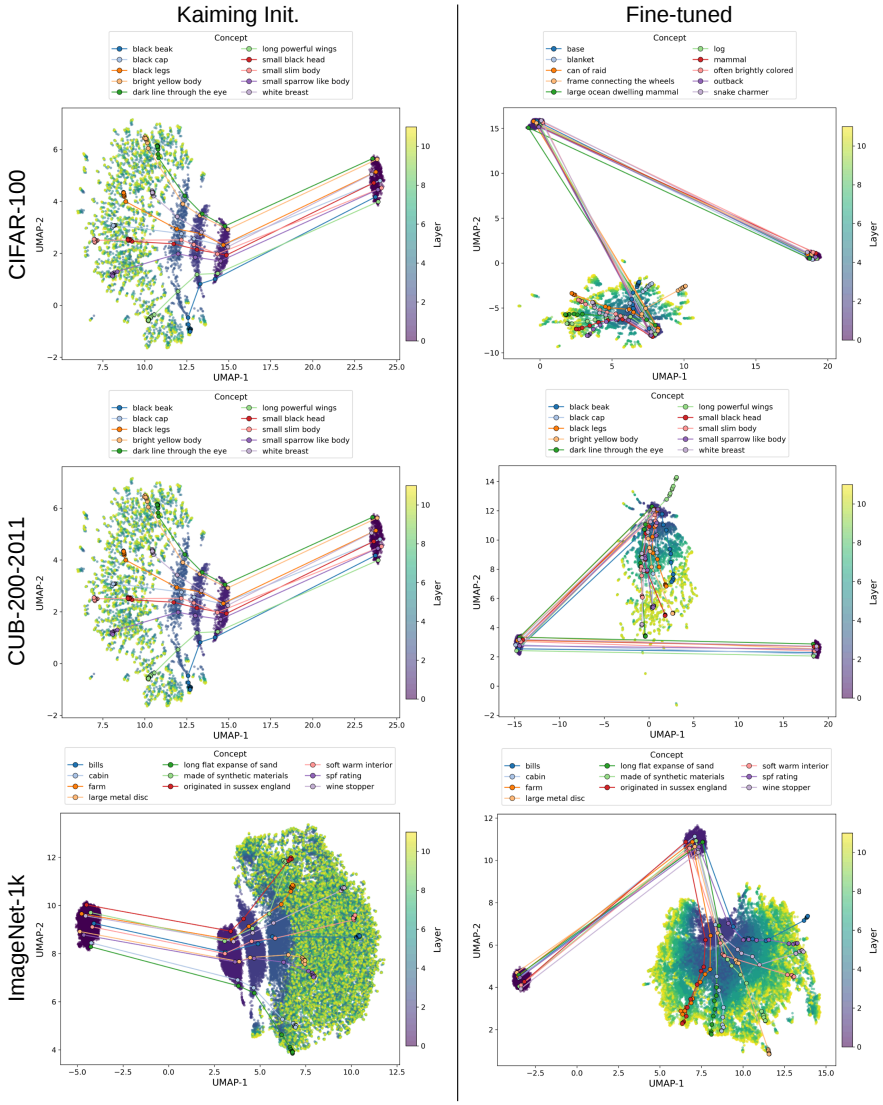
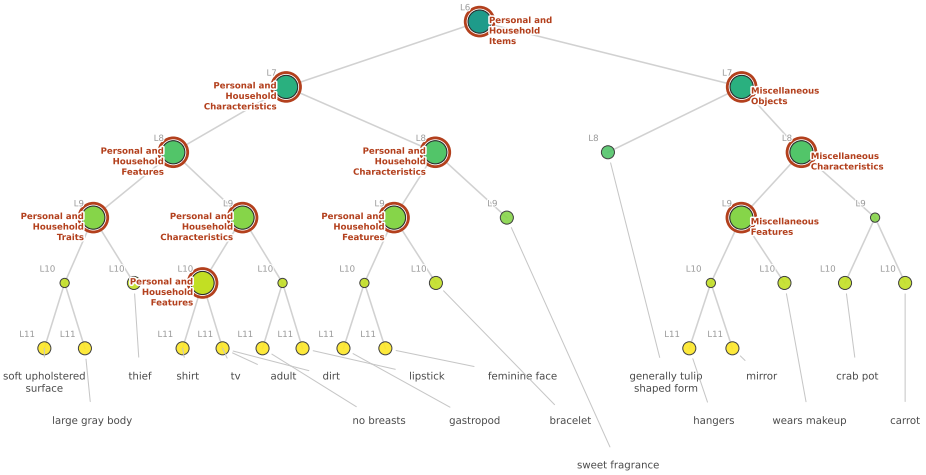
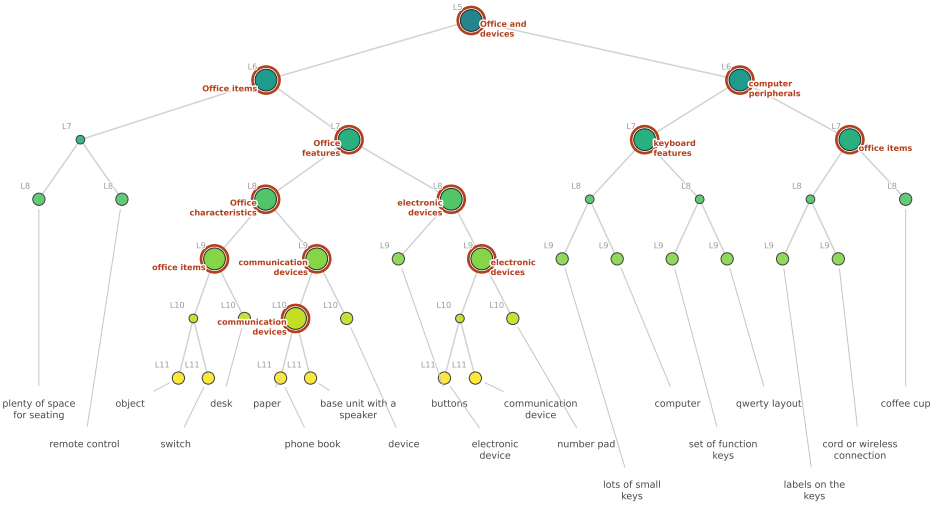


