Hyperbolic Concept Bottleneck Models

Daniel Uyterlinde
Informatics Institute
University of Amsterdam
Amsterdam, The Netherlands
danieluyterlinde@gmail.com

Swasti S. Mishra
Informatics Institute
University of Amsterdam
Amsterdam, The Netherlands
s.s.mishra@uva.nl

Pascal Mettes
Informatics Institute
University of Amsterdam
Amsterdam, The Netherlands
p.s.m.mettes@uva.nl

Abstract

Concept Bottleneck Models (CBMs) have become a popular approach to enable interpretability in neural networks by constraining classifier inputs to a set of human-understandable concepts. While effective, current models embed concepts in flat Euclidean space, treating them as independent, orthogonal dimensions. Concepts, however, are highly structured and organized in semantic hierarchies. To resolve this mismatch, we propose Hyperbolic Concept Bottleneck Models (HypCBM), a post-hoc framework that grounds the bottleneck in this structure by reformulating concept activation as asymmetric geometric containment in hyperbolic space. Rather than treating entailment cones as a pre-training penalty, we show they encode a natural test-time activation signal: the margin of inclusion within a concept's entailment cone yields sparse, hierarchy-aware activations without any additional supervision or learned modules. We further introduce an adaptive scaling law for hierarchically faithful interventions, propagating user corrections coherently through the concept tree. Empirically, HypCBM rivals post-hoc Euclidean models trained on $20\times$ more data in sparse regimes required for human interpretability, with stronger hierarchical consistency and improved robustness to input corruptions.

1 Introduction

The advent of large black-box models, particularly vision-language models [39, 24, 25], across high-stakes domains including healthcare [23, 30], finance [15], and autonomous systems [49], has intensified the need for models whose predictions are transparent and interpretable. While various frameworks offer post-hoc explanations [14, 13, 54], Concept Bottleneck Models (CBMs) [20] have emerged as a canonical approach by enforcing an intermediate, interpretable bottleneck. By requiring the model to predict human-understandable concepts before making a final decision, this architecture exposes internal reasoning and, crucially, enables users to intervene and correct erroneous predictions at the concept level. CBMs have established themselves in interpretable learning, enabling domain experts to de-bias models and improve reliability through human-in-the-loop interaction [46].

Although recent frameworks have improved CBM scalability and usability [57, 33, 48], they retain a limiting representational assumption: concept independence. In real-world settings, concepts are rarely independent, but visually co-occurring and semantically structured, exhibiting relationships ranging from synonymy to inherent hierarchies [3]. While some approaches enforce structure through auxiliary modules or constraints [37, 35, 50], the underlying bottleneck activations are derived from

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

a flat Euclidean embedding space, where concepts are treated as independent, orthogonal directions. This geometry is fundamentally at odds with hierarchical logic: in a tree-like structure, the number of nodes grows exponentially with depth, yet Euclidean space cannot embed branching relationships without forcing parent and child concepts into overlapping or distorted regions [45]. The result is redundant concept sets, logical inconsistencies (e.g., "*Dog*" without "*Animal*"), and inefficient interventions.

Hyperbolic geometry offers a solution to this mismatch. Its exponential volume growth makes it the natural domain for hierarchical representation learning, a property that has driven advances in taxonomy embedding [31, 11], image embeddings [18], segmentation [2], and hierarchical classification [9, 26]. Recently, hyperbolic vision-language models [8, 34] have shown that this geometry can be aligned with semantic structure at scale: through entailment cone training objectives, parent concepts come to geometrically subsume their children. We show that this geometric structure encodes a natural concept activation signal that is hierarchically consistent by construction and directly interpretable at inference time, without any additional supervision or learned modules.

Building on this insight, we introduce Hyperbolic Concept Bottleneck Models (HypCBM). Rather than relying on symmetric distance metrics that distort hierarchical relationships, we reformulate concept activation as an asymmetric geometric containment problem in hyperbolic space. Specifically, our contributions are:

- We propose a **hyperbolic entailment activation mechanism** that maps image representations to sparse, hierarchy-aware concept activations through entailment cone containment.
- We introduce an **adaptive scaling law for hierarchical interventions** to propagate corrections coherently through the concept tree. Because hyperbolic volume expands exponentially with depth, a fixed entailment threshold fails; we formulate a scaling mechanism that adapts strictness to the geometric specificity of the parent concept.
- We demonstrate that HypCBM consistently **improves accuracy** over existing post-hoc CBMs, rivaling Euclidean models trained on $20\times$ more data in sparse regimes, while also enhancing data efficiency, robustness, and hierarchical consistency.

2 Related Work

2.1 Concept Bottleneck Models

Concept Bottleneck Models (CBMs) represent a prominent paradigm in explainable machine learning. Unlike classical attribution methods (e.g., LIME [43], SHAP [27]), CBMs are interpretable-by-design. First proposed by Koh et al. [20], CBMs align the predictive process of neural networks with human reasoning by introducing an intermediate layer of semantically meaningful concepts. This is done by decomposing the classification task into two mappings: a concept extractor $f : \mathcal{X} \to \mathbb{R}^K$ that maps raw inputs $\mathbf{x}$ to a concept activation vector $\mathbf{c} \in \mathbb{R}^K$, and an interpretable predictor $g : \mathbb{R}^K \to \mathcal{Y}$ predicting label $\mathbf{y}$ based solely on $\mathbf{c}$. This bottleneck crucially enables test-time intervention: a human expert can correct wrongly predicted concept activations to correct the final model prediction [46].

Despite their promise, standard CBMs require dense concept supervision, limiting scalability. Early post-hoc interpretation methods such as TCAV [19] mitigated this by training linear classifiers per concept, but still required per-concept annotation. Post-hoc CBMs (PCBMs) [56] and their label-free variant LF-CBM [33] eliminated this by exploiting the shared embedding space of pre-trained VLMs: concept vectors $\mathbf{v}_i$ are retrieved via the text encoder, and activation is computed as cosine similarity $c_i = \langle f(\mathbf{x}), \mathbf{v}_i \rangle$, converting any backbone into an interpretable model without additional training.

Table 1: **Taxonomy of popular CBM methods.** HypCBM uniquely achieves a structured bottleneck completely post-hoc without learned modules. A full overview is provided in Appendix G.

Method	Post-hoc	Zero-shot Concepts	Structured Bottleneck	Learned Modules
LaBo [55]	✓	✓	✗	Optimized concept selection
VLG-CBM [47]	✗	✗	✗	Grounding + learned projection
Hierarchical CBM [37]	✗	✗	✓	Supervised hierarchy layers
CRM [6]	✗	✗	✓	Graph reasoning
LogicCBM [50]	✗	✗	✓	Neuro-symbolic reasoning
PCBM [56] / LF-CBM [33]	✓	✓	✗	**None**
HypCBM (Ours)	✓	✓	✓	**None**

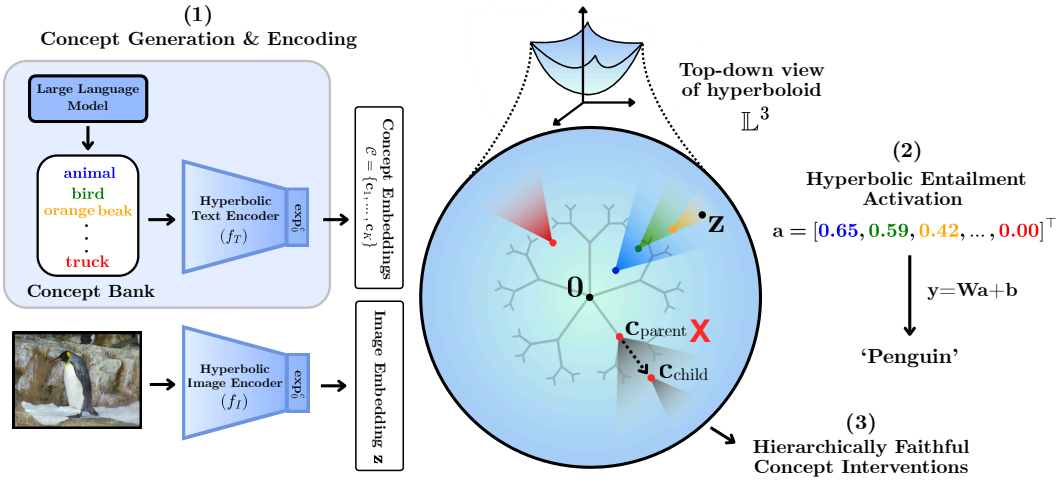


Figure 1: **Method overview.** **(1)** Generated concepts and the target image are encoded with a hyperbolic VLM onto the hyperbolic manifold, where they are hierarchically organized. **(2)** The activation of a concept is measured as the *margin of inclusion* of the image embedding in the entailment cone of the concept. **(3)** An intervention on a parent concept $(\mathbf{c}_{\text{parent}})$ is propagated to all entailed children $(\mathbf{c}_{\text{child}})$.

Subsequent frameworks extended this through language-supervised bottlenecks [55], learned concept embeddings [57], unsupervised concept discovery [42], and vision-grounded concept annotation [47].

A separate line of work addresses the structural limitation of flat concept spaces by enforcing explicit semantic relationships. These approaches achieve structural validity through supervised hierarchy layers [37], probabilistic coarse-to-fine modeling [35], or neuro-symbolic rules and graph-based reasoning [50, 10, 6]. However, all such methods require learned modules and depart from the post-hoc paradigm, entangling the effect of structural inductive bias with that of additional supervision. **Direct comparison with these methods is therefore neither fair nor feasible:** they require task-specific training incompatible with our post-hoc setting, and their auxiliary modules are designed for flat Euclidean spaces and cannot be natively extended to hyperbolic geometry.

2.2 Hyperbolic Vision-Language Models

Standard vision-language models like CLIP [39] and ALIGN [17] align image and text encoders via a symmetric contrastive objective. While remarkably effective for zero-shot generalization [41, 44], their reliance on Euclidean geometry limits their ability to capture the intrinsic hierarchy of visual concepts [8]. Recent works address this by generalizing VLMs to hyperbolic space, whose exponential volume growth naturally accommodates tree-like semantic structures. MERU [8] pioneered this direction by redefining image-text alignment as an entailment problem, using hyperbolic entailment cones to enforce asymmetric containment. HyCoCLIP [34] extended this to compositional object-part hierarchies, and HySAC [38] applied the same principle for safety-aware representations. We leverage these pre-trained hyperbolic backbones as the geometric foundation for constructing hierarchically faithful concept bottlenecks. Given the rapid growth of entailment-based hyperbolic VLMs [40, 16], HypCBM presents a zero-shot pathway to interpretability for this emerging model class, with concept hierarchy emerging naturally from the underlying geometry.

Table 1 positions HypCBM within this landscape. It is the only framework that achieves a structured bottleneck while remaining fully post-hoc with no learned modules. By treating the backbone as a frozen feature extractor, any empirical gains over post-hoc Euclidean baselines are attributable directly to the geometric formulation, free from confounding supervision artifacts.

3 Methodology

We introduce HypCBM, a post-hoc framework that enriches concept bottlenecks with a hierarchical organization through hyperbolic geometry. We begin by formalizing the necessary preliminaries of hyperbolic space. We then detail the concept generation and encoding phase, followed by our two

core technical contributions: a hyperbolic entailment activation mechanism and an adaptive strategy for hierarchically faithful concept interventions.

3.1 Preliminaries on Hyperbolic Geometry

We encode concepts and images in a hyperbolic embedding space, a Riemannian manifold of constant negative curvature $-c$, with $c > 0$. While several isometric models of hyperbolic space exist (e.g., the Poincaré ball), we utilize the Lorentz model (or hyperboloid model), known for its superior numerical stability for optimization compared to the Poincaré ball, particularly avoiding vanishing gradients near the boundary, and has become a popular choice for hyperbolic representation learning [32, 29].

