Contents

A. Extended Related Work

Interpretable-by-design models. Interpretable-by-design models aim to provide explanations for their predictions by using human-interpretable concepts as intermediate representations. Early works explored attribute-based classification for face verification [34] and learning to detect unseen object classes through attribute transfer [36]. Subsequent works include Concept Activation Vectors [28], which use linear classifiers to identify directions in the embedding space corresponding to specific concepts. Concept Bottleneck Models [29] extend this idea by training a model to predict concepts before predicting the final output. In an adjacent line of work, Information Pursuit [20] is used as a criterion to choose the most relevant concepts [7, 9, 30]. More recent works have explored leveraging pre-trained embeddings and sparse coding for identifying specific concept directions [4, 8, 19]. Our work builds upon these foundations by introducing a concept embedding framework that captures the hierarchical relationships among synsets in interpretable image classification.

Sparse Recovery. Sparse recovery aims to recover a sparse signal from a set of observations, often using techniques such as Orthogonal Matching Pursuit (OMP) [54] and Basis Pursuit [11]. Sparse coding has been widely used in image processing [43, 44], signal processing, and machine learning. Although there have been works on hierarchical sparse coding [25, 27, 35], they do not consider the hierarchical structure of concepts in the context of interpretable models or deep representation learning. More recently, sparse autoencoders (SAEs) [6, 13, 48] have been used in vision-language models [12, 50, 70] to interpret the structure of concepts beyond sparsity. Unlike SAE-based approaches, our work focuses on structured sparse coding with an explicitly hierarchical dictionary derived from semantic synset relationships.

B. Preliminaries

B.1. Interpretable-by-design vs. Chain-of-Thought

Chain-of-Thought (CoT) prompting [67] is a technique that enables large language models to generate step-by-step reasoning traces before producing a final answer, often improving performance on complex reasoning tasks. However, using CoT as an interpretability method does not guarantee faithfulness [5] because the final prediction is conditioned on both the input and the generated chain of thoughts. Several recent audits further question the faithfulness of CoT explanations [3, 38, 65]. Therefore, the development of CoT does not make interpretable-by-design models obsolete. In fact, there is a trend toward making foundation models (e.g., LLMs or diffusion models) interpretable-by-design by enforcing an interpretable concept bottleneck in the latent space [23, 60].

[5] An explanation is *faithful* if it accurately reflects the true computation process to the final prediction [24].

B.2. Canonical regular simplex

Define

$$\tilde{\boldsymbol{s}}_j \;=\; \boldsymbol{e}_j \;-\; \frac{1}{b}\,\mathbf{1}, \qquad j = 1, \ldots, b, \tag{9}$$

where $\{\boldsymbol{e}_j\}_{j=1}^{b}$ are the standard basis vectors of $\mathbb{R}^b$ and $\mathbf{1} \in \mathbb{R}^b$ is the all-ones vector. These centred vertices satisfy $\sum_{j=1}^{b} \tilde{\boldsymbol{s}}_j \;=\; \mathbf{0}$ and $\tilde{\boldsymbol{s}}_j^{\top} \tilde{\boldsymbol{s}}_k \;=\; \begin{cases} 1, & j = k, \\ -\dfrac{1}{b-1}, & j \neq k, \end{cases}$ i.e. they form a regular $(b-1)$-simplex of unit edge length in $\mathbb{R}^{b-1}$.

C. Limitations

While our framework demonstrates clear advantages in concept recovery, it also has several limitations:

Embedding Constraints. Our theoretical analysis (Prop. 3.3) establishes that embedding a hierarchy with leaf depth L and branching ratio b requires ambient dimension $d \geq L + b - 1$. For deep hierarchies (large L) or highly branching structures (large b), this constraint becomes restrictive. Real-world embeddings from models such as CLIP typically have fixed dimensions (e.g., $d = 768$), which limits the depth and complexity of hierarchies that can be faithfully represented. Moreover, as hierarchies deepen, the half-angles of the cones containing each subtree (Prop. 3.2) must decrease geometrically. As mentioned in §3.1, this limitation may necessitate exploring alternative geometries, such as hyperbolic spaces [15, 49], for more faithful hierarchical representations. At the same time, by using pretrained embeddings, HCEP provides accurate and interpretable classification without the additional compute required to finetune large models, and pretrained models are available and improving across multiple domains, making HCEP easy to extend. That said, hierarchy-aware finetuning could further improve the geometric conditions in §3 and thus boost HCEP's performance; we leave this direction to future work.

Hierarchy Quality Dependence. The performance of Hierarchical OMP critically depends on the quality of the predefined hierarchy. For ImageNet-based datasets, we leverage the well-curated WordNet hierarchy, which provides semantically meaningful relationships. However, for CIFAR-100, we rely on taxonomy induction methods [71], which may produce hierarchies with inconsistencies or unclear relationships. That said, high-quality hierarchies already exist in many domains beyond common objects, e.g., RadLex [37] for radiology, DERM12345 [69] for skin lesions, and iNaturalist [66] for species classification. Moreover, the quality of LLM-based taxonomy induction is steadily improving, as demonstrated by our CIFAR-100 experiments (§5.2). When the hierarchy is noisy, HCEP may produce incorrect explanation paths. A promising direction for future work is to interpolate between hierarchical and non-hierarchical solutions via ℓ_1 minimization with a hierarchy regularizer [25], thereby allowing the method to degrade gracefully when hierarchy quality is uncertain.