Figure I.3: UMAP plot of the concept subspaces identified across layers on Kaiming initialized and fine-tuned ViT-B/16 for CIFAR-100, CUB-200-2011 and ImageNet-1k. We utilized chordal distance instead of standard euclidean distance as subspaces are invariant to basis rotations.

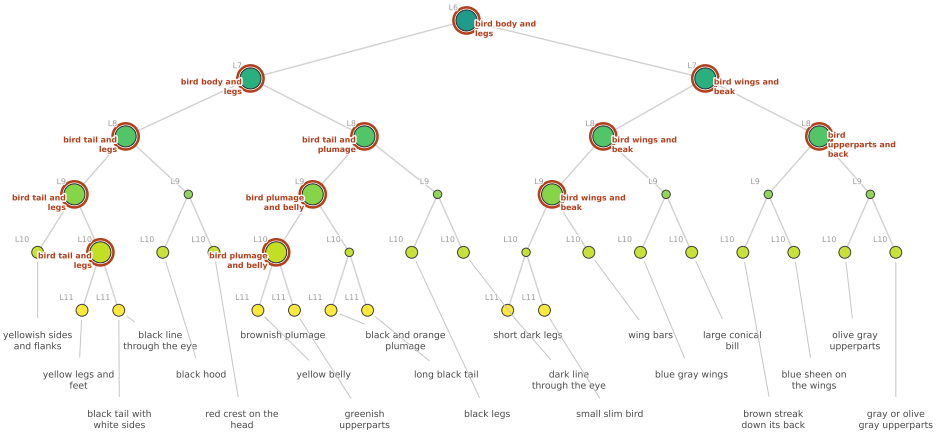


(a) Kaiming-initialized, untrained ViT-B/16

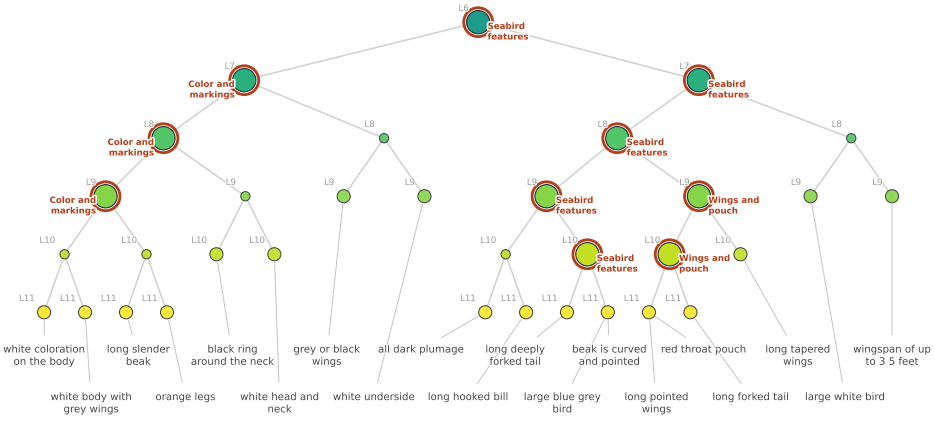


(b) Fine-tuned ViT-B/16