Formally, the n-dimensional Lorentz model $\mathbb{L}_c^n$ is defined as the upper sheet of a two-sheeted hyperboloid in an $(n+1)$-dimensional Minkowski spacetime [5]:

$$\mathbb{L}_c^n = \left\{ \mathbf{x} \in \mathbb{R}^{n+1} : \langle \mathbf{x}, \mathbf{x} \rangle_{\mathbb{L}} = -\frac{1}{c}, c > 0 \right\}. \tag{1}$$

A point $\mathbf{x} \in \mathbb{L}_c^n$ decomposes into a time component x_0 and spatial component $\tilde{\mathbf{x}} \in \mathbb{R}^n$, where $x_0 = \sqrt{1/c + \|\tilde{\mathbf{x}}\|^2}$ is fully determined by $\tilde{\mathbf{x}}$. This allows us to work exclusively with the n-dimensional spatial component while respecting manifold geometry.

Standard neural networks (like the CLIP image and text encoders) produce features $\mathbf{v} \in \mathbb{R}^n$ in a flat Euclidean space. To map these features onto the hyperbolic manifold, we utilize the exponential map at the origin, $\mathbf{o} = (\sqrt{1/c}, \mathbf{0})^\top$. This maps a tangent vector $\mathbf{v}$ (identified with $\mathbb{R}^n$) to $\mathbb{L}_c^n$ via:

$$\exp_{\mathbf{0}}(\mathbf{v}) = \cosh(\sqrt{c}\|\mathbf{v}\|)\mathbf{0} + \frac{\sinh(\sqrt{c}\|\mathbf{v}\|)}{\sqrt{c}\|\mathbf{v}\|}\mathbf{v}. \tag{2}$$

Inverse operations map points back to the tangent space via the logarithmic map $\log_{\mathbf{0}}(\cdot)$, enabling standard Euclidean operations when necessary.

3.2 Concept Generation and Encoding

Following the label-free, post-hoc paradigm of Oikarinen et al. [33], we automate concept discovery without manual annotations in two steps: generating a candidate set of visually descriptive attributes and encoding them onto the hyperbolic manifold.

For concept generation, we prompt an LLM with dataset-specific class labels to produce visually descriptive attributes per class. These per-class lists are aggregated into a global set of unique textual descriptions $\mathcal{T} = \{t_1, \ldots, t_M\}$. Full prompts and generation examples are provided in Appendix B.

The resulting descriptions are embedded into hyperbolic space using the frozen text encoder $f_T : \mathcal{T} \to \mathbb{L}_c^n$ of any pre-trained hyperbolic VLM. The initial concept bank $\mathcal{C}_{\text{init}}$ of size M is defined as:

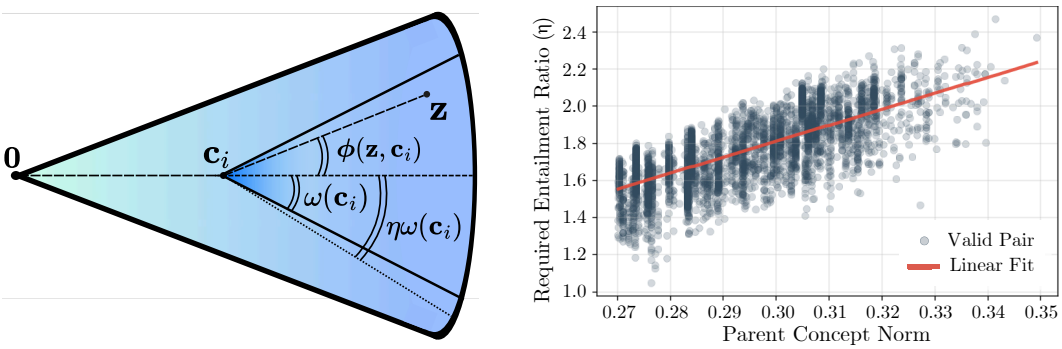
$$\mathcal{C}_{\text{init}} = \{\mathbf{c}_i \in \mathbb{L}_c^n \mid \mathbf{c}_i = f_T(t_i)\}_{i=1}^M. \tag{3}$$

Crucially, each vector $\mathbf{c}_i$ is positioned according to the semantic hierarchy learned by the backbone during pre-training: specific concepts lie within the cones of their generic parents, with correspondingly larger spatial norms [8, 34]. We exploit this intrinsic organization in the activation mechanism described next.

Norm-Based Filtering. While generated concept banks provide rich features, they often contain generic terms (e.g., 'object', 'entity') that lack discriminative power. By embedding concepts as vectors in a hyperbolic embedding space, we are able to filter out overly generic concepts simply by computing their distance to the origin. Overly generic concepts inflate the size of the concept bank without adding significant predictive value since such concepts are applicable to many samples. We therefore apply a geometric filter to prune them, defining the final concept bank $\mathcal{C}$ as:

$$\mathcal{C} = \{\mathbf{c} \in \mathcal{C}_{init} \mid \|\tilde{\mathbf{c}}\| \geq \tau\}. \tag{4}$$

We set $\tau = 0.27$ based on the HyCoCLIP [34] concept embeddings, filtering the bottom $\approx 20^{\text{th}}$ percentile of non-discriminative root concepts. To ensure a fair comparison, we utilize this resulting concept bank for both the Euclidean and hyperbolic models. We empirically validate in Appendix A.8 that this geometric filtering is critical for hyperbolic models, whereas Euclidean baselines show marginal gains from similar pruning. The final concept counts (N_c) per dataset are reported in Table 2.



(a) Geometric Entailment

(b) Concept-to-Concept Scaling Law

Figure 2: **Geometry and Scaling of Hyperbolic Entailment. (a)** An image $\mathbf{z}$ activates concept $\mathbf{c}_i$ if the exterior angle $\phi(\mathbf{z}, \mathbf{c}_i)$ falls within the scaled cone half-aperture $\eta\omega(\mathbf{c}_i)$. Here, η denotes the strictness parameter η_{img} (Eq. 7). **(b)** Empirical scaling law derived from WordNet. The entailment strictness η_{text} required to geometrically capture true descendants scales linearly with the parent concept's norm ($r = 0.729$).

3.3 Hyperbolic Entailment Activation

Unlike standard symmetric distance metrics (e.g., Euclidean cosine similarity), HypCBM formulates concept activation as an asymmetric geometric containment problem on the Lorentz manifold. While prior work restricts entailment cones to pre-training penalties [11, 8, 34], we show that the resulting geometric structure serves directly as an inference mechanism.

Specifically, the concept activation a_i measures the degree to which image embedding $\mathbf{z}$ is geometrically contained within the entailment cone of concept $\mathbf{c}_i$ (see Fig. 2a), yielding a sparse, geometrically grounded activation vector that is hierarchically consistent by construction.

The aperture of this cone is determined by the concept's specificity. Following [22, 8], we define the cone's half-aperture $\omega(\mathbf{c}_i)$ as inversely proportional to the concept's spatial norm, with scaling constant $K = 0.04$ balancing cone expressivity against saturation (see Appendix D):

$$\omega(\mathbf{c}_i) = \sin^{-1}\left(\frac{2K}{\sqrt{c}\|\tilde{\mathbf{c}}_i\|}\right). \tag{5}$$

To determine whether the image embedding $\mathbf{z}$ falls within the entailment cone, we compute the *exterior angle* $\phi(\mathbf{z}, \mathbf{c}_i)$ relative to the concept axis [8, 34]:

$$\phi(\mathbf{z}, \mathbf{c}_i) = \cos^{-1}\left(\frac{z_0 + (\mathbf{c}_i)_0\, c\langle \mathbf{z}, \mathbf{c}_i\rangle_{\mathbb{L}}}{\|\tilde{\mathbf{c}}_i\|\sqrt{(c\langle \mathbf{z}, \mathbf{c}_i\rangle_{\mathbb{L}})^2 - 1}}\right). \tag{6}$$

Geometrically, the partial order relation is satisfied when this exterior angle is strictly smaller than the cone's half-aperture. We define the concept activation as the *margin of inclusion*, normalized by the cone aperture:

$$a_i = \max\left(0, \eta_{\text{img}} - \frac{\phi(\mathbf{z}, \mathbf{c}_i)}{\omega(\mathbf{c}_i)}\right), \tag{7}$$

resulting in a naturally sparse, non-negative activation vector $\mathbf{a} = [a_1, \ldots, a_M]^\top$ where $a_i > 0$ implies geometric entailment. Here, η_{img} is a strictness parameter calibrated to separate true positive entailments from false associations (see Appendix C). Normalizing by $\omega(\mathbf{c}_i)$ measures *relative containment depth*, or the ratio of remaining margin to total cone aperture, making activations comparable across concepts of varying specificities: an identical activation value indicates the same relative geometric alignment regardless of whether the concept is generic or specific. Overly generic concepts with broad cones that would activate across most images are further suppressed by norm-based filtering (Section 3.2).

Finally, we predict class logits using a sparse linear layer $g(\mathbf{a}) = \mathbf{W}\mathbf{a} + \mathbf{b}$, where $\mathbf{W}$ is regularized with Elastic-Net ($\alpha = 0.99$) to ensure decisions rely on a small set of active concepts, with sparsity controlled by regularization parameter λ.



(a) **Top activating images.** (b) **Explaining Decisions.**

Figure 3: We validate the interpretability of HypCBM through **(a)** inspection of the top activating images per concept to verify semantic alignment, and **(b)** decision explanations showing the top activating concepts for two images, from ImageNet (top) and SUN397 (bottom).

3.4 Hierarchically Faithful Concept Interventions

A critical limitation of Euclidean CBMs is the lack of structural consistency during test-time interventions. Manually deactivating a parent concept (e.g., *"Animal"*) leaves its child concepts (e.g., *"Dog"*) active, resulting in logical paradoxes. HypCBM resolves this by automatically propagating interventions through the semantic concept hierarchy: when a user suppresses a parent concept, all geometric descendants are as well (see Fig. 1).

A concept $\mathbf{c}_{\text{child}}$ is identified as a descendant of $\mathbf{c}_{\text{parent}}$ if it falls within the parent's entailment cone under a text-to-text strictness threshold η_{text}. However, a fixed threshold is insufficient across hierarchy depths: due to the exponential volume growth of hyperbolic space, the entailment margin required to correctly capture true descendants varies with the parent's specificity.

To address this, we derive an adaptive scaling law for η_{text} from 10k WordNet [28] pairs, finding that the required entailment margin scales linearly with the parent's norm (Fig. 2b, Appendix C.2). When a user suppresses a concept $\mathbf{c}_{\text{parent}}$ (i.e., setting $a_{\text{parent}} \to 0$), we automatically suppress all $\mathbf{c}_{\text{child}}$ concepts satisfying:

$$\phi(\mathbf{c}_{\text{child}}, \mathbf{c}_{\text{parent}}) < \eta_{\text{text}}(\|\tilde{\mathbf{c}}_{\text{parent}}\|) \cdot \omega(\tilde{\mathbf{c}}_{\text{parent}}). \tag{8}$$

Interventions operate on the activation vector $\mathbf{a}$, not on the image embedding $\mathbf{z}$. The descendant set of each concept is determined once from concept-to-concept geometry and is independent of the input image. Propagation is strictly top-down, so suppressing a child concept leaves its parent unchanged. This mechanism allows users to suppress an entire branch of spurious concepts with a single interaction; on concept banks with deeper hierarchies, a single correction removes several concepts at once (Section 4.3).