Computational Complexity. Hierarchical OMP with beam search (Alg. 2) has complexity $O(TBb|\mathcal{D}_{\text{active}}|)$, where T is the number of iterations, B is the beam width, b is the branching ratio, and $|\mathcal{D}_{\text{active}}|$ is the size of the active dictionary at each level. In contrast, OMP has complexity $O(T|\mathcal{D}|)$, where $|\mathcal{D}|$ is the size of the entire dictionary. For large branching ratios or deep hierarchies, this may become computationally expensive. While we demonstrate better concept recovery accuracy over standard OMP, the computational cost remains a practical consideration for deployment at scale. In practice, the beam search hypotheses can be parallelized on GPU, which significantly reduces wall-clock time overhead (see Fig. 10).

Outlier Handling. HCEP assumes that the input belongs to a leaf class in the provided hierarchy. For outliers within a known class (e.g., a cat with three legs), classical results on the robustness of sparse coding to corruption [68] suggest that the outlier will still be closest to the correct synset path, while a larger sparse reconstruction error can flag such cases. If a novel class is entirely absent from the hierarchy, taxonomy induction can be used to append the new class before applying HCEP.

D. Proofs

D.1. Proof of Prop. 3.1

Statement. If subtree containment (Eq. (3)) and sibling-cone disjointness (Eq. (4)) hold, then the subtrees rooted at sibling nodes do not overlap.

Proof. Suppose j and j' are distinct children of a parent node i. We show that their subtrees are disjoint.

Let $k \in \operatorname{desc}(j)$. By subtree containment in Eq. (3),

$$\angle(\boldsymbol{a}^{(j)}, \boldsymbol{a}^{(k)}) \leq \theta_{\text{lev}(j)}. \tag{10}$$

Assume for contradiction that $k \in \operatorname{desc}(j')$ as well. Applying Eq. (3) to j' gives

$$\angle(\boldsymbol{a}^{(j')}, \boldsymbol{a}^{(k)}) \leq \theta_{\text{lev}(j')}. \tag{11}$$

Now consider the spherical triangle formed by the unit vectors $\boldsymbol{a}^{(j)}/\|\boldsymbol{a}^{(j)}\|$, $\boldsymbol{a}^{(k)}/\|\boldsymbol{a}^{(k)}\|$, and $\boldsymbol{a}^{(j')}/\|\boldsymbol{a}^{(j')}\|$. The triangle inequality yields

$$\begin{aligned} \angle(\boldsymbol{a}^{(j)}, \boldsymbol{a}^{(j')}) &\leq \angle(\boldsymbol{a}^{(j)}, \boldsymbol{a}^{(k)}) + \angle(\boldsymbol{a}^{(k)}, \boldsymbol{a}^{(j')}) \\ &\leq \theta_{\text{lev}(j)} + \theta_{\text{lev}(j')}, \end{aligned} \tag{12}$$

which contradicts sibling-cone disjointness in Eq. (4). Therefore k cannot belong to both $\operatorname{desc}(j)$ and $\operatorname{desc}(j')$.

Since j and j' were arbitrary siblings, the subtrees rooted at sibling nodes are disjoint. □

D.2. Proof of Prop. 3.2

Statement. If the half-angles satisfy $\theta_{l+1} \leq \min\{r, 1/b\}\,\theta_l$ with $r \in (0, 1/2)$, then there exists a placement of the nodes such that *subtree containment* (Eq. (3)) and *sibling-cone disjointness* (Eq. (4)) hold.

Proof. We verify the two requirements separately.

Step 1: Subtree containment. A sufficient condition for Eq. (3) is that the cumulative half-angles of all lower levels do not exceed the half-angle budget at level l, namely

$$\sum_{k=l+1}^{L} \theta_k \leq \theta_l, \quad \forall i \in \{1, \dots, N_L\},\ l = \mathrm{lev}(i). \tag{13}$$

Assume first that the half-angles satisfy the geometric decrease

$$\theta_{l+1} \leq r\,\theta_l. \tag{14}$$

Then, for any level l,

$$\sum_{k=l+1}^{L} \theta_k \leq \sum_{k=1}^{L-l} \theta_l\, r^k \text{(by Eq. (14))} \tag{15}$$

$$= \theta_l \sum_{k=1}^{L-l} r^k \tag{16}$$

$$= \theta_l\, r \sum_{k=0}^{L-l-1} r^k \tag{17}$$

$$\leq \theta_l\, r\, \frac{1 - r^{L-l-1}}{1 - r} \text{(sum of a geometric series).} \tag{18}$$

Taking $L \to \infty$ yields

$$\theta_l\, r \frac{1 - r^{L-l-1}}{1 - r} \to \theta_l\, \frac{r}{1 - r} \leq \theta_l, \qquad r < \frac{1}{2},$$

so Eq. (13) holds. This proves subtree containment.

Step 2: Sibling-cone disjointness. Next, we show that Eq. (4) follows from the simpler sufficient condition

$$\theta_{l+1} \leq \frac{1}{b}\,\theta_l. \tag{19}$$

Consider any 2D plane containing the parent cone axis. In that plane, a cone of half-angle α appears as a planar angle of magnitude 2α. Therefore, one can place b child cone axes inside the parent cone without overlap whenever b child angles of size $2\theta_{l+1}$ fit inside the parent angle $2\theta_l$, i.e., whenever $b\theta_{l+1} \leq \theta_l$. This is exactly Eq. (19).

