HIERARCHICAL SPECTRAL LEARNING: LOCALITY BEFORE FREQUENCY FOR GRAPH NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Graphs often contain multiple local signal regimes, such as smooth, homophilous regions alongside sharp, heterophilous boundaries. However, most GNNs propagate features using a single graph-wide operator, which mixes these regimes under a shared geometry. With depth, repeated propagation and degree imbalance can amplify hub influence, suppress discriminative variation, and exacerbate oversmoothing and long-range information bottlenecks. We propose **Hierarchical Spectral Learning (HSL)**, a multiresolution framework that builds spatially localized, orthogonal components via diffusion-guided hierarchical coarsening and a degree-aware Haar decomposition, then applies adaptive spectral filtering at each resolution. Diffusion refinement strengthens local separation, while the hierarchy preserves fine-scale contrasts and introduces broader interactions only at coarser levels. We show that HSL preserves hierarchical contrast energy independently of depth and enables hierarchy-mediated communication between distant nodes, helping resist oversmoothing and mitigate oversquashing. Across node classification, graph classification, and community detection benchmarks, HSL consistently outperforms or remains competitive with strong state-of-the-art baselines.

1 INTRODUCTION

Graph neural networks (GNN) typically propagate signals using an operator whose spatial support is defined globally over a *graph-wide propagation geometry*. A single underlying graph operator then determines the interaction pattern and the spectral coordinates used across all regions. This is restrictive when a graph contains multiple local regimes. Smoothing can be useful within a homophilous region, but the same propagation can destroy informative contrasts across heterophilous boundaries or allow high-degree neighborhoods to dominate smaller ones Zhu et al. (2020); Zheng et al. (2024). With increasing depth, these undesired interactions accumulate and contribute to oversmoothing and long-range information compression Oono & Suzuki (2019); Zhao & Akoglu (2019); Topping et al. (2022). The central issue is therefore not only *which frequencies to preserve*, but also *where a spectral component should act and when separated regions should begin to interact*.

Spectral GNNs improve frequency control through polynomial filters, adaptive bases, and learned spectral responses Chien et al. (2020); He et al. (2021); Wang & Zhang (2022); He et al. (2022); Guo & Wei (2023); Huang et al. (2024); Zheng et al. (2024); Guo et al. (2023). However, these filters are still typically defined with respect to a graph-wide propagation geometry. Changing the spectral response determines how strongly different frequencies are weighted, but it does not explicitly localize those spectral coordinates to regions with compatible signal structure. Hierarchical pooling provides multiscale representations, but its primary objective is graph compression. As a result, fine-scale variation may be aggregated before it can be preserved as an explicit spectral component Ying et al. (2018); Lee et al. (2019); Gao & Ji (2019). Hierarchical spectral methods based on repeated eigendecomposition can also incur up to $\mathcal{O}(n^3)$ worst-case cost during hierarchy construction Ma et al. (2019). HMH Hossen & Sahoo (2026) improves scalability through a sparse hierarchy, but it defines locality from learned pairwise affinities that do not necessarily reflect the underlying diffusion geometry. This leaves a gap between *locality* and *frequency*. Existing methods typically decide how to filter signals without first determining the spatial regions over which those filters should operate.

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

We address this limitation with **Hierarchical Spectral Learning (HSL)**. HSL follows the the principle of *locality before frequency*. It first combines graph topology and feature affinity to construct a multi-view geometry, then uses diffusion coherence to suppress structurally incompatible interactions. The resulting local regions are recursively organized into a hierarchy that defines the spectral coordinates themselves, rather than serving only as a pooling structure. Each region contributes a degree-aware scaling direction and an orthogonal contrast subspace. This yields localized multiresolution components on which HSL learns frequency- and resolution-dependent responses. Fine-scale contrast remain spatially localized, while progressively coarser regions introduce broader interactions through shared ancestors. This separates local signal preservation from long-range communication within a single spectral framework.

The construction also provides direct theoretical guarantees. We show that the hierarchy induces a complete orthogonal and locally supported decomposition. We also show that the depth-normalized spectral response preserves hierarchy contrast energy with a depth-independent lower bound, preventing repeated HSL spatial filtering from collapsing into the global scaling subspace. Finally, we show that distant nodes obtain an explicit nonzero interaction through their lowest common hierarchy region, providing a mechanism for mitigating oversquashing without relying on repeated propagation along the original graph path. Empirically, we evaluate these mechanisms through depth-wise contrast preservation, degree-stratified analysis, hierarchy and filter ablations, and scalability studies, together with node classification, graph classification, and community detection.

Our main contributions are as follows. **(i) Locality-first spectral learning:** HSL constructs spectral coordinates from diffusion-coherent local structure rather than a single graph-wide geometry. **(ii) Hierarchy-induced Haar filtering:** We develop a degree-aware multiresolution decomposition with orthogonal, localized scaling and contrast components and adaptive frequency-resolution responses. **(iii) Provable preservation and communication:** We establish local orthogonal decomposition, depth-independent contrast preservation, bounded external influence, and hierarchy-mediated long-range communication. **(iv) Scalable task-general learning:** HSL has near-linear complexity and achieves state-of-the-art or competitive performance across node and graph classification and community detection.

1.1 Preliminaries

Let $G = (V, E)$ be an undirected graph with adjacency matrix $A \in \mathbb{R}^{n\times n}$ and node features $X \in \mathbb{R}^{n\times d}$, where $n = |V|$ and d is the feature dimension. Let $D = \mathrm{diag}(A\mathbf{1})$. The normalized Laplacian is $L = I - D^{-1/2}AD^{-1/2} = U\Lambda U^\top$, where U is the orthonormal graph Fourier basis and Λ contains the Laplacian eigenvalues. When multiple affinities are used, their degree matrices are distinguished by subscripts, e.g., D_{top} and D_{feat}. A spectral filter acts as $g(L)X = Ug(\Lambda)U^\top X$. Scalable spectral GNNs typically avoid explicit eigendecomposition through polynomial approximations $g(L)X \approx \sum_{k=0}^{K} \theta_k P_k(L)X$.

Oversmoothing and Oversquashing. Oversmoothing occurs when repeated propagation progressively removes node-specific variation and drives representations toward a low-dimensional smooth subspace. Let $h_v^{(\ell)}$ denote the representation of node v at layer ℓ and x_u the input feature of node u. Long-range information transmission can be characterized by the sensitivity $\|\partial h_v^{(\ell)}/\partial x_u\|$, which may decrease when information must pass through long paths and structural bottlenecks.

1.2 Related Work

Spectral GNNs improve graph filtering through polynomial or adaptive frequency responses, including GPR-GNN Chien et al. (2020), BernNet He et al. (2021), JacobiConv Wang & Zhang (2022), ChebNetII He et al. (2022), OptBasisGNN Guo & Wei (2023), and UniFilter Huang et al. (2024). More localized variants such as NFGNN Zheng et al. (2024) and diverse spectral filtering Guo et al. (2023) further adapt responses across nodes or regions. However, these methods still largely operate on graph-wide propagation coordinates and do not explicitly construct multiresolution local spectral subspaces. Hierarchical methods such as DiffPool Ying et al. (2018), SAGPool Lee et al. (2019), TopKPool Diehl (2019), Graph U-Net Gao & Ji (2019), and HMH Hossen & Sahoo (2026) organize graph information across multiple resolutions , improving scalability via a sparse hierarchy learned from pairwise affinities. In contrast, HSL uses a diffusion-coherent hierarchy to define the