4 Experiments

We evaluate HypCBM across three domains: **CIFAR-100** [21] for general object classification, **SUN397** [52] for (hierarchical) scene understanding, and **ImageNet** [7] to assess scalability to real-world complexity. Additional results on **CUB-200** [51] are provided in Appendix A. To isolate the impact of geometry, we control for training data by utilizing the pre-trained checkpoints from Pal et al. [34], ensuring both the hyperbolic HyCoCLIP and Euclidean CLIP [39] are trained on the same GrIT-20M dataset [36]. We also report results for CLIP [39], trained on 400M samples, to contextualize performance against a high-resource upper bound. For all comparisons involving activation sparsity (Sections 4.2-4.4), the LF-CBM baseline is thresholded to match the average sparsity of HypCBM. This effectively binarizes the otherwise continuous activations into discrete active sets, ensuring a fair comparison. All experiments are conducted on a single NVIDIA A100 GPU with curvature $c = 0.1$, consistent with the pre-training configuration of the hyperbolic backbone, HyCoCLIP [34]. Due to the cost of the linear solver (up to 5 hours per operating point on ImageNet, Appendix F), main results are from a single run; across three seeds on CIFAR-100, the standard deviation at ANEC-20 is below 0.15 percentage points (Appendix A.5).

4.1 Accuracy at Number of Effective Concepts

Standard accuracy obscures interpretability costs, as a model can score well using dense activations across all concepts. We therefore report *Accuracy at Number of Effective Concepts* (ANEC) [47],

Table 2: **Accuracy at number of effective concepts.** Test accuracy at strict concept budgets (K). HypCBM consistently outperforms Euclidean baselines (LF-CBM) on identical backbones (HyCoCLIP-20M) and often even baselines that use large-scale pre-trained backbones (CLIP-400M and CLIP^{RE}-400M). Linear probing accuracy is reported as an upper bound reference.

			Accuracy @ Concept Budget (ANEC-K)						
Dataset	**Method**	**Backbone**	**5**	**10**	**20**	**50**	**100**	**Peak**	**LP**
CIFAR-100 ($N_c = 613$)	LF-CBM	CLIP-20M	39.7	48.9	55.4	60.6	64.5	67.9	73.1
	LF-CBM	HyCoCLIP-20M	25.6	34.6	56.3	61.6	62.6	69.5	76.3
	HypCBM (Ours)	HyCoCLIP-20M	**44.3**	**54.4**	**60.9**	**69.7**	**71.8**	**72.2**	76.3
	Ref: LF-CBM	*CLIP-400M*	*28.0*	*58.0*	*63.9*	*64.7*	*67.0*	*71.3*	*82.6*
SUN397 ($N_c = 1935$)	LF-CBM	CLIP-20M	45.8	53.4	61.3	67.0	68.6	70.5	74.8
	LF-CBM	HyCoCLIP-20M	34.3	44.2	50.4	59.1	61.5	72.1	75.9
	HypCBM (Ours)	HyCoCLIP-20M	**49.3**	**58.3**	**65.7**	**69.9**	**71.3**	**72.3**	75.9
	Ref: LF-CBM	*CLIP-400M*	*40.0*	*48.0*	*58.8*	*64.4*	*66.8*	*79.0*	*82.7*
ImageNet ($N_c = 2950$)	LF-CBM	CLIP-20M	7.7	15.3	30.7	47.3	53.7	55.3	63.7
	LF-CBM	HyCoCLIP-20M	10.2	15.5	26.0	40.6	46.2	57.9	65.4
	HypCBM (Ours)	HyCoCLIP-20M	**17.9**	**34.6**	**44.2**	**52.3**	**56.8**	**58.1**	65.4
	Ref: LF-CBM	*CLIP-400M*	*17.4*	*34.0*	*51.8*	*52.6*	*54.0*	*73.8*	*78.5*

which measures performance under a strict budget of K active concepts. We analyze the results in Table 2 through three findings, and evaluate data efficiency across reduced training set sizes.

1. Better accuracy. HypCBM consistently achieves higher peak performance than its Euclidean counterparts. Even when the bottleneck is relaxed ($K \to$ all), HypCBM outperforms the standard LF-CBM baseline across all three datasets. For instance, on CIFAR-100, HypCBM reaches a peak accuracy of 72.2%, surpassing LF-CBM (HyCoCLIP-20M, 69.5%) by 2.7 percentage points and LF-CBM (CLIP-20M, 67.9%) by over 4 percentage points. Since the backbone is identical in the former comparison, these gains are attributable directly to the geometric activation mechanism rather than representational capacity.

2. Sparsity leads to further gains. While HypCBM performs best across the board, its advantages are most pronounced in the high-sparsity regime, critical for interpretability. When restricted to small concept budgets, the performance gap widens significantly; for instance, on ImageNet ($K = 10$), HypCBM more than doubles the baseline accuracy (34.6% vs. 15.3%). These results rely on correct geometric alignment: applying a mismatched Euclidean activation mechanism to the hyperbolic backbone often degrades performance below the Euclidean baseline, showing the gains stem from our method as opposed to the backbone.

3. HypCBM compensates for scale. HypCBM frequently matches or outperforms the OpenAI CLIP baseline, despite being trained on $20\times$ less data. For instance, on the hierarchical SUN397 benchmark at ANEC-20, HypCBM achieves 65.7% accuracy, beating the Euclidean CLIP-400M baseline by nearly 7 percentage points (65.7% vs. 58.8%). HypCBM attains this performance with just 20 concepts, matching an accuracy level that the CLIP-400M baseline requires ≈ 70 active concepts to reach. This demonstrates that entailment-based concept activations can compensate for substantial pre-training data disadvantages in sparse interpretable regimes.

4. Data efficiency. HypCBM consistently outperforms LF-CBM across all training data regimes (Table 3). With only 1% of CIFAR-100 available as training data, HypCBM achieves an ANEC-20 of 43.4%, surpassing LF-CBM by over 13 percentage points, with low variance across seeds confirming result stability. This demonstrates that entailment-based concept activations remain effective even under severe data constraints. Full accuracy-sparsity curves with standard deviations are provided in Appendix A.5.

Table 3: **Data Efficiency:** ANEC-20 at 1% and 10% training data, CIFAR-100 (mean $\pm$ std, 3 seeds).

Method	**1%**	**10%**
LF-CBM	30.2 ± 1.7	45.9 ± 0.7
HypCBM	**43.4** ± 0.6	**54.7** ± 0.4

Table 4: **Hierarchical Consistency.** HypCBM (HyCoCLIP-20M) consistently outperforms the Euclidean LF-CBM (CLIP-20M) baseline across validated WordNet pairs, with relative gains exceeding $+100\%$ at the strictest budget.

	SUN397 ($N = 1503$)			**ImageNet** ($N = 829$)		
η_{img}	LF-CBM	HypCBM	**Gain**	LF-CBM	HypCBM	**Gain**
1.5	0.63	**0.84**	**+32.5%**	0.59	**0.75**	**+27.9%**
1.4	0.43	**0.54**	**+27.2%**	0.42	**0.63**	**+51.1%**
1.3	0.21	**0.33**	**+58.9%**	0.28	**0.51**	**+80.4%**
1.2	0.08	**0.17**	**+104.7%**	0.16	**0.33**	**+102.9%**

4.2 Hierarchical Consistency

Beyond accuracy, interpretable models should exhibit logical coherence: if a fine-grained concept is active (*'school'*), its parent (*'building'*) must be too. We quantify **Hierarchical Consistency** as:

$$\text{Consistency} = 1 - \frac{\sum \mathbf{1}[a_{\text{child}} = 1 \wedge a_{\text{parent}} = 0]}{\sum \mathbf{1}[a_{\text{child}} = 1]} \tag{9}$$

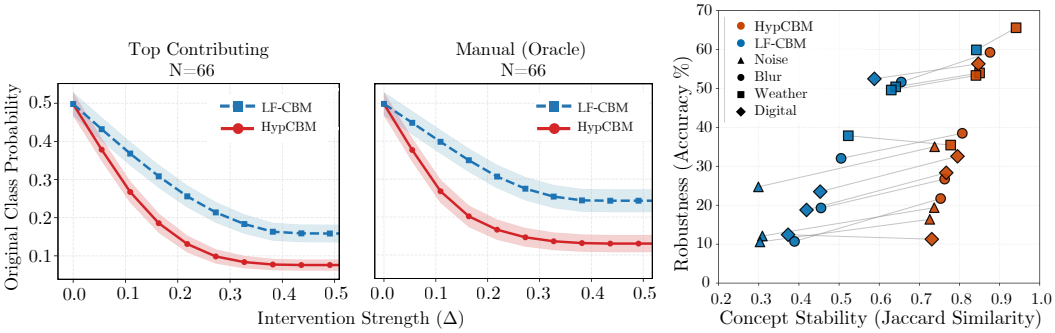
We evaluate on SUN397 and ImageNet using parent-child relations validated via WordNet, and use sparsity-matched baselines to prevent trivially achieving 100% consistency by activating all concepts.

Results. Table 4 reveals a consistent pattern: HypCBM's relative advantage over the Euclidean baseline grows as η_{img} tightens. At $\eta_{\text{img}} = 1.2$ the baseline degrades sharply (0.08 on SUN397) while HypCBM more than doubles its consistency, confirming that entailment-based activations prioritize hierarchically valid concepts even under severe budget constraints.

To disentangle the two components of our approach, Appendix A.6 includes two ablations. First, replacing cosine similarity with our entailment activation on the *identical* HyCoCLIP-20M backbone nearly doubles consistency at every threshold, isolating the activation mechanism as the primary driver of gains independent of backbone choice. Second, comparing against a hierarchy-aware Euclidean model [1] trained on 20× more data shows that HypCBM matches or exceeds it at the strictest budgets, suggesting the hyperbolic manifold contributes an inductive bias beyond what entailment-aligned training alone provides.

4.3 Hierarchically Faithful Concept Interventions

An essential part of any CBM is the ability to support human intervention. To quantify the utility of our hierarchical interventions, we simulate a human-in-the-loop correction scenario on misclassified test samples. For each incorrect prediction, we identify the active concepts contributing to the error and progressively suppress selected concepts by subtracting a scalar factor Δ from their activations. At each step, we measure the confidence assigned to the originally predicted (incorrect) class, yielding an intervention response curve.



(a) **Intervention Responsiveness.** (b) **Robustness and Concept Stability.**

Figure 4: **Intervention and Stability Analysis.** (a) HypCBM (red) displays steeper confidence decay than the Euclidean LF-CBM, indicating superior responsiveness to corrective interventions. Shaded bands show Standard Error (SE). (b) On CIFAR100-C, HypCBM maintains significantly higher semantic stability and accuracy across corruptions. LF-CBM uses the CLIP-20M backbone.

We consider three strategies: **top-contributing** (suppressing the highest $w_i \cdot a_i$ concept), **manual oracle** (suppressing a top-contributing, ground-truth absent concept), and **random** (control baseline). We report results for top-contributing and manual strategies; random suppression serves as a control baseline. HypCBM and LF-CBM are evaluated at matched sparsity for fairness.

Under manual intervention, HypCBM flips 19% more misclassified samples to the correct class than LF-CBM. A single hierarchically propagated correction therefore resolves more errors than the equivalent correction in a flat bottleneck, with a correspondingly steeper confidence decay (Fig. 4a). Propagation efficiency scales with hierarchy depth: a single top-contributing intervention prunes a branch on 83% of SUN397 and 91% of ImageNet images (median 3 and 4 concepts), versus mostly leaves on the flatter CIFAR-100 (60%, median 1).

Two controls support this advantage. First, random interventions removing a comparable number of concepts cause negligible confidence changes (Appendix A.4), confirming that observed gains stem from targeted suppression rather than general activation reduction. Second, augmenting LF-CBM with WordNet-derived propagation yields no measurable benefit: only $101/613$ CIFAR-100 concepts and $618/1935$ SUN397 concepts appear in WordNet, and none of the top contributing concepts have valid WordNet children. HypCBM's geometric propagation discovers implicit hierarchical relationships that taxonomy-based approaches cannot recover.