Since the assumption of the proposition is $\theta_{l+1} \leq \min\{r, 1/b\}\theta_l$, both Step 1 and Step 2 hold simultaneously. Hence there exists a placement satisfying subtree containment and sibling-cone disjointness. □

D.3. Proof of Prop. 3.3

Statement. Under the hierarchical orthogonality constraints in Eq. (30) and the regular-simplex difference condition in Eq. (31) at every internal node up to depth $L-1$ of a hierarchy whose leaves are at depth L in ambient space $\mathbb{R}^d$, it is necessary that the ambient dimension satisfies the depth–dimension condition $d \geq L + b - 1$.

Proof. Fix a level $l \geq 0$ and consider the ancestor path $\boldsymbol{A}_l = \{\boldsymbol{a}^{(\pi_0)}, \dots, \boldsymbol{a}^{(\pi_l)}\}$. The hierarchical orthogonality constraints in Eq. (30) define the affine feasible set for child candidates:

$$\mathcal{V}_l = \{\boldsymbol{x} \in \mathbb{R}^d : \boldsymbol{A}_l^\top \boldsymbol{x} = \boldsymbol{h}_l\},$$

which is Eq. (34). When the ancestor vectors are linearly independent, as is generically the case because each level introduces a new non-collinear direction, we have

$$\dim \mathcal{V}_l = d - (l + 1).$$

Now apply the regular-simplex condition in Eq. (31). It requires placing b child points whose differences relative to a feasible origin in $\mathcal{V}_l$ form a regular $(b-1)$-simplex. Since such a simplex has affine hull of dimension $b-1$, it can be embedded in $\mathcal{V}_l$ only if

$$\dim \mathcal{V}_l \ \geq\ b - 1.$$

Combining the two displays yields, for every level l, $d - (l+1) \geq b - 1 \iff d \geq l + b$. Requiring this inequality to hold at the deepest internal level $l = L - 1$ gives $d \geq (L-1) + b = L + b - 1$, as claimed. □

D.4. Intermediate results for Prop. 4.1

Lemma D.1 (Column normalization equivalence)**.** *Let $\boldsymbol{D} = [\boldsymbol{d}_1, \dots, \boldsymbol{d}_k] \in \mathbb{R}^{d \times k}$ with arbitrary nonzero column norms ($\|\boldsymbol{d}_j\|_2 > 0$ for all $j \in [k]$), and define the diagonal matrix $\boldsymbol{W} := \mathrm{diag}(\|\boldsymbol{d}_1\|_2, \dots, \|\boldsymbol{d}_k\|_2)$ and the column-normalized dictionary $\widehat{\boldsymbol{D}} := \boldsymbol{D}\,\boldsymbol{W}^{-1}$. For any s-sparse $\boldsymbol{z} \in \mathbb{R}^k$ with support S, set $\widehat{\boldsymbol{z}} := \boldsymbol{W}\boldsymbol{z}$. Then $\boldsymbol{x} = \boldsymbol{D}\boldsymbol{z} = \widehat{\boldsymbol{D}}\,\widehat{\boldsymbol{z}}$ and $\mathrm{supp}(\widehat{\boldsymbol{z}}) = S$. Moreover, OMP run on $\boldsymbol{D}$ with the selection rule*

$$j^\star \in \arg\max_j \frac{\left|\langle \boldsymbol{r}, \boldsymbol{d}_j \rangle\right|}{\|\boldsymbol{d}_j\|_2} = \arg\max_j \frac{\left|\langle \boldsymbol{r}, \boldsymbol{d}_j \rangle\right|}{\|\boldsymbol{d}_j\|_2\, \|\boldsymbol{r}\|_2} \tag{20}$$

is identical (same index picked at every iteration) to OMP run on $\widehat{\boldsymbol{D}}$ with the usual (unnormalized) correlation rule. Equivalently, this selects the atom with the highest absolute cosine similarity to the residual.

Proof. The representation identity is immediate:

$$\boldsymbol{x} = \boldsymbol{D}\boldsymbol{z} = \boldsymbol{D}\,\boldsymbol{W}^{-1}\boldsymbol{W}\boldsymbol{z} = \widehat{\boldsymbol{D}}\,\widehat{\boldsymbol{z}}, \tag{21}$$

and clearly $\mathrm{supp}(\widehat{\boldsymbol{z}}) = S$ because $\boldsymbol{W}$ is diagonal with positive entries.

For the selection rule, note that $\widehat{\boldsymbol{d}}_j = \boldsymbol{d}_j / \|\boldsymbol{d}_j\|_2$, so

$$\langle \boldsymbol{r}, \widehat{\boldsymbol{d}}_j \rangle = \frac{\langle \boldsymbol{r}, \boldsymbol{d}_j \rangle}{\|\boldsymbol{d}_j\|_2}. \tag{22}$$

Thus maximizing correlation with $\widehat{\boldsymbol{d}}_j$ is exactly the same as maximizing normalized correlation, equivalently absolute cosine similarity, with $\boldsymbol{d}_j$.