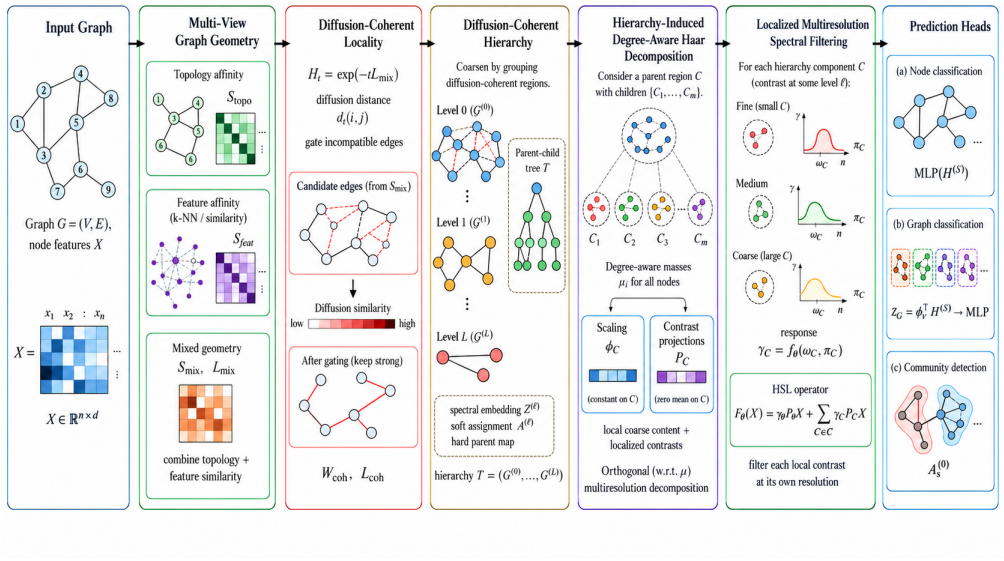

Figure 1: Overview of Hierarchical Spectral Learning Algorithm

spectral coordinates themselves, enabling locally supported orthogonal filtering at fine scales and progressively broader interaction at coarser resolutions.

2 Methodology

Overview. Given an undirected graph $G = (V, E)$ with adjacency matrix $A \in \mathbb{R}^{n\times n}$ and node features $X \in \mathbb{R}^{n\times d}$, HSL first constructs a multi-view geometry from topology and feature similarity and refines it using diffusion coherence (Figure 1). The refined geometry defines a recursive hierarchy, from which we construct degree-aware Haar scaling and contrast components. HSL then learns localized spectral responses over these multiresolution components for downstream prediction.

Multi-View Graph Geometry Topology and feature similarity provide complementary notions of local relatedness. We define the normalized topology affinity $S_{\text{top}} = D_{\text{top}}^{-1/2} A D_{\text{top}}^{-1/2}$, where $D_{\text{top}} = \text{diag}(A\mathbf{1})$. Let $x_i \in \mathbb{R}^d$ denote the feature vector of node i (the i-th row of X). For a symmetric feature-kNN graph, we set

$$(W_{\text{feat}})_{ij} = \exp\left(-\frac{\|x_i - x_j\|_2^2}{2\sigma_f^2}\right) \mathbf{1}\Big\{j \in \mathcal{N}_k^f(i) \text{ or } i \in \mathcal{N}_k^f(j)\Big\},$$

where $\mathcal{N}_k^f(i)$ is the set of k nearest neighbors of i in feature space, and normalize it as $S_{\text{feat}} = D_{\text{feat}}^{-1/2} W_{\text{feat}} D_{\text{feat}}^{-1/2}$. The two views are fused by $S_{\text{mix}} = \alpha S_{\text{top}} + (1-\alpha) S_{\text{feat}}$, with $L_{\text{mix}} = I - S_{\text{mix}}$.

Diffusion-Coherent Locality Local adjacency or feature similarity alone does not guarantee compatible structural behavior, so we refine the fused geometry using diffusion. For scale $t > 0$ (diffusion time), let $H_t = \exp(-tL_{\text{mix}})$ and define the exact heat distance $d_t(i,j) = \|e_i^\top H_t - e_j^\top H_t\|_2$, where $e_i \in \mathbb{R}^n$ is the i-th standard basis vector. In practice, we approximate it using $Y_t = p_{K_d}(L_{\text{mix}})R \in \mathbb{R}^{n\times q_d}$, $q_d \ll n$, where $R \in \mathbb{R}^{n\times q_d}$ is a random probe matrix and $p_{K_d}(\cdot)$ is a degree-K_d polynomial approximating the scalar map $\lambda \mapsto e^{-t\lambda}$, and $\widehat{d}_t(i,j) = \|Y_t(i,:) - Y_t(j,:)\|_2$, where $Y_t(i,:) \in \mathbb{R}^{q_d}$ denotes the i-th row of Y_t, without materializing H_t. On the sparse candidate set $E_c = \text{supp}(A) \cup \text{supp}(W_{\text{feat}})$, where $\text{supp}(\cdot)$ denotes the set of index pairs with nonzero entries, we define

$$g_t(i,j) = \exp\left(-\frac{\widehat{d}_t(i,j)^2}{\tau^2}\right) \mathbf{1}\{\widehat{d}_t(i,j) \le \varepsilon\}. \tag{1}$$

Let $G_t \in \mathbb{R}^{n\times n}$ denote the sparse matrix (diffusion-coherence gate) with $(G_t)_{ij} = g_t(i,j)$ for $(i,j) \in E_c$ and zero otherwise.

The refined affinity is $W_{\text{coh}} = S_{\text{mix}} \odot G_t$, with $D_{\text{coh}} = \text{diag}(W_{\text{coh}}\mathbf{1})$ and $L_{\text{coh}} = I - D_{\text{coh}}^{-1/2} W_{\text{coh}} D_{\text{coh}}^{-1/2}$. Thus, only candidate interactions with compatible diffusion behavior are retained. Approximation details and complexity are given in Appendix D.0.1.

2.1 Diffusion-Coherent Hierarchy

We use the diffusion-coherent geometry to construct a sequence of progressively coarsened graphs by clustering nodes into nested regions. Let $G^{(\ell)} = (V^{(\ell)}, E^{(\ell)})$, $\ell = 0, \dots, L$, denote the graph at level ℓ, with $G^{(0)} = G$, $n_\ell = |V^{(\ell)}|$, and L is the maximum hierarchy level produced by our stopping criterion. At each level, we form $K_\ell = \max\{1, \lfloor r n_\ell \rfloor\}$ parent regions, where $r \in (0,1)$ is the coarsening ratio. We obtain these regions as a partition of $V^{(\ell)}$ into K_ℓ disjoint subsets. These regions become the nodes of $G^{(\ell+1)}$, defining a parent–child relation that records how fine-scale regions are progressively merged into coarser ones.

Spectral partitioning. Let $L_{\text{coh}}^{(\ell)}$ denote the diffusion-coherent Laplacian at level ℓ. We construct a q_ℓ-dimensional spectral embedding $Z^{(\ell)} \in \mathbb{R}^{n_\ell \times q_\ell}$ by minimizing the total Laplacian smoothness (Dirichlet energy) of the embedding coordinates under orthonormality constraints, given by

$$\begin{aligned} \min_{Z^{(\ell)}} \quad & \text{Tr}\Big(Z^{(\ell)\top} L_{\text{coh}}^{(\ell)} Z^{(\ell)}\Big) \\ \text{s.t.} \quad & Z^{(\ell)\top} Z^{(\ell)} = I, \qquad Z^{(\ell)\top} s_0^{(\ell)} = 0, \end{aligned} \tag{2}$$

where $s_0^{(\ell)} = (D_{\text{coh}}^{(\ell)})^{1/2}\mathbf{1} / \|(D_{\text{coh}}^{(\ell)})^{1/2}\mathbf{1}\|_2$ removes the trivial degree-weighted constant direction. The solution is the q_ℓ lowest nontrivial eigenvectors of $L_{\text{coh}}^{(\ell)}$ Chung (1997) . Specifically, if $v_1^{(\ell)}, \dots, v_{q_\ell}^{(\ell)}$ denote these eigenvectors, then $Z^{(\ell)} = [v_1^{(\ell)}, \dots, v_{q_\ell}^{(\ell)}]$. The row $z_i^{(\ell)}$ therefore represents node i in a diffusion-coherent spectral space. We apply k-means with K_ℓ clusters to these row embeddings to obtain the K_ℓ parent regions, so nodes with similar diffusion behavior are preferentially grouped at the next hierarchy level.