4.4 Robustness to Corruptions and Concept Stability

Robustness is a prerequisite for interpretability: explanations that change inconsistently under benign perturbations are unreliable, even if predictions remain correct. We therefore evaluate robustness at both the prediction and concept levels using CIFAR100-C [12], which applies 15 types of corruption (e.g., noise, blur, weather) at varying severities.

Besides measuring **Prediction Robustness** (accuracy on corrupted data), we introduce **Concept Stability**: the semantic consistency of the model's explanations. To the best of our knowledge, no prior work has standardized a metric for explanation stability in CBMs, despite its importance for reliable human-in-the-loop interaction. We compute the Jaccard Similarity between the set of active concepts in the clean image (S_{clean}) and the corrupted image (S_{corr}):

$$J(S_{clean}, S_{corr}) = \frac{|S_{clean} \cap S_{corr}|}{|S_{clean} \cup S_{corr}|}. \tag{10}$$

HypCBM activations are naturally sparse and non-negative, allowing us to define concept activity via non-zero activation. We define the active concept sets S using the sparsity-matched binarization protocol described at the start of this section. This ensures that differences in stability arise strictly from geometric structure rather than differing activation densities.

Results. Fig. 4b highlights the resilience of our hyperbolic approach; the Euclidean LF-CBM baseline fails to maintain semantic consistency, with Jaccard similarity scores dropping as low as 0.3 on noise corruptions. In contrast, HypCBM exhibits significantly higher concept stability, achieving an average Jaccard improvement of $+0.3$ over the baseline across corruption categories (see Appendix A.7 for results at higher severities). Moreover, HypCBM achieves higher classification accuracy in 13 out of 15 corruption settings, confirming a strong association between stable semantic representations and robust predictive performance. This conclusion does not depend on binarization. Under a threshold-free metric, the centered cosine similarity between clean and corrupted activation vectors, HypCBM is more stable on all 15 corruptions at severity 3 (mean 0.866 vs. 0.606; Appendix A.7).

5 Conclusion

Existing Concept Bottleneck Models embed semantic concepts in flat Euclidean spaces, treating them as independent and orthogonal. We address this by grounding the concept bottleneck in hyperbolic geometry and replacing symmetric similarity with asymmetric geometric entailment. The resulting model, HypCBM, achieves concept activations that are hierarchically consistent by construction, without additional supervision or learned modules. The empirical results confirm that this geometric alignment translates into consistent gains in accuracy, data efficiency, robustness, and intervention quality, often rivaling models trained on substantially more data. Together, these findings suggest

that structural priors on the representation space can serve as a powerful substitute for scale in sparse interpretable regimes.

Our findings highlight that interpretability is not merely a property of model outputs, but is shaped by the geometry of the representation space. Embedding concepts in a manifold that reflects their hierarchical relationships leads to explanations that are more accurate, robust to corruptions, and hierarchically consistent.

Limitations and future work. Our ablations partially isolate the contributions of representational geometry and activation design, but fully disentangling them remains an open question. We compare only against post-hoc baselines with the same concept bank; a qualitative comparison with CBMs that learn their concept selection, such as LaBo [55], would show whether hierarchical structure also surfaces in their explanations, but would mostly reflect differences in concept selection rather than geometry. Similarly, we assess human usability only through proxies, namely qualitative explanations and simulated interventions. A user study comparing intervention burden across CBMs would strengthen these claims. Additionally, HypCBM relies on taxonomic hierarchies derived from text (e.g., WordNet [28]), which do not always align with continuous visual semantics. In fine-grained settings such as CUB-200 [51], discriminative cues are localized attributes (e.g., *"spotted wing"*) rather than semantic parent-child relationships; incorporating visually grounded hierarchies is a promising direction. Finally, HypCBM's performance depends on the quality of available hyperbolic VLMs. We do, however, view this as an opportunity: as hyperbolic foundation models continue to advance [16], HypCBM is designed to scale with them naturally, providing a framework for translating structured representations into interpretable and controllable classifiers.

Acknowledgments and Disclosure of Funding

This work was supported by the ELLIS unit Amsterdam and Qualcomm Technologies, Inc.

References

[1] Morris Alper and Hadar Averbuch-Elor. Emergent visual-semantic hierarchies in image-text representations. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

[2] Mina Ghadimi Atigh, Julian Schoep, Erman Acar, Nanne Van Noord, and Pascal Mettes. Hyperbolic Image Segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4453–4462, 2022.

[3] Pietro Barbiero, Francesco Giannini, Gabriele Ciravegna, Michelangelo Diligenti, and Giuseppe Marra. Relational concept bottleneck models. *Advances in Neural Information Processing Systems*, 37:77663–77685, 2024.

[4] Usha Bhalla, Alex Oesterling, Suraj Srinivas, Flavio P Calmon, and Himabindu Lakkaraju. Interpreting clip with sparse linear concept embeddings (splice). *Advances in Neural Information Processing Systems*, 37:84298–84328, 2024.

[5] James W Cannon, William J Floyd, Richard Kenyon, and Walter R Parry. Hyperbolic geometry. In *Flavors of geometry*, pages 59–115. Cambridge University Press, 1997.

[6] David Debot, Pietro Barbiero, Gabriele Dominici, and Giuseppe Marra. Interpretable Hierarchical Concept Reasoning through Attention-Guided Graph Learning, 2025. URL `https://arxiv.org/abs/2506.21102`.

[7] 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*, pages 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

[8] Karan Desai, Maximilian Nickel, Tanmay Rajpurohit, Justin Johnson, and Shanmukha Ramakrishna Vedantam. Hyperbolic Image-Text Representations. In *International Conference on Machine Learning*, pages 7694–7731. PMLR, 2023.

[9] Ankit Dhall, Anastasia Makarova, Octavian Ganea, Dario Pavllo, Michael Greeff, and Andreas Krause. Hierarchical image classification using entailment cone embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020.

[10] Giovanni De Felice, Arianna Casanova Flores, Francesco De Santis, Silvia Santini, Johannes Schneider, Pietro Barbiero, and Alberto Termine. Causally Reliable Concept Bottleneck Models, 2025. URL `https://arxiv.org/abs/2503.04363`.

[11] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic Entailment Cones for Learning Hierarchical Embeddings. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 80, pages 1646–1655. PMLR, 2018.

[12] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019.

[13] Lijie Hu, Yixin Liu, Ninghao Liu, Mengdi Huai, Lichao Sun, and Di Wang. Improving interpretation faithfulness for vision transformers. In *Forty-first International Conference on Machine Learning*, 2023.

[14] Lijie Hu, Yixin Liu, Ninghao Liu, Mengdi Huai, Lichao Sun, and Di Wang. Seat: stable and explainable attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 12907–12915, 2023.

[15] Jimin Huang, Mengxi Xiao, Dong Li, Zihao Jiang, Yuzhe Yang, Yifei Zhang, Lingfei Qian, Yan Wang, Xueqing Peng, Yang Ren, et al. Open-finllms: Open multimodal large language models for financial applications. *arXiv preprint arXiv:2408.11878*, 2024.

[16] Chuong Huynh, Hossein Souri, Abhinav Kumar, Vitali Petsiuk, Deen Dayal Mohan, and Suren Kumar. Argent: Adaptive hierarchical image-text representations, 2026. URL `https://arxiv.org/abs/2603.23311`.

[17] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139, pages 4904–4916. PMLR, 2021.

[18] Valentin Khrulkov, Leyla Mirvakhabova, Evgeniya Ustinova, Ivan Oseledets, and Victor Lempitsky. Hyperbolic Image Embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6418–6428, 2020.

[19] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV). In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2668–2677. PMLR, 2018. URL `https://proceedings.mlr.press/v80/kim18d.html`.

[20] 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*, pages 5338–5348. PMLR, 2020.

[21] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

[22] Matthew Le, Stephen Roller, Laetitia Papaxanthos, Douwe Kiela, and Maximilian Nickel. Inferring concept hierarchies from text corpora via hyperbolic embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3231–3241. Association for Computational Linguistics, July 2019. doi: 10.18653/v1/P19-1313. URL `https://aclanthology.org/P19-1313/`.

[23] Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. LLaVA-Med: Training a large language-and-vision assistant for biomedicine in one day. *Advances in Neural Information Processing Systems*, 36: 28541–28564, 2023.

[24] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022.

[25] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

[26] Teng Long, Pascal Mettes, Heng Tao Shen, and Cees G. M. Snoek. Searching for Actions on the Hyperbole. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1138–1147, 2020. doi: 10.1109/CVPR42600.2020.00122.

[27] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4768–4777, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

[28] George A. Miller. WordNet: a lexical database for English. *Commun. ACM*, 38(11):39–41, November 1995. ISSN 0001-0782. doi: 10.1145/219717.219748. URL `https://doi.org/10.1145/219717.219748`.

[29] Gal Mishne, Zhengchao Wan, Yusu Wang, and Sheng Yang. The Numerical Stability of Hyperbolic Representation Learning. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 24925–24949. PMLR, 23–29 Jul 2023. URL `https://proceedings.mlr.press/v202/mishne23a.html`.

[30] Michael Moor, Qian Huang, Shirley Wu, Michihiro Yasunaga, Yash Dalmia, Jure Leskovec, Cyril Zakka, Eduardo Pontes Reis, and Pranav Rajpurkar. Med-flamingo: a multimodal medical few-shot learner. In *Machine Learning for Health (ML4H)*, pages 353–367. PMLR, 2023.

[31] Maximilian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

[32] Maximilian Nickel and Douwe Kiela. Learning Continuous Hierarchies in the Lorentz Model of Hyperbolic Geometry. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 80, pages 3779–3788. PMLR, 2018.

[33] Tuomas Oikarinen, Subhro Das, Lam M Nguyen, and Tsui-Wei Weng. Label-Free Concept Bottleneck Models. In *International Conference on Learning Representations (ICLR)*, 2023.

[34] Avik Pal, Max Van Spengler, Guido D'Amely di Melendugno, Alessandro Flaborea, Fabio Galasso, and Pascal Mettes. Compositional entailment learning for hyperbolic vision-language models. In *International Conference on Learning Representations*, volume 2025, pages 87371–87399, 2025.

[35] Konstantinos P. Panousis, Dino Ienco, and Diego Marcos. Coarse-to-fine concept bottleneck models. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9798331314385.

[36] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, Qixiang Ye, and Furu Wei. Grounding multimodal large language models to the world. In *The Twelfth International Conference on Learning Representations*, 2024.

[37] Federico Pittino, Vesna Dimitrievska, and Rudolf Heer. Hierarchical concept bottleneck models for vision and their application to explainable fine classification and tracking. *Engineering Applications of Artificial Intelligence*, 118:105674, 2023.

[38] Tobia Poppi, Tejaswi Kasarla, Pascal Mettes, Lorenzo Baraldi, and Rita Cucchiara. Hyperbolic Safety-Aware Vision-Language Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[39] 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 *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139, pages 8748–8763. PMLR, 2021.

[40] Sameera Ramasinghe, Violetta Shevchenko, Gil Avraham, and Ajanthan Thalaiyasingam. Accept the modality gap: An exploration in the hyperbolic space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 27263–27272, June 2024.

[41] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical Text-Conditional Image Generation with CLIP Latents, 2022. URL `https://arxiv.org/abs/2204.06125`.

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

[43] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1135–1144. Association for Computing Machinery, 2016. ISBN 9781450342322. doi: 10.1145/2939672.2939778. URL `https://doi.org/10.1145/2939672.2939778`.

[44] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-Resolution Image Synthesis With Latent Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, June 2022.

[45] Rik Sarkar. Low Distortion Delaunay Embedding of Trees in Hyperbolic Plane. In Marc van Kreveld and Bettina Speckmann, editors, *Graph Drawing*, pages 355–366, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. ISBN 978-3-642-25878-7.