Finally, diagonal rescaling of the columns in $\boldsymbol{D}_S$ does not change $\mathrm{span}(\boldsymbol{D}_S)$, so the orthogonal projector onto this span is invariant under the rescaling $\boldsymbol{D}_S \mapsto \widehat{\boldsymbol{D}}_S$. Therefore, once the same index is selected, both versions of OMP produce the same least-squares fit and hence the same residual at every step. By induction over the iterations, the selected indices are identical throughout the run. □

Definition D.2 (Exact Recovery Coefficient (ERC) on normalized dictionary)**.** [6] For a support S with $\widehat{\boldsymbol{D}}_S$ full column rank, define

$$\mathrm{ERC}(\widehat{\boldsymbol{D}}; S) := \max_{j \in S^c} \big\| \widehat{\boldsymbol{D}}_S^\dagger \widehat{\boldsymbol{d}}_j \big\|_1, \tag{23}$$

$$\mathrm{ERC}(\widehat{\boldsymbol{D}}; S \,|\, T) := \max_{j \in T \setminus S} \big\| \widehat{\boldsymbol{D}}_S^\dagger \widehat{\boldsymbol{d}}_j \big\|_1, \tag{24}$$

for any $T \supseteq S$, where $\|\cdot\|_1$ denotes the vector ℓ_1 norm. Here $S^c := [k] \setminus S$, so $T \setminus S = T \cap S^c \subseteq S^c$; thus $\mathrm{ERC}(\widehat{\boldsymbol{D}}; S \,|\, T)$ is the same maximum as $\mathrm{ERC}(\widehat{\boldsymbol{D}}; S)$, but over the smaller index set $T \setminus S$.

Lemma D.3 (Monotone ERC improvement under subtree restriction)**.** *Let $\boldsymbol{D} \in \mathbb{R}^{d \times k}$ have arbitrary nonzero column norms and let $\boldsymbol{z}$ be s-sparse with support S. Let $T_0 \supset T_1 \supset \cdots \supset T_L$ be a nested sequence with $T_0 = [k]$ and $S \subseteq T_\ell$ for all $\ell = 0, \ldots, L$. Assume $\widehat{\boldsymbol{D}}_S$ has full column rank. Then the ERC decreases monotonically along the restriction:*

$$\mathrm{ERC}(\widehat{\boldsymbol{D}}; S \,|\, T_L) \le \mathrm{ERC}(\widehat{\boldsymbol{D}}; S \,|\, T_{L-1}) \tag{25}$$

$$\le \cdots \le \mathrm{ERC}(\widehat{\boldsymbol{D}}; S \,|\, T_0) \tag{26}$$

$$:= \mathrm{ERC}(\widehat{\boldsymbol{D}}; S). \tag{27}$$

Proof. By definition, $\mathrm{ERC}(\widehat{\boldsymbol{D}}; S \,|\, T) = \max_{j \in T \setminus S} \|\widehat{\boldsymbol{D}}_S^\dagger \widehat{\boldsymbol{d}}_j\|_1$. Since $\widehat{\boldsymbol{D}}_S^\dagger$ is fixed, shrinking T only restricts the index set over which this maximum is taken, so the value cannot increase. □

Lemma D.4 (ERC threshold implies restricted OMP success)**.** *Under the assumptions of Lemma D.3, if $\mathrm{ERC}(\widehat{\boldsymbol{D}}; S \,|\, T_L) < 1$, then OMP run on $\boldsymbol{D}$ with the normalized selection rule*

$$j^* \in \arg\max_j \frac{|\langle \boldsymbol{r}, \boldsymbol{d}_j \rangle|}{\|\boldsymbol{d}_j\|_2} = \arg\max_j \frac{|\langle \boldsymbol{r}, \boldsymbol{d}_j \rangle|}{\|\boldsymbol{d}_j\|_2 \, \|\boldsymbol{r}\|_2}, \tag{28}$$

restricted to T_L, recovers S in s steps. Equivalently, OMP on $\widehat{\boldsymbol{D}}_{T_L}$ with the standard rule succeeds in s iterations.

Proof. Lemma D.1 shows that the normalized-selection rule on $\boldsymbol{D}$ matches standard OMP on $\widehat{\boldsymbol{D}}$. The classical noiseless ERC theorem of Tropp [64] applied to the restricted dictionary $\widehat{\boldsymbol{D}}_{T_L}$ then yields exact support recovery in s iterations whenever $\max_{j \in T_L \setminus S} \|\widehat{\boldsymbol{D}}_S^\dagger \widehat{\boldsymbol{d}}_j\|_1 < 1$, equivalently whenever $\mathrm{ERC}(\widehat{\boldsymbol{D}}; S \,|\, T_L) < 1$. □

D.5. Proof of Prop. 4.1

Statement. There exist instances with $\mathrm{ERC}(\widehat{\boldsymbol{D}}; S) \ge 1$ yet $\mathrm{ERC}(\widehat{\boldsymbol{D}}; S \,|\, T_L) < 1$ for some nested $T_0 \supset T_1 \supset \cdots \supset T_L$ satisfying the right-subtree assumption $S \subseteq T_\ell$. Consequently, hierarchical OMP yields a strictly larger ERC-certified success region than global OMP on the full dictionary.

Proof. Choose any instance for which the maximizer of $\mathrm{ERC}(\widehat{\boldsymbol{D}}; S)$ lies outside T_L. Removing that maximizer from the admissible index set strictly decreases the maximum, so

$$\mathrm{ERC}(\widehat{\boldsymbol{D}}; S \,|\, T_L) < \mathrm{ERC}(\widehat{\boldsymbol{D}}; S).$$

Hence it is possible to have

$$\mathrm{ERC}(\widehat{\boldsymbol{D}}; S) \ge 1 \qquad \text{but} \qquad \mathrm{ERC}(\widehat{\boldsymbol{D}}; S \,|\, T_L) < 1.$$

Whenever this occurs, Lemma D.4 certifies exact recovery for Hierarchical OMP on the restricted set T_L, while the standard ERC guarantee for global OMP on the full dictionary does not apply. □