Soft assignment and parent selection. Let $p_k^{(\ell)}$ denote the k-means centroid of region k and $z_i^{(\ell)} \in \mathbb{R}^{q_\ell}$ the embedding of node i. We define the soft assignment $(A_s^{(\ell)})_{ik} = \frac{\exp\left(-\kappa \|z_i^{(\ell)} - p_k^{(\ell)}\|_2^2\right)}{\sum_{r=1}^{K_\ell} \exp\left(-\kappa \|z_i^{(\ell)} - p_r^{(\ell)}\|_2^2\right)}$, where $\kappa > 0$ controls assignment sharpness and each row sums to one. The hard parent is selected deterministically as $\pi^{(\ell)}(i) = \min \arg\max_k (A_s^{(\ell)})_{ik}$, where the minimum index resolves the rare case of an exact tie. We encode the resulting parent–child relation by $B^{(\ell)} \in \mathbb{R}^{n_\ell \times K_\ell}$, with $B_{ik}^{(\ell)} = 1$ if $k = \pi^{(\ell)}(i)$ and 0 otherwise..

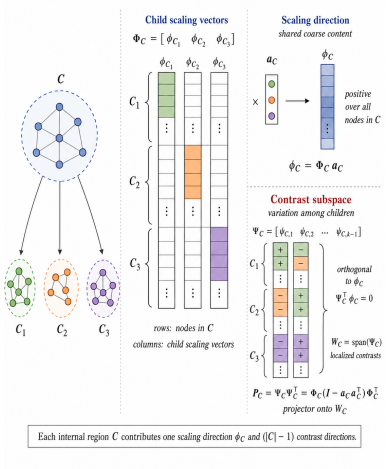

Figure 2: Hierarchy-induced Haar coordinates

Graph coarsening. Each parent region becomes one node at the next level, with $W_{\text{coh}}^{(\ell+1)} = B^{(\ell)\top} W_{\text{coh}}^{(\ell)} B^{(\ell)}$, $D_{\text{coh}}^{(\ell+1)} = \text{diag}(W_{\text{coh}}^{(\ell+1)}\mathbf{1})$, and $L_{\text{coh}}^{(\ell+1)} = I - (D_{\text{coh}}^{(\ell+1)})^{-1/2} W_{\text{coh}}^{(\ell+1)} (D_{\text{coh}}^{(\ell+1)})^{-1/2}$. We repeat the procedure until $|V^{(L)}| \le h_{\min}$, where $h_{\min}$ is the terminal coarse size.. If multiple regions remain, connect them to a virtual root representing V. This produces the parent–child hierarchy $\mathcal{T}$ used to construct the multiresolution coordinates.

2.2 Hierarchy Induced Decomposition

Each node of $\mathcal{T}$ represents a region $C \subseteq V$, whose children $\text{ch}(C) = \{C_1, \dots, C_m\}$ are the finer regions merged to form C. We use this parent-child structure to separate the representation into

a scaling direction summarizing shared information within C and an orthogonal contrast subspace capturing variation among its children. Applying this construction recursively produces a localized Haar-type multiresolution decomposition. Figure 2 illustrates the construction.

For each original node i, define the positive structural mass $\mu_i = d_i^{\mathrm{coh}} + \delta$, where $d_i^{\mathrm{coh}} = \sum_j (W_{\mathrm{coh}})_{ij}$ and $\delta > 0$ ensures positive mass for isolated nodes. For any hierarchy region C, let $Q_C = \sum_{i \in C} \mu_i$ denote its total mass.

Scaling component. The scaling direction $\phi_C(i) = \sqrt{\mu_i / Q_C}$ for $i \in C$ and $\phi_C(i) = 0$ otherwise summarizes the degree-weighted coarse content of region C, with $\|\phi_C\|_2 = 1$.

Contrast component. For an internal region C with children $\{C_1, \ldots, C_m\}$, define $\Phi_C = [\phi_{C_1}, \ldots, \phi_{C_m}] \in \mathbb{R}^{n \times m}$, $a_C = \left[\sqrt{\frac{Q_{C_1}}{Q_C}}, \ldots, \sqrt{\frac{Q_{C_m}}{Q_C}}\right]^\top \in \mathbb{R}^m$. Since the children partition C, $\|a_C\|_2 = 1$ and $\phi_C = \Phi_C a_C$. The orthogonal contrast subspace is $\mathcal{W}_C = \{\Phi_C z : a_C^\top z = 0\}$, which has dimension $m - 1$. If $\Psi_C \in \mathbb{R}^{n \times (m-1)}$ is any orthonormal basis of $\mathcal{W}_C$, its projector is

$$P_C = \Psi_C \Psi_C^\top = \Phi_C (I - a_C a_C^\top) \Phi_C^\top = \sum_{j=1}^{m} \phi_{C_j} \phi_{C_j}^\top - \phi_C \phi_C^\top. \tag{3}$$

The projector P_C extracts the contrast (difference) component within C by removing the scaling direction ϕ_C and retaining only variation among the children. Thus, $P_C X$ captures variation among the children of C. Fine hierarchy regions describe localized contrasts, while ancestor regions capture contrasts over progressively broader scales. These orthogonal, localized components now provide the coordinates on which HSL learns its spectral response.

Localized Multiresolution Spectral Filtering. The hierarchy provides the complete decomposition $X = P_0 X + \sum_{C \in \mathcal{I}(\mathcal{T})} P_C X$, where $P_0 = \phi_V \phi_V^\top$ and $\mathcal{I}(\mathcal{T})$ contains all internal regions. Each contrast is characterized by its effective frequency $\omega_C = \mathrm{Tr}(P_C L_{\mathrm{mix}}) / \mathrm{rank}(P_C)$ and normalized hierarchy resolution $\eta_C = 1 - \ell(C)/L \in [0, 1]$, where $\ell(C)$ denotes its hierarchy level from fine to coarse. Thus η_C is close to 1 for fine regions and close to 0 for coarse regions, and L is the final hierarchy depth. At HSL layer $s \in \{1, \ldots, S\}$, the response of component C is $\gamma_{C,s} = \exp\left[\frac{\beta}{S} \tanh f_{\theta_s}(\omega_C, \eta_C)\right]$, where $\beta > 0$ controls the total response budget. Since $|\tanh(\cdot)| \leq 1$, $e^{-\beta/S} \leq \gamma_{C,s} \leq e^{\beta/S}$. We define the spatial operator $\mathcal{F}_s(X) = P_0 X + \sum_{C \in \mathcal{I}(\mathcal{T})} \gamma_{C,s} P_C X$.

The HSL update is

$$H^{(s+1)} = \rho\Big(\mathcal{F}_s(H^{(s)}) W_s + H^{(s)} W_{\mathrm{skip},s}\Big). \tag{4}$$

Prediction Task. For node classification we apply a node-wise classifier to the final representations, $\widehat{Y} = \mathrm{MLP}(H^{(S)})$. For graph classification, $z_G = \phi_V^\top H^{(S)}$ (a mass-weighted global pooling induced by the root scaling vector) and $\widehat{y}_G = \mathrm{MLP}(z_G)$. Both supervised tasks are optimized using cross-entropy loss. For community detection, we use the finest-level assignment $A_s^{(0)}$ computed during hierarchy construction. Hard assignments define non-overlapping communities, while soft assignments are converted to overlapping memberships using the fixed rule described in Appendix D.1. Ground-truth community labels are used only for evaluation.