[46] Sungbin Shin, Yohan Jo, Sungsoo Ahn, and Namhoon Lee. A closer look at the intervention procedure of concept bottleneck models. In *International Conference on Machine Learning*, pages 31504–31520. PMLR, 2023.

[47] Divyansh Srivastava, Ge Yan, and Tsui-Wei Weng. VLG-CBM: Training Concept Bottleneck Models with Vision-Language Guidance. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 79057–79094. Curran Associates, Inc., 2024. doi: 10.52202/079017-2510.

[48] David Steinmann, Wolfgang Stammer, Felix Friedrich, and Kristian Kersting. Learning to intervene on concept bottlenecks. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

[49] Xiaoyu Tian, Junru Gu, Bailin Li, Yicheng Liu, Yang Wang, Zhiyong Zhao, Kun Zhan, Peng Jia, Xianpeng Lang, and Hang Zhao. DriveVLM: The Convergence of Autonomous Driving and Large Vision-Language Models, 2024. URL `https://arxiv.org/abs/2402.12289`.

[50] Deepika SN Vemuri, Gautham Bellamkonda, Aditya Pola, and Vineeth N Balasubramanian. LogicCBMs: Logic-Enhanced Concept-Based Learning, 2025. URL `https://arxiv.org/abs/2512.07383`.

[51] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. CUB200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[52] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3485–3492, 2010. doi: 10.1109/CVPR.2010.5539970.

[53] Haotian Xu, Tsui-Wei Weng, Lam M. Nguyen, and Tengfei Ma. Graph concept bottleneck models. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=a4azUYjRhU.

[54] Shu Yang, Lijie Hu, Lu Yu, Muhammad Asif Ali, and Di Wang. Human-ai interactions in the communication era: Autophagy makes large models achieving local optima. *CoRR*, 2024.

[55] Yue Yang, Artemis Panagopoulou, Shenghao Zhou, Daniel Jin, Chris Callison-Burch, and Mark Yatskar. Language in a bottle: Language model guided concept bottlenecks for interpretable image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19187–19197, 2023.

[56] Mert Yuksekgonul, Maggie Wang, and James Zou. Post-hoc concept bottleneck models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=nA5AZ8CEyow.

[57] Mateo Zarlenga, Pietro Barbiero, Gabriele Ciravegna, Giuseppe Marra, Francesco Giannini, Michelangelo Diligenti, Zohreh Shams, Frederic Precioso, Stefano Melacci, Adrian Weller, Pietro Lio, and Mateja Jamnik. Concept embedding models: Beyond the accuracy-explainability trade-off. *Advances in Neural Information Processing Systems*, 35, 2022.

A Additional Results

A.1 CUB200

In this section, we provide a detailed analysis of HypCBM's performance on CUB-200-2011 [51]. Unlike the other datasets in this paper, which use LLM-generated concept banks, here we use the 312 expert-annotated visual attributes provided with the CUB-200-2011 dataset [51], demonstrating that HypCBM generalizes beyond LLM-generated concept banks to pre-defined ones.

Table 5: **Additional Results.** Test accuracy at strict concept budgets (K) on CUB-200 using pre-defined expert-annotated attributes. We report peak accuracy and compare against the larger CLIP-400M backbone.

			Accuracy @ Concept Budget (ANEC-K)						
Dataset	**Method**	**Backbone**	**5**	**10**	**20**	**50**	**100**	**Peak**	**LP**
CUB-200 ($N_c = 317$)	LF-CBM	CLIP-20M	6.4	11.0	17.7	30.8	40.7	41.8	56.5
	LF-CBM	HyCoCLIP-20M	2.9	4.3	7.3	15.1	21.9	28.1	58.7
	HypCBM (Ours)	HyCoCLIP-20M	**13.5**	**20.8**	**25.8**	**36.8**	**43.6**	**45.3**	58.7
	Ref: LF-CBM	*CLIP-400M*	*6.2*	*16.4*	*31.2*	*45.1*	*50.2*	*55.0*	*81.8*

CUB-200. The CUB-200 dataset presents a challenging fine-grained classification task that relies heavily on localized features (e.g., 'beak shape', 'wing pattern'). In general, our multimodal approach is not well-suited for these kinds of scenarios, as two concepts like 'white undertail covert' and 'white undertail feather' might be represented very semantically similar, while they could be important discriminators. As shown in Table 5, the pre-trained HyCoCLIP and CLIP backbones (trained on 20M image-text pairs) struggle to capture these fine-grained distinctions compared to the OpenAI CLIP backbone (trained on 400M pairs). The standard Euclidean LF-CBM built on HyCoCLIP collapses under this representational gap, achieving a poor 2.86% accuracy with 5 active concepts.

However, despite this poor performance, HypCBM demonstrates a remarkable ability to compensate for its weaker backbone. Most notably, at extremely low sparsity budgets (K=5 and K=10), HypCBM (trained on 20M data) surprisingly outperforms the LF-CBM CLIP-400M baseline (13.47% vs. 6.23% at K=5), suggesting that hyperbolic geometry allows for more efficient disentanglement of available semantic information in low-data regimes. As the concept budget relaxes ($K \geq 20$), the raw feature quality of the 400M backbone eventually dominates, leading to a higher peak accuracy for the OpenAI baseline.

Another interesting result we observed, is the fact that accuracy does not keep increasing when more concepts are made available, but instead peaks at ≈ 150 active concepts and then degrades. This result may seem like a flaw at first, but is logically faithful; when classifying bird species, a model should not be able to use almost all of its available concepts. When it starts using false positive concepts this adds noise, hurting classification accuracy. In contrast, with the Euclidean baselines this happens only at the very end of the accuracy/sparsity curve, indicating information leakage; standard CBMs allow the model to assign small, non-zero weights to semantically irrelevant concepts, trading interpretability for the raw predictive power of the underlying dense embedding.

A.2 Hyperbolic Entailment Activation, CLIP, MERU & HyCoCLIP

Results. Table 6 isolates the impact of embedding geometry. We observe that mapping standard CLIP features to $\mathbb{L}_c^n$ fails to create a functional hierarchy: because CLIP features are L2-normalized, they map to a fixed distance from the origin, resulting in constant (very small) cone apertures for all concepts. Without radial variance to distinguish generic from specific attributes, the entailment condition collapses, forcing the model to rely solely on angular separation. Furthermore, while MERU is natively hyperbolic, it significantly underperforms in the sparse regime ($K \leq 20$), suggesting that hyperbolic geometry alone is insufficient. While MERU enforces image-text entailment, it does not explicitly train concept-to-concept partial orders; HyCoCLIP's compositional cone training ensures that concept embeddings are radially ordered by specificity, which is the structure HypCBM's activation mechanism depends on.

Table 6: **Ablation of Backbone Geometry.** We evaluate HypCBM on SUN397 using different pre-trained backbones. CLIP embeddings are mapped to $\mathbb{L}^n$ via the exponential map. HyCoCLIP substantially outperforms MERU and CLIP, particularly in the sparse regime ($K \leq 20$), confirming the necessity of entailment-aligned embeddings.

			Accuracy @ Concept Budget (ANEC-K)					
Dataset	**Method**	**Backbone**	**5**	**10**	**20**	**50**	**100**	**Peak**
SUN397 ($N_c = 1935$)	HypCBM	CLIP-20M	32.3	35.9	42.9	51.7	54.2	67.3
	HypCBM	MERU-20M	23.1	28.3	37.5	51.1	58.5	69.7
	HypCBM	HyCoCLIP-20M	**49.3**	**58.3**	**65.7**	**69.9**	**71.3**	**72.3**

The results in Table 6 are achieved by significantly inflating the strictness parameter η_{img} to about twice the value used for HyCoCLIP, forcing most concepts to be activated by effectively reverting the model to a standard Euclidean LF-CBM (with non-negative activations) based mostly on angular separation, negating the benefits of the hierarchy.

A.3 Sparsity-Matched Evaluation

A potential confound in ANEC comparisons is that HypCBM's entailment-based activations are naturally sparse, whereas LF-CBM produces dense continuous activations. To verify that HypCBM's gains stem from geometrically meaningful activations rather than sparsity alone, we evaluate a sparsity-matched variant of LF-CBM (LF-CBM Matched) in which only the top-K highest concept activations are retained, matching HypCBM's average active concept count at each budget.

Table 7: **Sparsity-Matched Evaluation.** ANEC-K accuracy when LF-CBM activations are thresholded to retain only the top-K highest activations (LF-CBM Matched), controlling for sparsity. Enforcing sparsity on LF-CBM consistently degrades performance, confirming that HypCBM's gains arise from geometrically meaningful activations rather than sparsity alone.

		Sparsity Matched Accuracy @ K				
Dataset	**Method**	**5**	**10**	**20**	**50**	**100**
CIFAR-100	LF-CBM (CLIP-20M)	39.7	48.9	55.4	60.6	64.5
	LF-CBM Matched (CLIP-20M)	25.0	39.1	55.1	61.3	63.1
	HypCBM (HyCoCLIP-20M)	**44.3**	**54.4**	**60.9**	**69.7**	**71.8**
	Ref: LF-CBM (CLIP-400M)	*28.0*	*58.0*	*63.9*	*64.7*	*67.0*
SUN397	LF-CBM (CLIP-20M)	45.8	53.4	61.3	67.0	68.6
	LF-CBM Matched (CLIP-20M)	42.3	50.2	58.4	67.2	68.7
	HypCBM (HyCoCLIP-20M)	**49.3**	**58.3**	**65.7**	**69.9**	**71.3**
	Ref: LF-CBM (CLIP-400M)	*40.0*	*48.0*	*58.8*	*64.4*	*66.8*

Table 7 shows that enforcing sparsity on LF-CBM consistently *hurts* performance, particularly at low budgets (e.g., CIFAR-100 at $K = 5$: 39.7% $\rightarrow$ 25.0%). This confirms that the gains of HypCBM are not an artifact of activation sparsity: if sparsity were the primary driver, enforcing it on LF-CBM would help rather than hurt. Instead, HypCBM's sparse activations are geometrically meaningful — the concepts that activate are those for which the image genuinely falls within the entailment cone, whereas forcing sparsity on cosine similarity activations discards informative but lower-ranked concepts. HypCBM outperforms LF-CBM Matched by a wide margin across all budgets and both datasets.

A.4 Additional Intervention Results

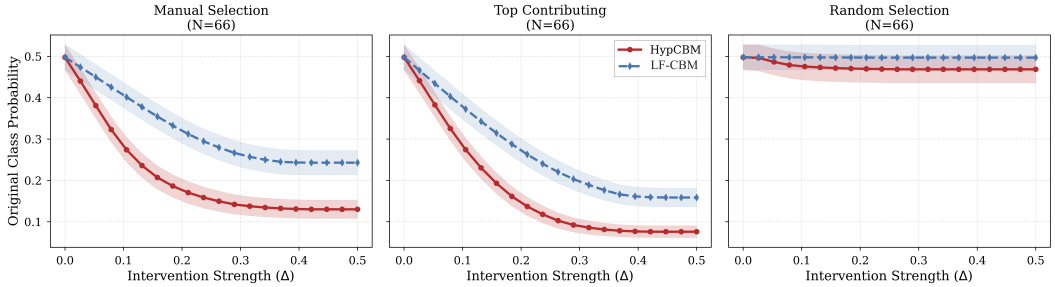


Figure 5: **Complete Intervention Analysis.** We compare the probability response of HypCBM (Red) and LF-CBM (Blue) across three distinct strategies. **Left (Manual):** We intervene on the concept with the highest contribution that is ground-truth absent (a false positive). HypCBM displays the sharpest confidence decay, indicating it is highly responsive to valid human corrections. **Center (Top-Contributing):** We intervene on the single highest contributing concept. The trend mirrors the manual setting, confirming utility for automated debugging. **Right (Random):** We intervene on random active concepts. Both models show negligible response, serving as a negative control; this confirms that the performance drops observed in the other settings are driven by semantic relevance, not model instability.