E. Step-by-step construction of a Hierarchical Concept Embedding

Assume we are at depth $l > 0$ of the hierarchy. The path from the root to the *current parent* π_l (with embedding $\boldsymbol{a}^{(\pi_l)} \in \mathbb{R}^d$) consists of the $l+1$ ancestor vectors

$$\boldsymbol{A}_l = \{\boldsymbol{a}^{(\pi_0)}, \boldsymbol{a}^{(\pi_1)}, \ldots, \boldsymbol{a}^{(\pi_l)}\}, \qquad \pi_0 < \pi_1 < \cdots < \pi_l. \tag{29}$$

We seek embeddings $\{\boldsymbol{a}^{(j)}\}_{j=1}^b \subset \mathbb{R}^d$ for its b children that satisfy the following conditions:

(i) **Hierarchical Orthogonality**:

$$(\boldsymbol{a}^{(j)} - \boldsymbol{a}^{(\pi_k)})^\top \boldsymbol{a}^{(\pi_k)} = 0, \quad k = 0, \ldots, l, \quad j = 1, \ldots, b. \tag{30}$$

[6] We use Exact Recovery Coefficient (ERC) for the quantity whose threshold at 1 is the classical exact recovery condition in Tropp [64].

By induction, the current parent node π_l already satisfies these orthogonality constraints with respect to its ancestors.

(ii) **Regular $(b-1)$-simplex structure**:

$$(\boldsymbol{a}^{(j)}-\boldsymbol{g}_l)^\top(\boldsymbol{a}^{(k)}-\boldsymbol{g}_l)=\begin{cases}\lambda_l^2, & j=k,\\ -\dfrac{\lambda_l^2}{b-1}, & j\neq k,\end{cases} \tag{31}$$

where $\boldsymbol{g}_l$ is any point satisfying all $l+1$ equations in Eq. (30), and λ_l scales the simplex so that $\angle(\boldsymbol{a}^{(j)},\boldsymbol{a}^{(\pi_l)})=\theta_l$.

(iii) **Cone condition w.r.t. the current parent**:

$$\begin{aligned}&\angle(\boldsymbol{a}^{(j)},\boldsymbol{a}^{(\pi_l)})=\theta_l\\ \iff &\|\boldsymbol{a}^{(j)}-\boldsymbol{a}^{(\pi_l)}\|=\|\boldsymbol{a}^{(\pi_l)}\|\tan\theta_l,\\ &j=1,\dots,b.\end{aligned} \tag{32}$$

This equivalence holds because Eq. (30) implies $\langle \boldsymbol{a}^{(j)}-\boldsymbol{a}^{(\pi_l)},\boldsymbol{a}^{(\pi_l)}\rangle=0$.

E.1. Feasible Subspace Induced by Hierarchical Orthogonality

Let

$$\boldsymbol{A}_l=\begin{bmatrix}\boldsymbol{a}^{(\pi_0)} & \boldsymbol{a}^{(\pi_1)} & \dots & \boldsymbol{a}^{(\pi_l)}\end{bmatrix}\in\mathbb{R}^{d\times(l+1)}. \tag{33}$$

The $l+1$ hyperplanes in (30) intersect in the affine subspace

$$\mathcal{V}_l=\{\boldsymbol{x}\in\mathbb{R}^d:\boldsymbol{A}_l^\top\boldsymbol{x}=\boldsymbol{h}_l\}, \tag{34}$$

where $\boldsymbol{h}_l=[\|\boldsymbol{a}^{(\pi_0)}\|^2,\dots,\|\boldsymbol{a}^{(\pi_l)}\|^2]^\top$. If the ancestor columns of $\boldsymbol{A}_l$ are linearly independent[7], then

$$\dim\mathcal{V}_l=d-(l+1). \tag{35}$$

To be able to embed a regular $(b-1)$-simplex for all depths we therefore require the *depth-dimension condition*

$$d\geq L+b. \tag{36}$$

Equation (36) quantifies the depth–dimension trade-off: one ambient degree of freedom is lost per additional ancestor constraint, while $(b-1)$ directions are always needed to accommodate the regular simplex of conditionally independent children.

We now give a constructive procedure for the children of node π_l at level l.

1. **Find one feasible origin.** Solve the linear system $\boldsymbol{A}_l^\top\boldsymbol{x}=\boldsymbol{h}_l$ to obtain any particular solution $\boldsymbol{g}_l\in\mathcal{V}_l$. If we also impose the cone half-angle condition, the feasible set can be further restricted to the intersection of $\mathcal{V}_l$ with the cone centered at the parent embedding $\boldsymbol{a}^{(\pi_l)}$ and half-angle θ_l from §3.

   A convenient choice, which preserves the full half-angle budget, is to take the current parent embedding itself as the feasible origin, i.e., $\boldsymbol{g}_l=\boldsymbol{a}^{(\pi_l)}$. By construction, every node is orthogonal to all of its ancestors. Hence, for each $k\in\{0,\dots,l\}$,

$$(\boldsymbol{a}^{(\pi_l)}-\boldsymbol{a}^{(\pi_k)})^\top\boldsymbol{a}^{(\pi_k)}=0 \tag{37}$$

$$\implies \boldsymbol{a}^{(\pi_l)\top}\boldsymbol{a}^{(\pi_k)}=\|\boldsymbol{a}^{(\pi_k)}\|^2. \tag{38}$$

2. **Basis for the difference linear space.** Compute an orthonormal basis

$$\boldsymbol{U}_l\in\mathbb{R}^{d\times(d-l-1)}, \tag{39}$$

$$\boldsymbol{A}_l^\top\boldsymbol{U}_l=\boldsymbol{0}, \tag{40}$$

$$\boldsymbol{U}_l^\top\boldsymbol{U}_l=I_{d-l-1}, \tag{41}$$

   e.g., by taking the $d-(l+1)$ left singular vectors of $\boldsymbol{A}_l$ associated with the smallest singular values, denoted $\boldsymbol{U}_{l+2:d}$.