2.3 Theoretical Properties

The following results characterize successive linear spatial operators $\mathcal{F}_s$ under a fixed hierarchy, excluding the subsequent learned feature transformations and nonlinearities.

Theorem 1 (Local Orthogonal Haar Decomposition) *For the hierarchy $\mathcal{T}$, the root scaling component and hierarchy contrast projectors form a complete orthogonal decomposition, $\phi_V \phi_V^\top + \sum_{C \in \mathcal{I}(\mathcal{T})} P_C = I, \qquad P_C P_D = 0 \ \ (C \neq D)$. Moreover, each P_C is supported only on region C. Hence, every graph signal decomposes into mutually orthogonal, spatially localized components.*

*Proof in Appendix A.*The hierarchy therefore determines not only the resolution of each component, but also where it can act, i.e., a contrast associated with C cannot directly mix signals outside C.

Theorem 2 (Hierarchy-Bounded External Influence) *Let $i \neq j$ and let $A = \mathrm{LCA}(i,j)$ be their lowest common ancestor. For S successive linear HSL spatial filters with a fixed hierarchy, $\left|e_i^\top \mathcal{F}_S \cdots \mathcal{F}_1 e_j\right| \leq \frac{2e^\beta \sqrt{\mu_i \mu_j}}{Q_A}$. No hierarchy region strictly below A contributes to this interaction. Moreover, since $Q_A \geq \mu_j$, $\left|e_i^\top \mathcal{F}_S \cdots \mathcal{F}_1 e_j\right| \leq 2e^\beta \sqrt{\frac{\mu_i}{\mu_j}}$.* Proof in Appendix B.

An external hub can affect node i only through a hierarchy region containing both nodes, and its contribution is normalized by that region's total mass. Thus, for $\mu_h \gg \mu_i$, the upper bound decreases as $\sqrt{\mu_i/\mu_h}$, preventing hub degree from producing unbounded external influence.

Theorem 3 (Depth-Independent Contrast Preservation) *Let $\mathcal{E}_\mathcal{T}(X) = \|(I - P_0)X\|_F^2 = \sum_{C \in \mathcal{I}(\mathcal{T})} \|P_C X\|_F^2$ denote the hierarchy contrast energy. For S successive linear HSL spatial filters with the depth-stable response, $e^{-2\beta}\mathcal{E}_\mathcal{T}(X^{(0)}) \leq \mathcal{E}_\mathcal{T}(X^{(S)}) \leq e^{2\beta}\mathcal{E}_\mathcal{T}(X^{(0)})$. Hence, whenever $\mathcal{E}_\mathcal{T}(X^{(0)}) > 0$, $\inf_{S \geq 1} \mathcal{E}_\mathcal{T}(X^{(S)}) > 0$. Repeated HSL spatial filtering therefore cannot collapse all localized contrasts into the global scaling subspace.*

*Proof in Appendix C.*So, unlike a fixed per-layer attenuation whose product may vanish with depth, the depth-normalized response bounds the total attenuation of localized contrasts independently of network depth, providing resistance to oversmoothing caused by repeated spatial filtering.

Theorem 4 (Hierarchy-Mediated Long-Range Communication) *Let u and v lie in different children of their lowest common ancestor C, and define $\mathcal{F}_{C,s}(X) = \gamma_{C,s} P_C X$. Then $(P_C)_{uv} = -\frac{\sqrt{\mu_u \mu_v}}{Q_C}$, $\left\| \frac{\partial [\mathcal{F}_{C,s}(X)]_u}{\partial x_v} \right\|_2 = \gamma_{C,s} \frac{\sqrt{\mu_u \mu_v}}{Q_C} > 0$. Thus, unlike edge-local propagation whose source–target influence may decay through repeated path-wise multiplication, HSL provides a non-vanishing hierarchy-mediated channel once u and v share a region. If $n_{\ell+1} \leq r n_\ell$ for $0 < r < 1$, every node pair shares such a region within $\mathcal{O}(\log n)$ hierarchy levels.*

Proof in Appendix D. This provides a direct mechanism for resisting path-dependent oversquashing without requiring repeated propagation along the original graph path.

3 Experiments and Results

Experimental protocol. We evaluate HSL on node classification, graph classification, and non-overlapping community detection. For the heterophily benchmarks of Platonov et al. (2023), we use the official splits and metrics, including ROC-AUC for Minesweeper, Tolokers, and Questions and accuracy for the remaining classification datasets. Results are averaged across the benchmark splits or independent random seeds as specified in the appendix. HSL is trained on NVIDIA RTX 4090 GPUs. Dataset statistics and split protocols are reported in Appendix D.2, hyperparameter search and selected configurations in Appendix D.3, and implementation details in Appendix D.4.

Node Classification We evaluate HSL on homophilic, heterophilic, and large-scale node-classification benchmarks. We compare against standard message-passing baselines and representative spectral and heterophily-oriented methods, including GCN, GAT, GCNII, GPR-GNN Chien et al. (2020), BernNet He et al. (2021), ChebNetII He et al. (2022), JacobiConv Wang & Zhang (2022), OptBasisGNN Guo & Wei (2023), UniFilter Huang et al. (2024), Specformer, TFE-GNN, M2M-GNN, and SLOG. Complete baseline settings and additional datasets are provided in Appendix E.

Results. Table 1 reports the main node-classification results. HSL achieves the strongest overall performance across the benchmark suite, with particularly consistent gains on challenging heterophilic graphs while remaining highly competitive on large-scale datasets. These results support the central motivation of HSL, i.e., constructing spectral coordinates from local diffusion geometry allows discriminative local contrasts to be preserved instead of being repeatedly mixed through a single graph-wide propagation operator.

Graph Classification. We evaluate whether the same hierarchy transfers to graph-level prediction on eight molecular and social-network benchmarks: PROTEINS, NCI1, NCI109, MUTAG, D&D, IMDB-M, REDDIT-12K, and Mutagenicity. We compare against representative hierarchical and pooling-based methods, including DiffPool Ying et al. (2018), EigenPool Ma et al. (2019), gPool

Table 1: Node classification performance (%). Higher is better.