A.5 Data Efficiency

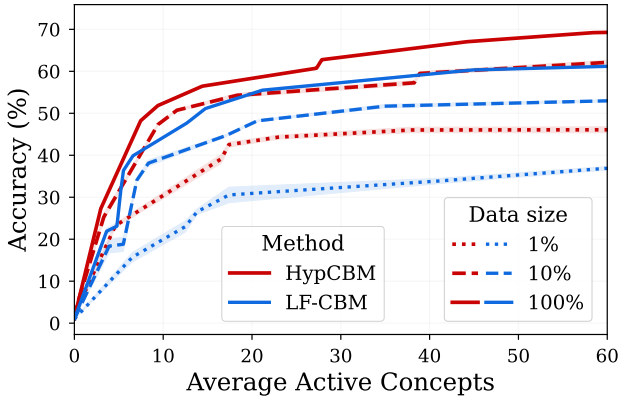


Figure 6: **Data efficiency on CIFAR100.** HypCBM outperforms LF-CBM (CLIP-20M) for any data budget.

Figure 6 shows ANEC curves on CIFAR-100 across training set sizes, averaged over 3 seeds with variance bands. A striking cross-regime comparison emerges: HypCBM trained on just 10% of the data matches LF-CBM trained on the full dataset across all concept budgets, demonstrating a 10× data efficiency advantage. At 100% training data, variance across seeds is negligible for both methods (std at ANEC-20: 0.09% for HypCBM, 0.13% for LF-CBM), confirming that main results are stable. At 1% data, LF-CBM shows substantially higher variance whereas HypCBM remains comparatively stable, suggesting that entailment-based geometric activations provide a more reliable inductive bias precisely when supervision is scarce.

A.6 Additional Hierarchical Consistency Results

Table 8 extends the hierarchical consistency analysis from the main paper by including two additional baselines: LF-CBM with a large-scale CLIP backbone (CLIP-400M) and a hierarchy-aware Euclidean CLIP model (CLIP^{RE}-400M [1]), both trained on 20× more data than HyCoCLIP. From this experiment, we gain three insights.

Table 8: **Hierarchical Consistency, SUN397 & ImageNet.** We measure the rate at which activating a child concept implies activation of its parent across N validated WordNet pairs. Despite a $20\times$ pre-training data disadvantage, HypCBM achieves the highest consistency at the strictest margin ($\eta = 1.2$) on both datasets, and on ImageNet already from $\eta \leq 1.3$.

	SUN397 ($N = 1503$)				**ImageNet** ($N = 829$)			
η_{img}	LF-CBM (CLIP-20M)	LF-CBM (CLIP-400M)	LF-CBM (CLIP^{RE}-400M)	**HypCBM (Ours) (HyCoCLIP-20M)**	LF-CBM (CLIP-20M)	LF-CBM (CLIP-400M)	LF-CBM (CLIP^{RE}-400M)	**HypCBM (Ours) (HyCoCLIP-20M)**
1.5	0.63	0.63	0.80	**0.84**	0.59	0.70	**0.82**	0.75
1.4	0.43	0.46	**0.62**	0.54	0.42	0.56	**0.68**	0.63
1.3	0.21	0.30	**0.36**	0.33	0.28	0.41	0.48	**0.51**
1.2	0.08	0.14	0.14	**0.17**	0.16	0.22	0.20	**0.33**

Data scale helps for consistency. Scaling from CLIP-20M to CLIP-400M improves consistency mainly at strict thresholds (e.g., 0.08 $\rightarrow$ 0.14 at $\eta = 1.2$ on SUN397), yet CLIP-400M remains below HypCBM at every threshold on both datasets, showing that data scale alone does not close the gap.

Hierarchical training helps more, but geometry still wins at strict budgets. CLIP^{RE}, which adds an entailment-aware training objective on top of the larger dataset, improves substantially over standard CLIP-400M, particularly at loose thresholds ($\eta \geq 1.4$). However, the advantage narrows at stricter thresholds: at $\eta = 1.2$, HypCBM matches or exceeds CLIP^{RE} on both datasets (SUN397: 0.17 vs. 0.14; ImageNet: 0.33 vs. 0.20), despite operating with $20\times$ less pre-training data. This crossover suggests that the hyperbolic manifold provides an inductive bias for hierarchical consistency that entailment-aligned training in Euclidean space cannot fully replicate.

Table 9: **Hierarchical Consistency: Activation Mechanism Ablation.** Using the *identical* HyCoCLIP-20M backbone on SUN397, replacing standard cosine similarity (LF-CBM) with our entailment activation (HypCBM) nearly doubles consistency, isolating the geometric activation mechanism as the primary source of the performance gains.

	SUN397 (HyCoCLIP-20M)	
η_{img}	LF-CBM	HypCBM
1.5	0.44	**0.84**
1.4	0.26	**0.54**
1.3	0.12	**0.33**
1.2	0.05	**0.17**

The activation mechanism is the primary driver. Table 9 isolates the contribution of our entailment activation by holding the backbone fixed. Using the identical HyCoCLIP-20M backbone, replacing cosine similarity with our margin-of-inclusion activation nearly doubles consistency at every threshold (e.g., 0.44 $\rightarrow$ 0.84 at $\eta = 1.5$, 0.05 $\rightarrow$ 0.17 at $\eta = 1.2$ on SUN397). Since the backbone is identical, these gains are exclusively attributable to the activation mechanism. Combined with the CLIP^{RE} comparison above, the evidence suggests that both components are necessary: hyperbolic geometry provides the representational structure, and the margin-of-inclusion activation reads it out effectively at inference time.

For reference, Table 10 reports accuracy for the hierarchy-aware Euclidean CLIP variant CLIP^{RE} [1][1]. A notable pattern emerges: despite being trained on $20\times$ more data, CLIP^{RE} performs substantially worse than standard CLIP-20M at strict concept budgets (e.g., SUN397 at K=5: 26.9% vs. 45.8%), suggesting that Euclidean entailment training trades sparse accuracy for hierarchical structure. HypCBM avoids this trade-off by grounding structure in the geometry of the manifold rather than the training objective, achieving both stronger consistency (Table 8) and higher accuracy at low budgets.

[1] CLIP^{RE} features are L2-normalized prior to computing cosine similarity.

Table 10: **Main Results.** Test accuracy at strict concept budgets (K). HypCBM consistently outperforms Euclidean baselines (LF-CBM) on identical backbones (HyCoCLIP-20M) and often even baselines that use large-scale pre-trained backbones (CLIP-400M and CLIP^{RE}-400M).

			Accuracy @ Concept Budget (ANEC-K)					
Dataset	**Method**	**Backbone**	**5**	**10**	**20**	**50**	**100**	**Peak**
CIFAR-100 ($N_c = 613$)	LF-CBM	CLIP-20M	39.7	48.9	55.4	60.6	64.5	67.9
	LF-CBM	HyCoCLIP-20M	25.6	34.6	56.3	61.6	62.6	69.5
	HypCBM (Ours)	HyCoCLIP-20M	**44.3**	**54.4**	**60.9**	**69.7**	**71.8**	**72.2**
	Ref: LF-CBM	*CLIP-400M*	*28.0*	*58.0*	*63.9*	*64.7*	*67.0*	*71.3*
	Ref: LF-CBM	*CLIP^{RE}-400M*	*20.4*	*26.0*	*31.7*	*46.6*	*58.3*	*70.2*
SUN397 ($N_c = 1935$)	LF-CBM	CLIP-20M	45.8	53.4	61.3	67.0	68.6	70.5
	LF-CBM	HyCoCLIP-20M	34.3	44.2	50.4	59.1	61.5	72.1
	HypCBM (Ours)	HyCoCLIP-20M	**49.3**	**58.3**	**65.7**	**69.9**	**71.3**	**72.3**
	Ref: LF-CBM	*CLIP-400M*	*40.0*	*48.0*	*58.8*	*64.4*	*66.8*	*79.0*
	Ref: LF-CBM	*CLIP^{RE}-400M*	*26.9*	*34.4*	*40.1*	*51.5*	*59.9*	*73.2*

A.7 Additional Robustness Results (CIFAR100-C)

In the main text, we analyzed concept stability at Severity 1. To ensure these findings are consistent across varying levels of image degradation, we evaluated both models across the full range of corruption severities (1–5) in the CIFAR-100-C benchmark.

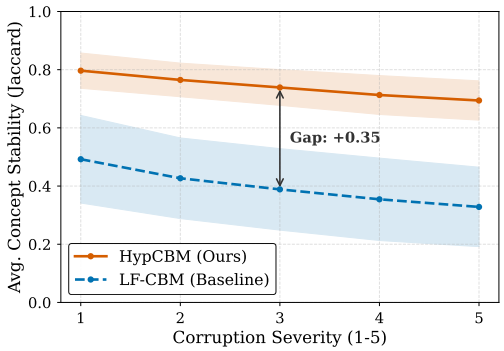


Figure 7: **Semantic stability across five severities of input corruption.** The plot shows the average Jaccard similarity across all 15 corruption types as a function of severity. The shaded regions represent one standard deviation. The baseline (LF-CBM) shows a rapid decrease in concept stability as severity increases, while HypCBM maintains high stability ($J > 0.7$) even at Severity 5. The stability gap remains consistently large (~ 0.35) throughout.

Figure 7 presents the aggregated results. We observe a consistent gap between the hyperbolic and Euclidean models. Notably, at Severity 3 (the tipping point for accuracy), the gap in concept stability is approximately 0.35, confirming that HypCBM's explanations are significantly more robust to input perturbations.

Threshold-free stability. Since the Jaccard metric requires binarizing LF-CBM's continuous activations, we also compute the Pearson correlation (centered cosine similarity) between clean and corrupted activation vectors. Centering is necessary because uncentered cosine similarity is dominated by the baseline shared across all concepts: the fraction $f = n\bar{v}^2/\|\mathbf{v}\|_2^2$ of each activation vector attributable to this baseline is 0.915 for LF-CBM but only 0.198 for HypCBM, which inflates LF-CBM's uncentered similarity (0.967 vs. 0.893) without reflecting a more stable concept pattern, consistent with observations on CLIP embeddings [4]. With the centered metric, HypCBM is more stable on 15/15 corruptions at severity 3 (mean 0.866 vs. 0.606).

A.8 Hyperbolic Norm Filtering

We evaluate the effect of filtering low-norm concepts, as introduced in Section 3, on the SUN397 dataset. In hyperbolic representations, embedding norm correlates with specificity; points near the origin represent broad, generic concepts (e.g., "object") with naturally wide entailment cones.

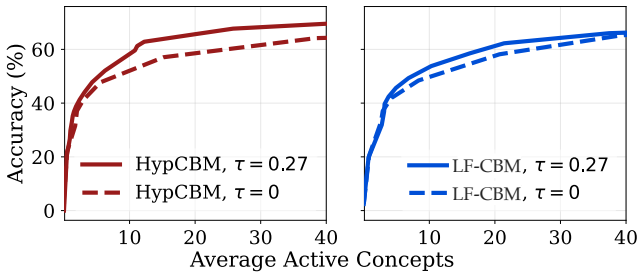


Figure 8: **Ablation on Hyperbolic Norm Filtering, SUN397.** $\tau = 0.27$**.**

As shown in Figure 8, filtering these generic concepts ($\tau = 0.27$) lifts the accuracy-sparsity frontier for both models, with HypCBM profiting disproportionally. Because our entailment mechanism links cone width to embedding norm, concepts near the origin have excessively wide apertures that activate indiscriminately, hurting efficiency. Filtering these trivial activations allows HypCBM to recover substantial gains. The Euclidean LF-CBM, in contrast, relies on angular similarity; pruning general concepts does not lead to a comparable reduction in trivial activations.