3. **Canonical regular simplex in $\mathbb{R}^{b-1}$.** Use the centred construction $\tilde{\boldsymbol{d}}_i=\boldsymbol{e}_i-\frac{1}{b}\boldsymbol{1},\ i=1,\dots,b$ (cf. Eq. (9)).

4. **Scale $\tilde{\boldsymbol{d}}_j$ to satisfy the cone condition.**

$$\lambda_l=\|\boldsymbol{a}^{(\pi_l)}\|\tan\theta_l, \tag{42}$$

$$\boldsymbol{d}_j=\lambda_l\,\tilde{\boldsymbol{d}}_j. \tag{43}$$

5. **Embed and translate.**

$$\boldsymbol{a}^{(j)}=\boldsymbol{a}^{(\pi_l)}+\boldsymbol{U}_{l+2:d}\,\boldsymbol{d}_j,\qquad j=1,\dots,b. \tag{44}$$

   Choosing the scale factor

$$\lambda_l=\|\boldsymbol{a}^{(\pi_l)}\|\tan\theta_l, \tag{45}$$

forces $\|\boldsymbol{a}^{(j)}-\boldsymbol{a}^{(\pi_l)}\|=\|\boldsymbol{a}^{(\pi_l)}\|\tan\theta_l$ for all j, and therefore $\angle\big(\boldsymbol{a}^{(j)},\boldsymbol{a}^{(\pi_l)}\big)=\theta_l$, which is precisely the requirement in Eq. (32).

F. Additional Experimental Results

F.1. Additional Synthetic Experiment Details

Branching ratio $b=3$, hierarchy depth $L=7$, dimension $d=50$. Initial cone half angle = 85 degrees. Initial vector norm = 0.8. Geometric reduction factor = 0.4. Total leaf nodes = 2187. Total nodes = number of atoms = 3280. Gaussian noise for each leaf for data generation $\sigma^2=10^{-5}$. Generate 5 samples per leaf for a total of 10,935 samples.

[7] This is typical because every level adds a new non-collinear vector.

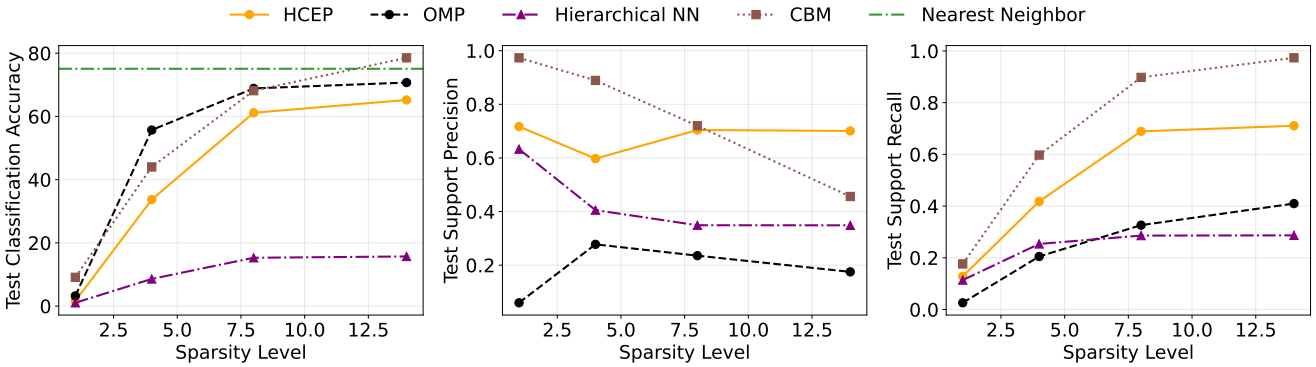

Figure 11. On ImageNet, HCEP achieves competitive accuracy while having higher concept precision/recall than sparse concept prediction baselines (OMP, Hierarchical NN).

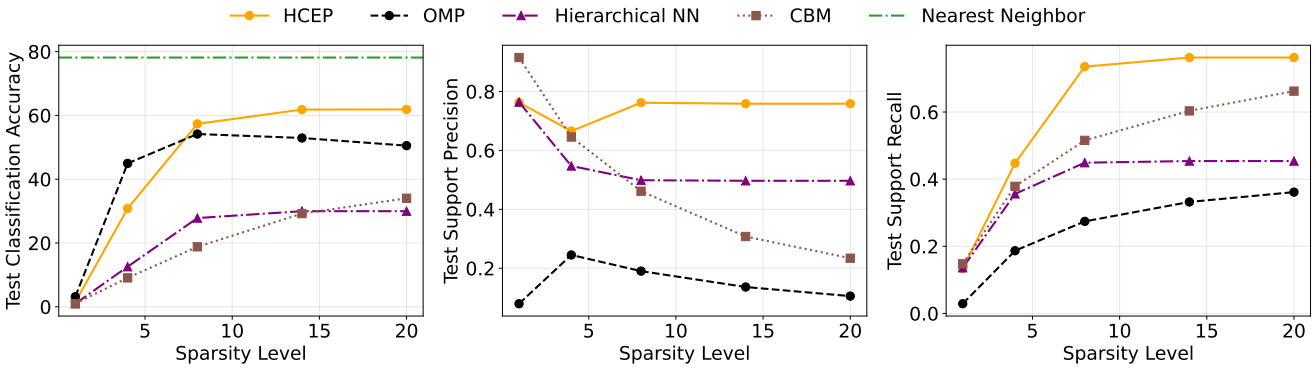

Figure 12. Interpretable image classification on ImageNet (12-shot) using SigLIP [72] embeddings. HCEP exhibits similar improvements in support precision and recall over baselines as with CLIP (cf. Fig. 11), demonstrating generality across vision-language models.