	Heterophilic Graphs					Large-Scale Graphs		
Method	**Minesweeper**	**Tolokers**	**Roman-empire**	**Squirrel (filtered)**	**Chameleon (filtered)**	**Penn94**	**Genius**	**ogbn-arxiv**
GCN	78.80 ± 0.00	78.40 ± 0.10	53.74 ± 0.29	23.40 ± 0.70	27.20 ± 1.20	81.8 ± 0.6	88.9 ± 0.5	73.2 ± 0.4
GAT	78.70 ± 0.10	77.60 ± 0.00	52.74 ± 0.29	24.60 ± 0.60	28.40 ± 2.00	$82.6 \pm 0.6^{\dagger}$	$90.1 \pm 0.5^{\dagger}$	$67.60 \pm 0.30^{\ddagger}$
GPR-GNN	79.10 ± 0.00	77.50 ± 0.10	74.74 ± 0.29	34.30 ± 0.90	$\mathbf{43.20 \pm 2.00}$	$\mathbf{85.9 \pm 0.4}$	91.2 ± 0.5	73.5 ± 0.3
BernNet	78.80 ± 0.00	77.20 ± 0.70	72.74 ± 0.29	36.10 ± 0.70	37.80 ± 0.70	83.7 ± 0.5	89.0 ± 0.4	72.1 ± 0.4
ChebNetII	82.30 ± 0.10	78.30 ± 0.30	73.74 ± 0.29	35.00 ± 0.40	33.50 ± 0.50	84.1 ± 0.4	92.3 ± 0.3	72.8 ± 0.2
GCNII	78.80 ± 0.00	77.80 ± 0.10	74.74 ± 0.29	31.40 ± 0.80	38.00 ± 1.00	$83.8 \pm 0.5^{\dagger}$	$90.4 \pm 0.4^{\dagger}$	$69.38 \pm 0.40^{\ddagger}$
Specformer	88.93 ± 0.41	80.42 ± 0.55	69.94 ± 0.34	27.60 ± 1.70	28.20 ± 2.30	$84.6 \pm 0.5^{\dagger}$	$91.8 \pm 0.4^{\dagger}$	$72.6 \pm 0.4^{\dagger}$
OptBasisGNN	84.80 ± 2.40	79.85 ± 1.63	68.34 ± 0.30	28.66 ± 1.10	39.70 ± 2.40	84.8 ± 0.6	91.0 ± 0.2	73.0 ± 0.3
HMH	87.40 ± 0.80	84.56 ± 1.90	76.10 ± 0.70	38.80 ± 0.46	40.85 ± 0.65	–	–	73.30 ± 0.40
UniFilter	86.80 ± 0.70	78.12 ± 0.83	74.50 ± 0.56	33.10 ± 0.80	41.20 ± 1.30	$85.1 \pm 0.5^{\dagger}$	$91.6 \pm 0.4^{\dagger}$	$73.6 \pm 0.3^{\dagger}$
TFE-GNN	86.85 ± 0.33	79.23 ± 1.13	74.20 ± 0.35	31.47 ± 1.15	41.16 ± 1.41	$85.4 \pm 0.4^{\dagger}$	$92.1 \pm 0.3^{\dagger}$	$71.34 \pm 0.55^{\ddagger}$
M2M-GNN	82.25 ± 1.43	80.85 ± 1.13	68.36 ± 3.23	33.60 ± 1.70	32.20 ± 2.30	$84.3 \pm 0.5^{\dagger}$	$91.4 \pm 0.4^{\dagger}$	$72.15 \pm 0.83^{\ddagger}$
SLOG(N)	84.40 ± 0.80	81.00 ± 0.60	–	38.20 ± 1.20	43.02 ± 2.20	$85.3 \pm 0.4^{\dagger}$	$92.0 \pm 0.3^{\dagger}$	$71.90 \pm 0.20^{\ddagger}$
HSL (ours)	$\mathbf{89.40 \pm 0.80}$	$\mathbf{84.78 \pm 0.86}$	$\mathbf{76.60 \pm 1.24}$	$\mathbf{39.50 \pm 0.20}$	41.20 ± 0.20	84.0 ± 0.3	$\mathbf{92.81 \pm 0.3}$	$\mathbf{75.1 \pm 0.2}$

Table 2: Graph classification accuracy (%) . Results are mean $\pm$95% confidence interval

Method	**PROTEINS**	**NCI1**	**NCI109**	**MUTAG**	**D&D**	**IMDB-M**	**REDDIT-12K**	**Mutagenicity**
DiffPool	68.90 ± 2.95	77.73 ± 0.83	77.13 ± 1.49	79.22 ± 1.02	78.61 ± 1.32	51.31 ± 0.72	44.8 ± 1.5	80.78 ± 1.12
EigenPool	70.84 ± 1.06	77.24 ± 0.96	75.99 ± 1.42	–	78.63 ± 1.36	49.81 ± 0.48	44.23 ± 1.3	80.11 ± 0.73
gPool	71.71 ± 1.75	76.25 ± 1.39	76.61 ± 1.39	67.85 ± 1.38	77.02 ± 1.32	48.3 ± 1.6	44.35 ± 0.76	80.30 ± 1.54
SAGPool(G)	71.72 ± 2.19	77.88 ± 1.59	75.74 ± 1.47	76.78 ± 2.12	78.70 ± 2.29	49.47 ± 0.56	42.3 ± 1.6	79.72 ± 0.79
TopKPool	70.48 ± 1.01	67.61 ± 3.36	73.63 ± 0.55	77.61 ± 3.36	73.63 ± 0.55	48.59 ± 0.72	45.3 ± 1.43	$\mathbf{82.45 \pm 1.32}$
SEP	75.22 ± 0.63	74.83 ± 1.49	76.58 ± 1.04	85.83 ± 1.49	78.58 ± 1.04	50.78 ± 0.75	45.3 ± 1.6	79.1 ± 1.2
HMH	76.8 ± 2.1	80.9 ± 2.5	80.7 ± 2.0	89.50 ± 0.30	79.7 ± 1.8	52.5 ± 2.2	46.1 ± 1.2	81.7 ± 1.8
Mincut-Pool	76.25 ± 0.8	82.38 ± 0.4	72.32 ± 0.48	89.7 ± 1.1	79.2 ± 0.8	52.8 ± 0.5	–	79.34 ± 1.24
MaxCut-Pool	$\mathbf{77.10 \pm 2.5}$	81.20 ± 1.3	–	79.2 ± 4.5	$\mathbf{81.3 \pm 2.7}$	$\mathbf{54.1 \pm 2.4}$	43.46 ± 0.45	80.34 ± 1.24
HSL (ours)	76.38 ± 1.41	$\mathbf{83.9 \pm 1.55}$	$\mathbf{81.5 \pm 1.30}$	$\mathbf{94.50 \pm 1.8}$	80.24 ± 1.8	53.3 ± 1.35	$\mathbf{46.50 \pm 0.54}$	$\mathbf{82.57 \pm 1.8}$

Gao & Ji (2019), SAGPool Lee et al. (2019), TopKPool Diehl (2019), SEP Wu et al. (2022), MinCut-Pool Bianchi et al. (2019), and MaxCut-Pool Abate & Bianchi (2025). We report mean accuracy with 95% confidence intervals over 10 random splits. The details of the complete split and training settings are provided in Appendix F.

As shown in Table 2, HSL achieves the best performance on NCI1, MUTAG, REDDIT-12K, and Mutagenicity and remains competitive on the remaining benchmarks. In particular, HSL reaches 94.5% on MUTAG, improving by 4.8 percentage points over the strongest listed baseline. These results indicate that the hierarchy-induced representation preserves discriminative local substructures while integrating them into a compact graph-level representation.

Non-Overlapping Community Detection. We evaluate on PolBlog, Airport, Cora, Football, PolBooks, and HighSchool using normalized mutual information (NMI), defined as $\mathrm{NMI}(C, \widehat{C}) = \frac{2I(C;\widehat{C})}{H(C)+H(\widehat{C})}$, where $I(\cdot;\cdot)$ denotes mutual information and $H(\cdot)$ is entropy. Larger values indicate stronger agreement between predicted and ground-truth communities. We compare against representative embedding-based methods, including node2vec+K-means Grover & Leskovec (2016), DeepWalk+K-means Perozzi et al. (2014), and Laplacian Eigenmaps, together with direct community-detection approaches such as BigCLAM Yang & Leskovec (2013), SNMF Kuang et al. (2015), G2R Xiaotian Han (2022), GCC Fettal et al. (2022), and CPGC-G Li et al. (2023).

Table 3: Non-overlapping community detection (NMI). Higher is better.