B Concept Bank

As the concept banks for CIFAR100, ImageNet and CUB-200 were already created and made public by Oikarinen et al. [33], we only apply the concept bank creation process for SUN397. We use the following prompts for each class in the dataset, as introduced by Oikarinen et al. [33]:

1. Important Features

```
"List the most important features
for recognizing something as a
'goldfish':
- bright orange color
- a small, round body
- a long, flowing tail
- a small mouth
- orange fins

List the most important features
for recognizing something as a
'beer glass':
- a tall, cylindrical shape
- clear or translucent color
- opening at the top
- a sturdy base
- a handle

List the most important features
for recognizing something as a
'{}':"
```

2. Superclass

```
"Give superclasses for the word
'tench':
- fish
- vertebrate
- animal

Give superclasses for the word
'beer glass':
- glass
- container
- object

Give superclasses for the word
'{}':"
```

3. Context (Around)

```
"List the things most commonly
seen around a 'tench':
- a pond
- fish
- a net
- a rod
- a reel
- a hook
- bait

List the things most commonly
seen around a 'beer glass':
- beer
- a bar
- a coaster
- a napkin
- a straw
- a lime
- a person

List the things most commonly
seen around a '{}':"
```

Figure 9: **LLM Prompts for Concept Generation.** We use three distinct prompt templates (Important Features, Superclass, Context) to generate diverse visual attributes. These few-shot examples are fed to GPT-3 to produce the raw concept bank.

After this initial set of candidate concepts is created, a few processing steps are applied. First, concepts that are too long (longer than 30 tokens) are removed. Then, we calculate cosine similarity between all class labels and all concepts and remove those with a similarity > 0.85. Finally, the same thing is done for the concepts themselves; any concept with a similarity of > 0.9 to another concept is removed (the concept with lower average similarity to other concepts is deleted). This process

results in a different concept bank size, N_c, for each dataset, mostly dependent on how many classes they contain, and how much redundancy across generated concepts is present.

C Calibration of Entailment Strictness Parameters η_{text} and η_{img}.

C.1 Finding the Optimal Inter-Modal η_{img}

Because standard image datasets lack exhaustive, multi-level hierarchical annotations, we leverage the fundamental premise of aligned multimodal spaces: semantic entailment should be modality-agnostic. In a well-aligned shared manifold, if a specific text (e.g., "Persian cat") entails a general text ("cat"), an image depicting that specific concept should geometrically entail the general concept as well. Thus, we utilize WordNet's text-text hypernym chains as a proxy to calibrate the image-text threshold η_{img}.

To determine a robust value for the entailment strictness parameter η_{img}, we treat the calibration as a binary classification problem: distinguishing "true" semantic entailments from non-entailing pairs based on their geometric entailment ratio:

$$R = \frac{\phi(\mathbf{z}, \mathbf{c})}{\omega(\mathbf{c})}.$$

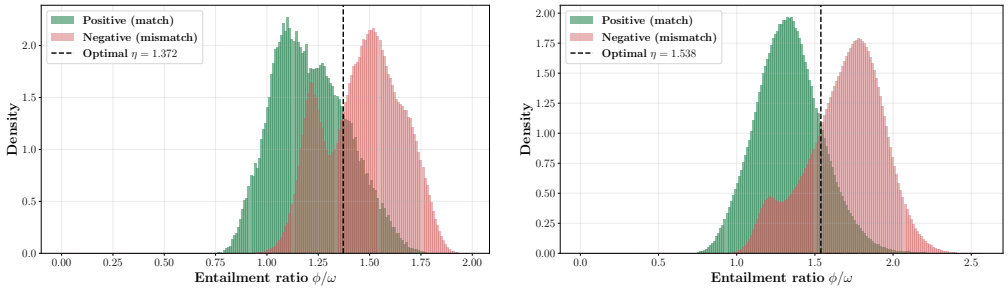
Since η_{img} acts as the decision boundary ($R \leq \eta_{img}$), we seek the threshold that optimally separates these two classes. We construct the following two proxy datasets:

- **Positive Set**: A set of ground-truth entailment pairs derived from WordNet hypernym chains (e.g., Persian cat $\rightarrow$ cat), filtered to ensure the parent concept has a lower hyperbolic norm than the child.
- **Negative Set**: A set of randomly sampled concept pairs with no known semantic relationship, serving as a baseline for non-entailment.

Then, we compute the entailment ratio R for all pairs in both sets. We then sweep across possible values of η_{img} to calculate the True Positive Rate (Sensitivity, or the fraction of WordNet pairs correctly identified as entailed ($R \leq \eta_{img}$)) and the False Positive Rate ($1-$ specificity: the fraction of random pairs incorrectly identified as entailed, $R \leq \eta_{img}$).

To select the optimal η_{img}, we maximize Youden's J statistic (J):

$$J(\eta_{img}) = \text{Sensitivity}(\eta_{img}) + \text{Specificity}(\eta_{img}) - 1$$



(a) CIFAR100 (b) ImageNet

Figure 10: Image-text entailment distributions. The plot shows the entailment ratio ϕ/ω, or the minimum η_{img} required for entailment. The dotted line shows Youden's J statistic, or the optimal η_{img} that separates the positive and negative distributions. The overlap in the two distributions shows that false positives and negatives are intrinsic to the backbone's pre-trained geometry, which we mitigate by selecting the threshold η_{img} that maximizes separation.

This statistic captures the optimal trade-off point where the model maximizes the recovery of true semantic hierarchies while minimizing spurious activations from unrelated concepts.

We find the optimal η_{img} values to be 1.372 for CIFAR100, and $\eta_{img} = 1.538$ for Imagenet. We round these to 1.4 and 1.5 in our experiments.

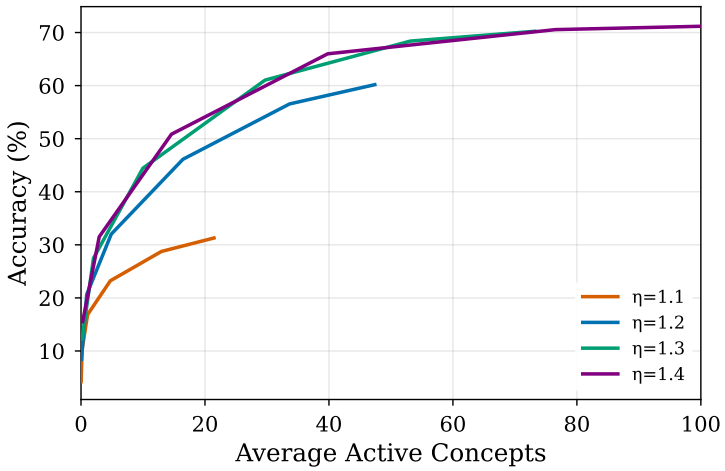


Figure 11: **Accuracy vs. number of active concepts, SUN397.** We sweep η_{img} on a validation set to determine the optimal value on datasets where the distribution shift between proxy class labels and the concept bank is large.

SUN397 & CUB-200 For the SUN397 and CUB-200 datasets, this method of selecting the optimal η_{img} resulted in poor performance, likely due to a larger distribution shift between the proxy class labels and the used concept bank. As an alternative option for datasets like these, we tune η_{img} by sweeping a range of candidate values, and analyzing the performance on a validation split of the training set. We select $\eta_{img} = 1.4$ and $\eta_{img} = 1.5$ for SUN397 and CUB-200 respectively. See Fig. 11 for an example on SUN397.

C.2 Finding the Optimal Intra-Modal η_{text}

Experimental Setup. Due to the exponential expansion of volume in hyperbolic space, the aperture required to capture a semantic subtree varies drastically with hierarchy depth. A fixed η_{text} that is sufficient for general concepts (low norm) is often too narrow for specific concepts (high norm), leading to false negatives in entailment detection.

To derive a robust scaling law for concept-to-concept entailment, we performed a controlled experiment using WordNet hierarchies as a ground-truth reference. We initiate a crawler from 15 diverse mid-level roots (e.g., *mammal*, *tool*, *structure*) to generate candidate parent-child pairs, across 3 depths.

Evaluation metrics. To ensure the derived statistics were applicable to our trained model, we applied a few filters to the candidate pairs:

- **Norm Consistency**: We kept only pairs where both parent and child embeddings had a norm of $\|\tilde{\mathbf{c}}\| > \|\tilde{\mathbf{c}}_{min}\|$, in our case 0.27.
- **Hierarchical Integrity**: Secondly, we filtered for pairs satisfying $\|\tilde{\mathbf{c}}_{parent}\| < \|\tilde{\mathbf{c}}_{child}\|$ ensuring the parent is closer to the origin than its descendant.

For each of the $N = 4504$ valid pairs, we compute the required entailment ratio (η_{text}), defined as the scalar multiplier required for the parent's cone to strictly contain the child:

$$\eta_{\text{text}} = \frac{\phi(\mathbf{c}_{\text{child}}, \mathbf{c}_{\text{parent}})}{\omega(\mathbf{c}_{\text{parent}})} \tag{11}$$

where ϕ is the exterior angle, and ω is the half-aperture.

Empirical Scaling Law. Figure 2b in the main text illustrates the relationship between concept specificity (proxied by the hyperbolic norm $\|\mathbf{c}\|$) and the required threshold η_{text}. We observe a

positive correlation ($r = 0.729$), confirming that deeper concepts require wider entailment cones relative to their base aperture.

We fit a linear regression model to this distribution, yielding the relationship:

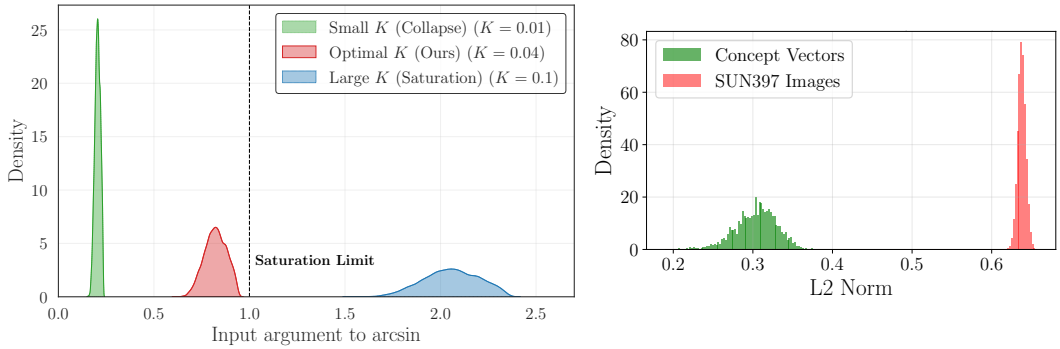
$$\eta_{text}(\|\tilde{\mathbf{c}}\|) \approx 8.62 \cdot (\|\tilde{\mathbf{c}}\| - 0.09). \tag{12}$$

D Cone parameterization and Distribution of Concept and Image Embeddings

Our half-aperture definition:

$$\omega(\mathbf{c}_i) = \sin^{-1}\left(\frac{2K}{\sqrt{c}\,\|\tilde{\mathbf{c}}_\mathbf{i}\|}\right)$$

depends inversely on the spatial concept embedding norm $\|\tilde{\mathbf{c}}_\mathbf{i}\|$. This formulation renders the aperture sensitive to the scale of the embeddings: if the argument $u_i = \frac{2K}{\sqrt{c}\,\|\tilde{\mathbf{c}}_\mathbf{i}\|}$ exceeds 1, the cone angle becomes undefined (or, effectively, fully saturated at $\pi/2$). To ensure semantically meaningful cones, we carefully calibrate the scaling factor K against the empirical norm distribution of the concept bank.