F.2. Text Interpretation of Synset Differences

Optionally, to qualitatively evaluate alternative textual meanings of the synset differences, which form the atoms in our hierarchical dictionary, we use CLIP text embeddings and GPT-5 [51]. First, for each parent-child pair, we generate a text description of the difference between the parent and child synsets using GPT-5. Then, we pool the text embeddings of these descriptions to form a set of candidate concept embeddings. Next, for each synset difference vector, we find its top-k neighbors among the candidate concept embeddings. Finally, we use GPT-5 to generate a summary of the top-k neighbor descriptions, which helps interpret the synset difference. Table 1 shows example interpretations for several parent-child pairs in WordNet [47], the hierarchy underlying ImageNet.

F.3. Additional Real-Data Experiment Details

Model Architecture and Training Details. We use CLIP-ViT-L/14 as the backbone. To train the linear classifier, we use the AdamW optimizer [41] with weight decay 10^{-4} and learning rate 10^{-1}. To train the CBM, we use the Adam optimizer with learning rate 10^{-1} for 500 epochs. We provide the detailed hyperparameters in Table 2.

Table 1. Example text interpretations of child–parent synset differences produced via CLIP embeddings and GPT summarization.

Parent→Child Pair	Text Interpretation
`bear` → `polar bear`	thick matte white fur blending with snow.
`container` → `basket`	open-top woven or perforated sides with handles.
`structure` → `lumbermill`	vertical log-sawing machines and plank conveyors.
`citrus` → `orange`	round, bright orange, pebbled rind.

Synset Difference Interpretations. We use CLIP text embeddings [55] and GPT-5 [51] to generate textual interpretations of the synset differences. We provide the top-10 concepts for each parent-child pair in Table 3. We also include the GPT-5 prompt in Table 5.

Ablation Study on Beam Size. We perform an ablation

study on the beam size for Hierarchical OMP. We vary the beam size from 1 to 8 and evaluate concept recovery accuracy on ImageNette. We report the results in Fig. 14.

Taxonomy Generation Prompt. We use the taxonomy generation prompt from Zeng et al. [71] in Table 4 to generate the taxonomy for CIFAR-100.

Table 2. Key hyperparameters used in experiments for each dataset.

Hyperparameter	**ImageNette**	**CIFAR-100**	**ImageNet**
Batch size	4096	4096	16384
Classification training epochs	500	500	1000
HCEP beam size	8	16	32

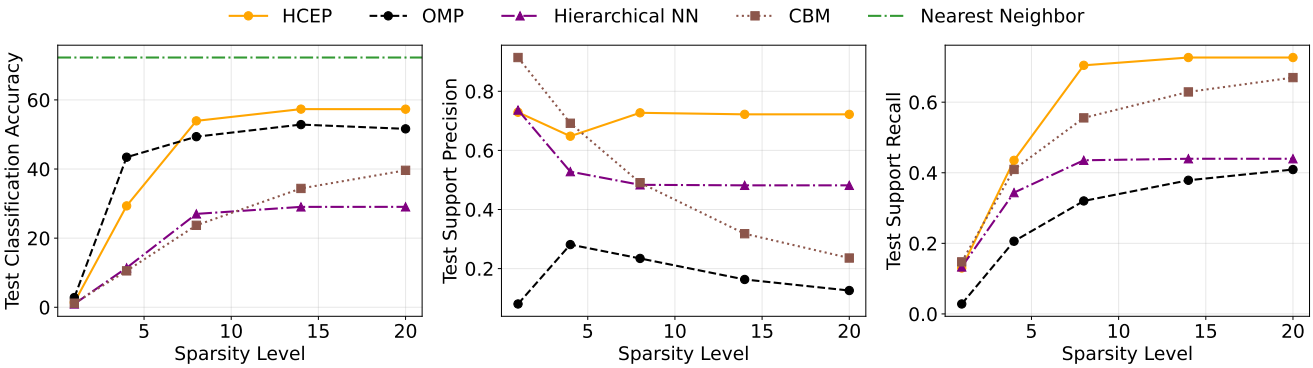

Figure 13. When we restrict the ImageNet training set to 25 images per class, HCEP outperforms all interpretable baselines.

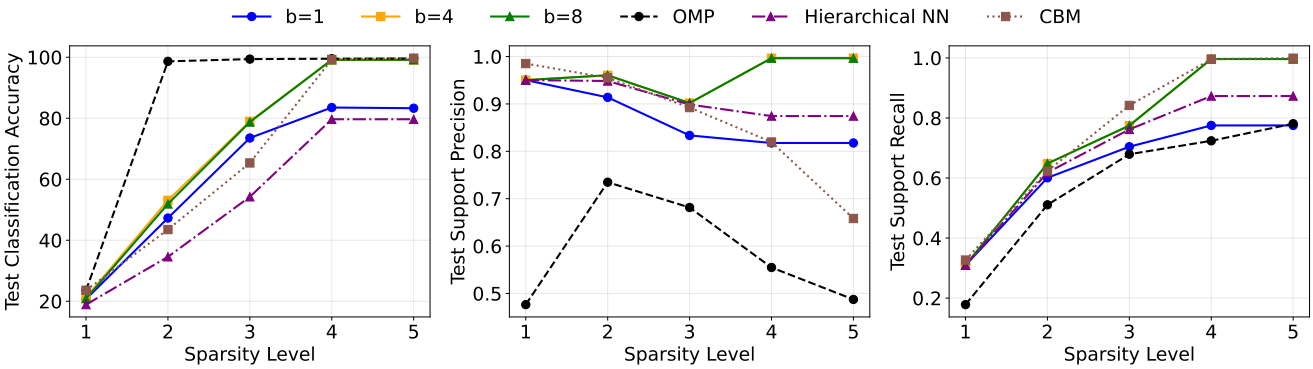

Figure 14. We vary the beam size over $\{1, 4, 8\}$ and evaluate on ImageNette.