Method	PolBlog	Airport	Cora	Football	PolBooks	HighSchool
node2vec + K-means	0.71	0.62	0.52	0.79	0.68	0.64
DeepWalk + K-means	0.69	0.61	0.50	0.78	0.66	0.63
Laplacian Eigenmaps	0.66	0.58	0.49	0.75	0.64	0.61
BigCLAM	0.70	0.60	0.48	0.76	0.67	0.62
SNMF	0.68	0.57	0.47	0.74	0.63	0.60
G2R	0.62	0.51	0.55	0.66	0.63	0.59
GCC	0.4507	0.4785	**0.57**	0.5538	0.5555	0.5235
CPGC-G	0.1561	0.2313	0.4091	0.4238	0.3786	0.3198
HSL (ours)	**0.74**	**0.66**	0.56	**0.83**	**0.71**	**0.68**

Results. Table 3 shows that HSL achieves the strongest overall NMI across the evaluated networks, indicating that the diffusion-coherent hierarchy captures meaningful community structure without supervised labels. Results for overlapping communities are provided in Appendix D.1.

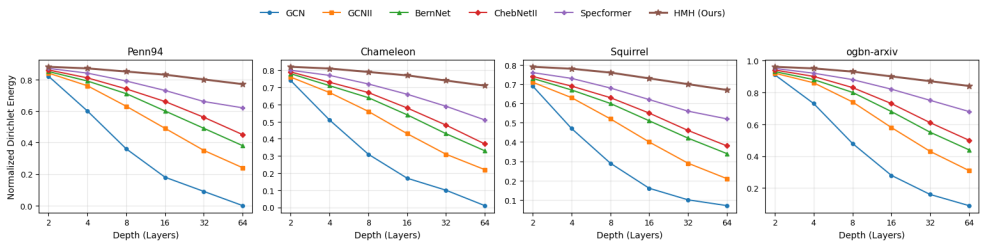

Figure 3: **Oversmoothing analysis.** Normalized Dirichlet energy across network depth. Higher retained energy indicates stronger preservation of non-smooth graph-signal variation.

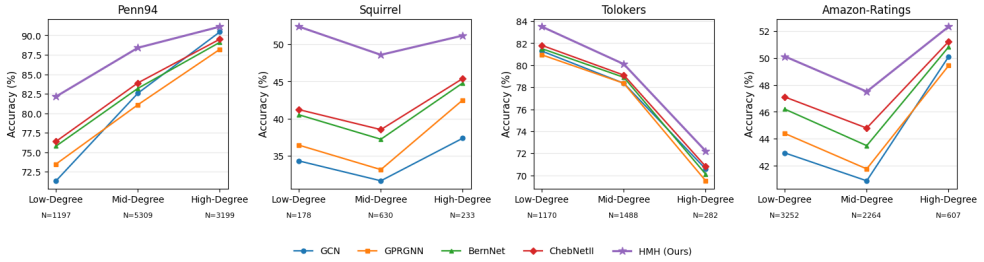

Figure 4: **Hub-degree bias analysis** across low-degree, medium-degree, and high-degree nodes.

Resistance to oversmoothing. We examine how graph signal variation evolves with network depth. Figure 3 reports the normalized Dirichlet energy as the number of layers increases. Conventional propagation-based models rapidly lose variation as depth grows, whereas HSL retains substantially higher energy (i.e., less oversmoothing) even for deep networks. This provides complementary empirical support for Theorem 3, which guarantees depth-independent preservation of hierarchy contrast energy under repeated HSL spatial filtering. Results on additional datasets and deeper networks are provided in Appendix G.

Hub-degree bias. We next examine whether node degree disproportionately affects prediction. Test nodes are grouped into *Spoke* (low degree), *Medium*, and *Hub* (high degree) cohorts. As shown in Fig. 4, standard propagation models exhibit a larger performance gap between low- and high-degree nodes, whereas HSL produces more balanced performance across degree groups. This is consistent with the local support property of the Haar contrast projectors, i.e., an external hub cannot directly modify fine-scale contrast components outside its hierarchy region. The choice of degree thresholds and the effect of degree-aware Haar weighting are further examined in Appendices H and H.2.

Mitigating oversquashing. We use Tree-NeighborsMatch as a controlled benchmark for long-range information propagation. As the tree radius increases, the target depends on progressively more distant leaf nodes, while an exponentially growing amount of information must pass through intermediate bottlenecks. The benchmark construction, training protocol, and additional settings are described in the appendix. As shown in Table 4, the performance of conventional message-passing and graph-wide spectral baselines decreases as the required communication distance grows. In contrast, HSL introduces additional communication channels through progressively coarser hierarchy regions, consistent with Theorem 4.

Table 4: Tree-NeighborsMatch accuracy (%).

Radius	GIN	GCNII	GPR-GNN	ChebNetII	HSL
4	96.8	97.5	98.0	97.8	**99.4**
6	83.5	88.1	89.7	90.2	**99.1**
8	53.2	64.7	68.9	66.1	**98.7**
10	50.1	53.8	56.4	54.9	**98.2**

Ablation study. We ablate one HSL component at a time: *w/o feature view* uses topology only; *w/o diffusion* constructs the hierarchy directly from the multi-view affinity; *w/o hierarchy* removes multiresolution coarsening; and *uniform Haar* replaces degree-aware masses with uniform weights. All variants follow the same training protocol. As shown in Table 5, removing diffusion or the hierarchy yields the largest and most consistent degradation, particularly on heterophilic graphs, while topology-only and uniform-Haar variants also underperform the full model. The trend persists on larger benchmarks such as Questions and ogbn-arxiv, indicating that multi-view geometry, diffusion locality, hierarchical organization, and degree-aware coordinates

Table 5: Ablation study on representative graph and node benchmarks (mean±std over 5 seeds).

Variant	Mutag.	Actor	Cham.	Squir (un-filtered).	Roman	Tolok.	Amazon	Mine.	Questions	ogbn-arxiv
Full HSL	**94.4** $\pm$ 4.8	**43.3** $\pm$ 0.7	**41.20** $\pm$ 1.2	**50.9** $\pm$ 1.4	**76.60** $\pm$ 0.7	**84.78** $\pm$ 1.9	**48.6** $\pm$ 0.6	**89.40** $\pm$ 0.8	**98.2** $\pm$ 0.4	**75.10** $\pm$ 0.4
Topology-only	89.8 $\pm$ 0.7	41.1 $\pm$ 1.0	39.5 $\pm$ 1.5	49.2 $\pm$ 1.4	73.2 $\pm$ 0.8	82.1 $\pm$ 1.7	45.9 $\pm$ 0.7	84.9 $\pm$ 0.9	97.1 $\pm$ 0.5	71.8 $\pm$ 0.5
w/o diffusion	88.1 $\pm$ 0.6	38.9 $\pm$ 1.2	36.6 $\pm$ 1.7	46.1 $\pm$ 1.6	70.8 $\pm$ 1.0	79.6 $\pm$ 1.8	43.7 $\pm$ 0.8	82.1 $\pm$ 1.1	95.8 $\pm$ 0.6	70.6 $\pm$ 0.5
w/o degree-aware Haar	89.0 $\pm$ 0.6	41.7 $\pm$ 0.9	40.1 $\pm$ 1.3	49.0 $\pm$ 1.5	73.9 $\pm$ 0.9	82.7 $\pm$ 1.6	46.3 $\pm$ 0.7	85.2 $\pm$ 0.9	97.3 $\pm$ 0.5	72.1 $\pm$ 0.4
w/o hierarchy	88.4 $\pm$ 0.5	37.8 $\pm$ 1.3	37.4 $\pm$ 1.8	45.0 $\pm$ 1.7	69.7 $\pm$ 1.1	78.9 $\pm$ 2.0	42.8 $\pm$ 0.9	81.3 $\pm$ 1.2	95.1 $\pm$ 0.7	69.9 $\pm$ 0.6