(a) **Aperture Calibration.** (b) **Modality Gap.**

Figure 12: **Geometric Properties and Calibration. (a)** We calibrate the cone scaling factor K to ensure that the distribution of concept apertures is well-posed (i.e. all arguments to arcsin are smaller than 1), avoiding numerical saturation limits (dashed line). For the default value of $K = 0.1$, we observe that all possible text embeddings are clipped to 1, leading to a constant half-aperture of $\omega(\mathbf{c_i}) = \frac{1}{2}\pi$, $\forall i$. **(b)** We visualize the native norm distribution of the backbone's representations, which informs our choice of filtering thresholds and scaling parameters. This plot highlights the *modality gap*, where representations for the text and image modalities are clearly separated by norm.

Calibration and Saturation. As shown in Figure 12b, the backbone exhibits a distinct *modality gap*, where text-derived concept embeddings possess significantly smaller norms than image embeddings. This low-norm regime puts concepts at risk of saturation. Figure 12a illustrates this sensitivity. With a standard scaling of $K = 0.1$ (blue), the argument u_i exceeds the saturation limit (1.0) for the entire concept bank. This forces the half-aperture to clip to $\pi/2$, effectively treating all concepts as "universal" and destroying the hierarchical granularity of the bottleneck. Conversely, a very small $K = 0.01$ (green dotted line) collapses the distribution near zero, resulting in "needle" cones with no entailment capacity.

We select $K = 0.04$ (red) to balance this trade-off. For our filtered concept bank (where we enforce $\|\tilde{\mathbf{c}}_\mathbf{i}\| \geq 0.27$), this choice ensures that the aperture arguments are widely distributed across the valid range $[0, 1)$, maximizing geometric expressivity. Consequently, the argument to the arcsine function is maximized at the lower bound of our filter, $\|\tilde{\mathbf{c}}_\mathbf{i}\| = 0.27$, which yields:

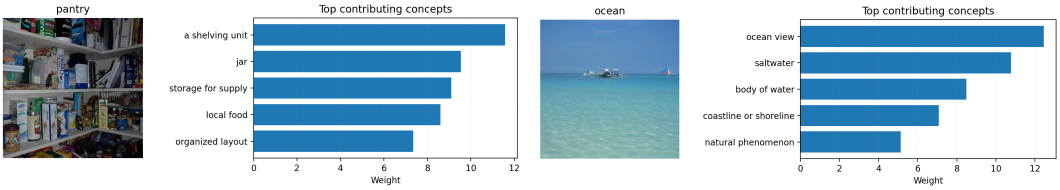
$$u_{\max} \approx 0.937$$

staying safely below the singularity at 1.0. As a final safeguard for numerical stability, our implementation clamps the argument prior to the arcsin operation:

$$\tilde{u}_i = \text{clip}\left(u_i, -1 + \epsilon,\, 1 - \epsilon\right), \quad \epsilon = 10^{-6}$$

ensuring that $\omega(\tilde{\mathbf{c}}_i)$ remains well-defined regardless of any potential outlier norms or floating-point variations. This clamping logic also applies for the input to the $\arccos$ for the definition of the exterior angle (Eq. 6). Together, the analytic constraint on K and the norm-based filtering step ensure that the cone parameterization is numerically stable and semantically diverse.

E Qualitative Examples



(a) Pantry (b) Ocean

Figure 13: **Global Explanations.** We visualize the top contributing concepts (weight $\times$ activation) for the classes 'pantry' and 'ocean', along with two sample images from SUN397. More examples comparing LF-CBM and HypCBM are in the supplementary material.

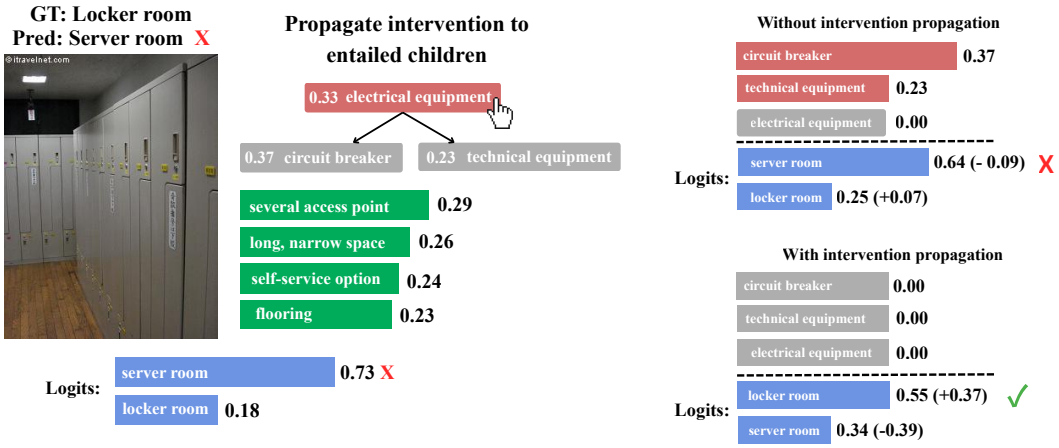


Figure 14: **Intervention Propagation.** An example that shows our intervention propagation mechanism on an image of a locker room that is misclassified as 'server room'. When intervening on 'electrical equipment', HypCBM automatically intervenes on entailed children 'circuit breaker' and 'technical equipment' too. Without this propagation, the prediction is still wrong, whereas with propagation the prediction flips to the correct class, 'locker room'.

F Implementation Details

F.1 ANEC Interpolation and Pareto Frontier Estimation

To accurately estimate the Accuracy at Number of Effective Concepts (ANEC) for specific integer budgets K, we employ piecewise linear interpolation between measured operating points. To ensure the fidelity of this approximation, we utilize an adaptive grid refinement strategy. We first perform a coarse sweep of regularization strengths $\lambda \in [10^{-1}, 10^{-7}]$ on a logarithmic scale. We then iteratively refine the grid by sampling additional λ values in regions where the change in the number of active concepts is large, ensuring that the accuracy-sparsity curve is densely sampled near the target budgets (e.g., K=5,10,20).

F.2 Training Time

The computation of the entailment activation matrix scales linearly with the number of samples N, the concept bottleneck size M, and the embedding dimension D ($D = 512$ for HyCoCLIP), resulting

in a time complexity of $\mathcal{O}(NMD)$. Specifically, the pairwise Lorentz exterior angle $\phi(\mathbf{z}, \mathbf{c})$ relies on the Minkowski dot product, which requires the same number of floating-point operations (FLOPs) as the standard Euclidean dot product used in linear classifiers.

To maintain memory efficiency, we use batch size $B = 512$, ensuring that the peak GPU memory usage scales as $\mathcal{O}(BM)$ rather than $\mathcal{O}(NM)$. This allows HypCBM to scale to thousands of concepts (e.g., ImageNet, $M = 2950$) on consumer-grade hardware. Consequently, the primary computational bottleneck is not the geometric activation, but the optimization of the sparse linear classifier. Training time for CIFAR100 is approximately 2-3 minutes, whereas this scales from ≈ 30 minutes on SUN397 to ≈ 5 hours for ImageNet, depending on the regularization strength, λ. For very low values of λ ($\lambda < 1e-7$), the linear solver takes significantly longer.

Due to the numerical sensitivity of hyperbolic operations in the Lorentz model, we perform all geometric computations involving hyperbolic embeddings in double precision (FP64).

G Comprehensive CBM Taxonomy

Table 11: **Comprehensive taxonomy of CBM methodologies.** While recent methods successfully enforce bottleneck structure, they rely on task-specific training, graph priors, or auxiliary supervision (e.g., bounding boxes or logical rules). HypCBM is the only framework that achieves a structured bottleneck completely post-hoc, requiring no learned modules or auxiliary supervision beyond the pre-trained manifold.

Method	Post-hoc	Zero-shot	Structured	Backbone	Auxiliary Supervision	Learned Modules
LaBo [55]	✓	✓	✗	Any VLM	Concept Optimization	Optimization-based selection
CEM [57]	✗	✗	✗	Task-specific	Concept Labels	Soft concept embeddings
VLG-CBM [47]	✗	✗	✗	Any VLM	Object Bounding Boxes	Learned CBL projection
DN-CBM [42]	✗	✗	✗	CLIP [39]	**None**	Sparse Autoencoder
Hierarchical CBM [37]	✗	✗	✓	Task-specific	Hierarchy Labels	Supervised hierarchy layers
C2F-CBM [35]	✗	✗	✓	Task-specific	Concept Labels	Probabilistic modules
LogicCBM [50]	✗	✗	✓	Task-specific	Logical Rules	Neuro-symbolic reasoning
Relational CBM [3]	✗	✗	✓	Task-specific	Relational Graphs	Relational graph/weights
Graph CBM [53]/CRM [6]	✗	✗	✓	Task-specific	Relational Graphs	Graph reasoning
PCBM [56] / LF-CBM [33]	✓	✓	✗	Any VLM	**None**	**None**
HypCBM (Ours)	✓	✓	✓	**Hyperbolic VLM**	**None**	**None**

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The claims we make are accurately supported through extensive experimentation.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We discuss the limitations of our work in the Conclusion; reliance on hyperbolic VLMs and limited performance on fine-grained tasks.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [N/A]

Justification: We do not have theoretical results requiring proofs.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: Our code and all other experimental details are available.

   Guidelines:

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

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

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

Answer: [Yes]

Justification: Code is provided in the Supplementary Material. All data and model checkpoints are open access.

Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: These details are presented in Appendix F.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [No]

   Justification: Main results (Table 2) report single-run results. This is justified because ANEC curves aggregate over the full ElasticNet regularization path ($\sim$30 operating points) on frozen activations, leaving random seed as the only source of variance. The data efficiency experiment (Table 3) reports results averaged over 3 random seeds, with variance bands across seeds provided in Appendix A.5. At 100% training data, standard deviation at ANEC-20 is 0.09% for HypCBM and 0.13% for LF-CBM, confirming negligible variance in the main results regime. Additionally, experiments on ImageNet are computationally expensive.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Presented in Appendix F

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We comply with the code of ethics in every respect.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [No]

    Justification: We believe our work primarily has positive societal impacts by advancing the domain of interpretable image classification and ensuring transparency in black-box models. Therefore, we did not include a dedicated discussion of negative societal impacts, as our contributions represent foundational algorithmic improvements to model interpretability rather than specific downstream applications with foreseeable direct harms.

Guidelines:

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

11. **Safeguards**

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

    Answer: [N/A]

    Justification: We only use publicly available datasets without sensitive information or are susceptible to misuse (CUB, CIFAR10/100, ImageNet, SUN397).

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: We properly credit the creators of all utilized pre-trained models (OpenAI CLIP, MERU, HyCoCLIP) and codebase foundations (PCBM, LF-CBM) by citing their respective papers. All of these assets are open-source (primarily under the MIT License), and we adhere strictly to their terms of use, documenting this in our supplementary material. Links to GitHub repositories built upon:

    - PCBM (MIT License): https://github.com/mertyg/post-hoc-cbm
    - LF-CBM (MIT License): https://github.com/Trustworthy-ML-Lab/Label-free-CBM

- HyCoCLIP (CC BY-NC 4.0): https://github.com/PalAvik/hycoclip
- MERU (CC BY-NC 4.0): https://github.com/facebookresearch/meru
- CLIP (MIT License): https://github.com/openai/CLIP

We credit all datasets in the References section.

Guidelines:

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

13. **New assets**

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

Answer: [N/A]

Justification: We do not introduce new assets.

Guidelines:

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

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

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

Answer: [N/A]

Justification: The paper does not involve crowdsourcing.

Guidelines:

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

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

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

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [Yes]

Justification: We use an LLM to generate our concept banks, according to the methodology of Oikarinen et al. [33].

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.