Figure I.4: Subset of branches for the concept subspaces identified across layers on CIFAR-100.

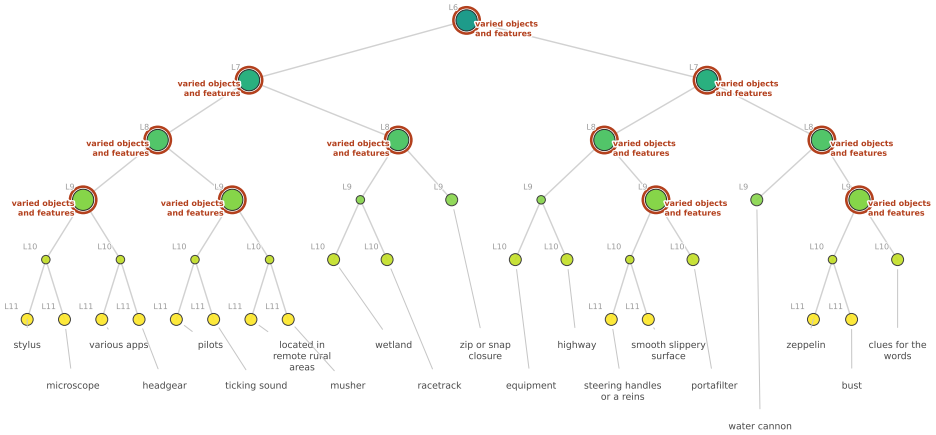


(a) Kaiming-initialized, untrained ViT-B/16

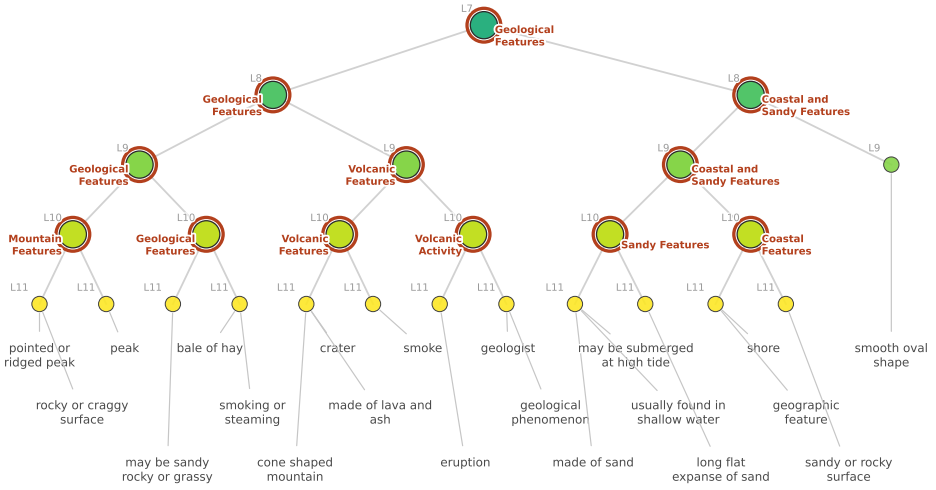


(b) Fine-tuned ViT-B/16

Figure I.5: Subset of branches for the concept subspaces identified across layers on CUB-200-2011.



(a) Kaiming-initialized, untrained ViT-B/16



(b) Fine-tuned ViT-B/16

Figure I.6: Subset of branches for the concept subspaces identified across layers on ImageNet-1k.