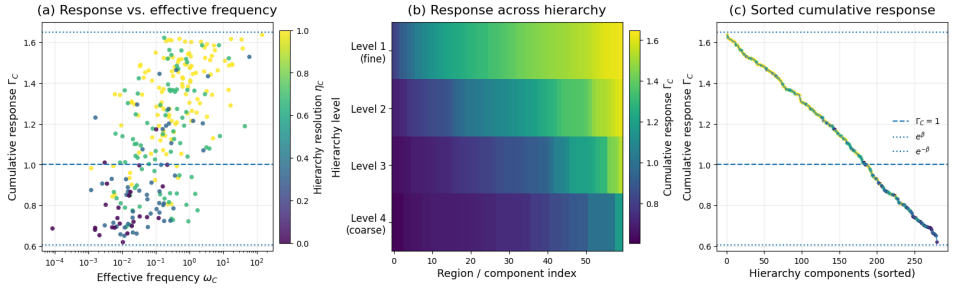

Figure 5: **Learned multiresolution spectral responses on Squirrel.** For each hierarchy region C, $\Gamma_C = \prod_{s=1}^{S} \gamma_{C,s}$ denotes its cumulative spectral response. (a) Γ_C versus effective frequency ω_C, with color indicating normalized hierarchy resolution η_C. (b) Level-wise response map across the four hierarchy resolutions; responses within each level are sorted and binned only for visualization. (c) Sorted responses over all hierarchy components. The dashed line denotes the neutral response $\Gamma_C = 1$, and dotted lines denote the theoretical bounds $e^{-\beta}$ and e^{β}.

each contribute to HSL. Expanded node- and cross-task ablations, hierarchy-quality comparisons, and sensitivity to diffusion are provided in Appendices H.7–H.6.

Learned Multiresolution Spectral Responses. Figure 5 shows that HSL learns a distinctly nonuniform response across hierarchy-induced spectral components. In Fig. 5(a), components with similar effective frequency ω_C can receive different cumulative gains Γ_C, showing that the response is not determined by frequency alone but also by the hierarchy resolution η_C. Figure 5(b) makes this spatial-scale dependence explicit. Fine, intermediate, and coarse hierarchy levels exhibit different response patterns rather than sharing one global spectral profile. Finally, Fig. 5(c) shows that HSL learns both attenuation and amplification while keeping all cumulative responses within the depth-stable range $[e^{-\beta}, e^{\beta}]$. Together, these results support the central HSL principle of *locality before frequency*, i.e., the model adapts filtering jointly to the spectral character of a component and the spatial resolution at which that component occurs.

Scalability. Let $M = \mathcal{O}(|E| + kn)$ denote the size of the sparse multi-view candidate graph. HSL separates one-time hierarchy construction from downstream training. For fixed diffusion order, spectral embedding dimension, eigensolver iterations, and geometric coarsening ratio, hierarchy construction consists of sparse operations and scales near-linearly with M. Once the hierarchy is fixed, the recursive Haar transform and localized spectral filtering require $\mathcal{O}(nd)$ computation per layer, yielding training-time spatial cost linear in the number of nodes. Detailed preprocessing time, runtime, memory, and complexity analysis are reported in Appendix H.9.

Limitations. HSL assumes a fixed hierarchy during training, making it less suitable for dynamic graphs where connectivity or region structure changes over time. Updating such graphs would require periodically reconstructing the hierarchy, introducing additional preprocessing cost. Extending HSL to learn or update hierarchical regions online is an important direction for future work.

Conclusion We introduced HSL, a locality-first multiresolution spectral framework in which diffusion-coherent graph structure determines the spectral coordinates before filtering is performed. HSL combines multi-view diffusion geometry, hierarchical coarsening, and degree-aware Haar decomposition to separate locally supported contrast information from progressively coarser graph structure. Our analysis establishes improved diffusion separation, orthogonal and localized multiresolution representations, finite-depth resistance to contrast collapse, and a hierarchical mechanism for long-range communication. Experiments on node classification, graph classification, and community detection demonstrate the effectiveness of the proposed framework while maintaining scalable sparse computation.

Reproducibility Statement

We provide dataset statistics, evaluation protocols, hyperparameter search spaces, selected configurations, architectural details, and computational complexity in the main paper and appendix. The hierarchy construction, diffusion approximation, multiresolution filtering, and training procedure are specified to support independent reproduction of HSL.

Impact Statement

HSL is a general graph-learning method intended to improve representation learning on graphs with heterogeneous structure. Potential applications include scientific, biological, and network data analysis. As with other graph-learning methods, predictions may inherit biases or errors present in the underlying data, particularly when applied to human-centered networks. We therefore recommend evaluating data quality and model behavior carefully in application-specific settings.

AI Assistance Statement

AI-assisted tools were used to help improve language clarity, grammar, and formatting during manuscript preparation.

References

Carlo Abate and Filippo Maria Bianchi. Maxcutpool: differentiable feature-aware maxcut for pooling in graph neural networks. In *International Conference on Learning Representations*, volume 2025, pp. 52506–52533, 2025.

Filippo Maria Bianchi, Daniele Grattarola, and Cesare Alippi. Mincut pooling in graph neural networks. *arXiv preprint arXiv:1907.00481*, 5, 2019.

Eli Chien, Jianhao Peng, Pan Li, and Olgica Milenkovic. Adaptive universal generalized pagerank graph neural network. *arXiv preprint arXiv:2006.07988*, 2020.

Fan RK Chung. *Spectral graph theory*, volume 92. American Mathematical Soc., 1997.

Frederik Diehl. Edge contraction pooling for graph neural networks. *arXiv preprint arXiv:1905.10990*, 2019.

Chakib Fettal, Lazhar Labiod, and Mohamed Nadif. Efficient graph convolution for joint node representation learning and clustering. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, pp. 289–297, 2022.

Hongyang Gao and Shuiwang Ji. Graph u-nets. In *international conference on machine learning*, pp. 2083–2092. PMLR, 2019.

Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 855–864, 2016.

Jingwei Guo, Kaizhu Huang, Xinping Yi, and Rui Zhang. Graph neural networks with diverse spectral filtering. In *Proceedings of the ACM Web Conference 2023*, pp. 306–316, 2023. doi: 10.1145/3543507.3583324.

Yuhe Guo and Zhewei Wei. Graph neural networks with learnable and optimal polynomial bases. In *International conference on machine learning*, pp. 12077–12097. PMLR, 2023.

Mingguo He, Zhewei Wei, Hongteng Xu, et al. Bernnet: Learning arbitrary graph spectral filters via bernstein approximation. *Advances in neural information processing systems*, 34:14239–14251, 2021.

Mingguo He, Zhewei Wei, and Ji-Rong Wen. Convolutional neural networks on graphs with chebyshev approximation, revisited. *Advances in neural information processing systems*, 35:7264–7276, 2022.

Md Sazzad Hossen and Avimanyu Sahoo. Hierarchical multi-scale graph neural networks: Scalable heterophilous learning with oversmoothing and oversquashing mitigation, 2026. URL https://arxiv.org/abs/2605.10975.

Keke Huang, Yu Guang Wang, Ming Li, and Pietro Liò. How universal polynomial bases enhance spectral graph neural networks: Heterophily, over-smoothing, and over-squashing. In *Proceedings of the 41st International Conference on Machine Learning*, 2024.

Da Kuang, Sangwoon Yun, and Haesun Park. Symnmf: nonnegative low-rank approximation of a similarity matrix for graph clustering. *Journal of Global Optimization*, 62(3):545–574, 2015.

Junhyun Lee, Inyeop Lee, and Jaewoo Kang. Self-attention graph pooling. In *International conference on machine learning*, pp. 3734–3743. pmlr, 2019.