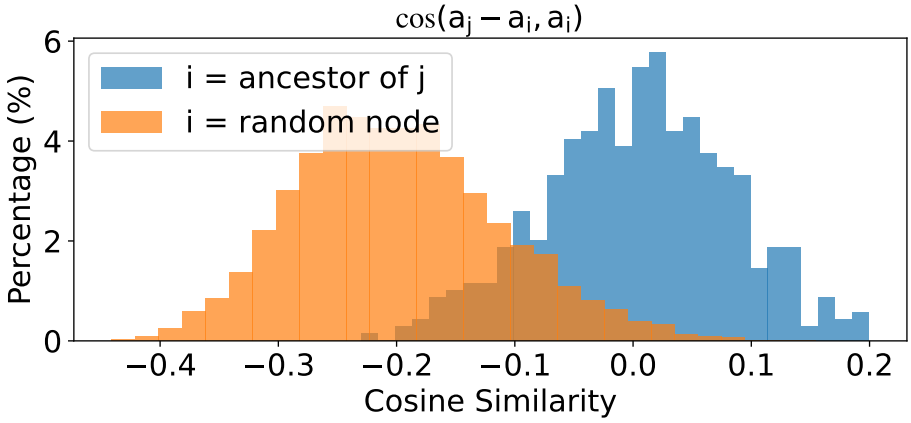

Figure 15. Observed hierarchical orthogonality on CIFAR-100. The cosine similarity between child-parent difference vectors and their parents is close to zero, while random non-parent pairs have significantly lower cosine similarity.

Table 3. Example text interpretations of child–parent synset differences produced via CLIP embeddings and GPT summarization, with the top-10 contributing concepts.

Parent→Child Pair	Top-10 Concepts	Text Interpretation
`bear` → `polar bear`	1. Matte white texture 2. Thick white winter fur coat 3. White plumage in winter season 4. White fur with cream patches 5. Long, silky white coat 6. White fur blending with surroundings 7. Pure white fluffy coat 8. White coat with lemon markings 9. White cue ball 10. Long, corded white coat	thick matte white fur blending with snow.
`container` → `basket`	1. Wicker baskets filled with baguettes 2. Rectangular shopping baskets 3. Stacked woven baskets 4. Rear storage basket 5. Rectangular open-top basket 6. Woven rattan backrest 7. Plastic shopping baskets 8. Perforated cutlery basket in lower rack 9. Woven rattan seating surfaces 10. Rectangular basket frame	open-top woven or perforated sides with handles.
`structure` → `lumbermill`	1. Vertical log slicing machines 2. Massive log cutting machines 3. Metallic sawmill machinery 4. Heavy-duty sawmill frames 5. Conveyor belts with wood pieces 6. Wooden board sorting systems 7. Exposed wooden axles 8. Wooden log feeding chutes 9. Stacks of cut wooden planks 10. Narrow wooden steering wheel	vertical log-sawing machines and plank conveyors.

Table 4. LLM prompt used for taxonomy generation from root and leaf concepts.

Taxonomy Generation Prompt

Given root concept <root> and leaf concepts <leaves>, generate a detailed hierarchical taxonomy that organizes these leaves under the root. Create multiple levels of intermediate category hierarchies to build a rich, fine-grained classification structure. Use as many hierarchical levels as needed to create meaningful semantic groupings and subgroupings.
The format is: 1. Parent Concept 1.1 Child Concept 1.1.1 Grandchild Concept.
CRITICAL: Every single leaf concept from the list must appear exactly as given in the taxonomy as the deepest level nodes. You may and should add multiple levels of intermediate concepts but do not add new leaf concepts. Before finishing, verify that each leaf concept from <leaves> appears in your taxonomy. Aim for depth and semantic richness in the hierarchy.

Table 5. LLM prompt used to generate child-vs-parent residual phrases.

Residual Concept Generation Prompt

TASK: Generate a concise phrase (3-10 words) that describes what distinguishes "<child>" from its parent category "<parent>". This is for a hierarchical sparse model. A residual represents the visual difference between a child and parent category. Most correlated visual concepts (from CLIP embeddings): <concepts_string> REQUIREMENTS:
1. Generate ONE short phrase (3-10 words) that captures the key distinguishing features
2. Base your phrase on the correlated concepts provided above
3. Focus on the most salient visual features
4. Be specific and concrete, not vague or generic
5. Use natural language that a human would use to describe the difference
6. IMPORTANT: Do NOT use the synset names ("<parent>" or "<child>") in your phrase
7. IMPORTANT: Describe only the visual features, not the category name
EXAMPLES OF GOOD PHRASES: - "tawny coat with distinctive facial markings" (for a specific dog breed) - "long curved neck and pink coloration" (for flamingo vs bird) - "striped pattern and elongated body" (for a specific fish) - "metallic surface with cylindrical shape" (for a lighter)