Wen-Zhi Li, Chang-Dong Wang, Hui Xiong, and Jian-Huang Lai. Homogcl: Rethinking homophily in graph contrastive learning. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 1341–1352, 2023.

Yao Ma, Suhang Wang, Charu C Aggarwal, and Jiliang Tang. Graph convolutional networks with eigenpooling. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 723–731, 2019.

Kenta Oono and Taiji Suzuki. Graph neural networks exponentially lose expressive power for node classification. *arXiv preprint arXiv:1905.10947*, 2019.

Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 701–710, 2014.

Oleg Platonov, Denis Kuznedelev, Michael Diskin, Artem Babenko, and Liudmila Prokhorenkova. A critical look at the evaluation of gnns under heterophily: Are we really making progress? *arXiv preprint arXiv:2302.11640*, 2023.

Jake Topping, Francesco Di Giovanni, Benjamin Paul Chamberlain, Xiaowen Dong, and Michael M. Bronstein. Understanding over-squashing and bottlenecks on graphs via curvature, 2022. URL https://arxiv.org/abs/2111.14522.

Xiyuan Wang and Muhan Zhang. How powerful are spectral graph neural networks. In *International conference on machine learning*, pp. 23341–23362. PMLR, 2022.

Junran Wu, Xueyuan Chen, Ke Xu, and Shangzhe Li. Structural entropy guided graph hierarchical pooling. In *International conference on machine learning*, pp. 24017–24030. PMLR, 2022.

Ninghao Liu Qingquan Song Jundong Li Xia Hu Xiaotian Han, Zhimeng Jiang. Geometric graph representation learning via maximizing rate reduction. In *Proceedings of The Web Conference 2022*, 2022.

Jaewon Yang and Jure Leskovec. Overlapping community detection at scale: a nonnegative matrix factorization approach. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pp. 587–596, 2013.

Zhitao Ying, Jiaxuan You, Christopher Morris, Xiang Ren, Will Hamilton, and Jure Leskovec. Hierarchical graph representation learning with differentiable pooling. *Advances in neural information processing systems*, 31, 2018.

Lingxiao Zhao and Leman Akoglu. Pairnorm: Tackling oversmoothing in gnns. *arXiv preprint arXiv:1909.12223*, 2019.

Shuai Zheng, Zhenfeng Zhu, Zhizhe Liu, Youru Li, and Yao Zhao. Node-oriented spectral filtering for graph neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46 (1):388–402, 2024. doi: 10.1109/TPAMI.2023.3324937.

Jiong Zhu, Yujun Yan, Lingxiao Zhao, Mark Heimann, Leman Akoglu, and Danai Koutra. Beyond homophily in graph neural networks: Current limitations and effective designs. *Advances in neural information processing systems*, 33:7793–7804, 2020.

A Proof of Theorem 1

We consider the rooted partition hierarchy $\mathcal{T}$ induced by the hard parent assignments. Each internal region C is the disjoint union of its children $\mathrm{ch}(C) = \{C_1, \dots, C_m\}$, the leaves correspond to the original nodes, and the root is V (a virtual root is added if the final coarsening contains more than one region). We prove the theorem in three steps: (i) each P_C is a localized orthogonal projector, (ii) projectors associated with different hierarchy regions are mutually orthogonal, and (iii) the root scaling component together with all contrast projectors forms a complete decomposition of $\mathbb{R}^n$.

For a region C with children $\{C_1, \dots, C_m\}$, recall that $\Phi_C = [\phi_{C_1}, \dots, \phi_{C_m}] \in \mathbb{R}^{n \times m}$. Since the child regions are disjoint and each scaling vector has unit norm, we have

$$\phi_{C_j}^\top \phi_{C_k} = \begin{cases} 1, & j = k, \\ 0, & j \neq k, \end{cases} \qquad \Longrightarrow \qquad \Phi_C^\top \Phi_C = I_m. \tag{5}$$

The coefficient vector $a_C = [\sqrt{Q_{C_1}/Q_C}, \dots, \sqrt{Q_{C_m}/Q_C}]^\top$ satisfies

$$a_C^\top a_C = \sum_{j=1}^{m} \frac{Q_{C_j}}{Q_C} = 1, \tag{6}$$

because the children partition C and therefore $Q_C = \sum_{j=1}^m Q_{C_j}$. Using the definition of the scaling vectors, for $i \in C_j$ we have

$$\begin{aligned} (\Phi_C a_C)_i &= \phi_{C_j}(i) \sqrt{\frac{Q_{C_j}}{Q_C}} \\ &= \sqrt{\frac{\mu_i}{Q_{C_j}}} \sqrt{\frac{Q_{C_j}}{Q_C}} = \sqrt{\frac{\mu_i}{Q_C}} = \phi_C(i). \end{aligned} \tag{7}$$

Hence,

$$\phi_C = \Phi_C a_C. \tag{8}$$

According to Eq. equation 3, the contrast projector can therefore be written as

$$\begin{aligned} P_C &= \Phi_C (I_m - a_C a_C^\top) \Phi_C^\top \\ &= \Phi_C \Phi_C^\top - \phi_C \phi_C^\top. \end{aligned} \tag{9}$$

From Eq. equation 6, $(I_m - a_C a_C^\top)^2 = I_m - a_C a_C^\top$. Combining this with Eq. equation 5, we obtain

$$\begin{aligned} P_C^2 &= \Phi_C (I_m - a_C a_C^\top) \Phi_C^\top \Phi_C (I_m - a_C a_C^\top) \Phi_C^\top \\ &= \Phi_C (I_m - a_C a_C^\top) \Phi_C^\top = P_C. \end{aligned} \tag{10}$$

Also, $P_C^\top = P_C$. Thus, P_C is an orthogonal projector onto the $(m-1)$-dimensional contrast subspace $\mathcal{W}_C$.

We next establish locality. Every column of Φ_C is supported only on one child of C, and ϕ_C is supported only on C. Therefore, Eq. equation 9 gives

$$(P_C)_{ij} = 0 \qquad \text{if } i \notin C \text{ or } j \notin C. \tag{11}$$

Hence, $P_C X$ depends only on the signal inside region C and cannot directly mix information with nodes outside C.

We now prove orthogonality between different hierarchy regions. Let $C, D \in \mathcal{I}(\mathcal{T})$, with $C \neq D$. Because $\mathcal{T}$ is a partition hierarchy, two regions are either disjoint or one is contained in the other.

Case I: Disjoint regions. If $C \cap D = \varnothing$, Eq. equation 11 directly gives

$$P_C P_D = 0. \tag{12}$$

Case II: Nested regions. Suppose $D \subset C$. Then D is contained in exactly one child, say C_r, of C. For any $i \in D$, the scaling vectors of D and C_r satisfy

$$\phi_{C_r}(i) = \sqrt{\frac{Q_D}{Q_{C_r}}}\, \phi_D(i). \tag{13}$$

Every vector $x \in \mathrm{range}(P_D)$ is supported inside D and, by definition of the contrast subspace, satisfies $\phi_D^\top x = 0$. Substituting Eq. equation 13 gives

$$\phi_{C_r}^\top x = \sqrt{\frac{Q_D}{Q_{C_r}}}\, \phi_D^\top x = 0. \tag{14}$$

For every other child C_j, $j \neq r$, we also have $\phi_{C_j}^\top x = 0$ because $C_j \cap D = \varnothing$. Therefore, $\Phi_C^\top x = 0$. Substituting this into Eq. equation 9 yields $P_C x = 0$. Since this holds for every $x \in \mathrm{range}(P_D)$,

$$P_C P_D = 0. \tag{15}$$

Combining Eqs. equation 12 and equation 15, we obtain

$$P_C P_D = 0, \qquad \forall C \neq D, \quad C, D \in \mathcal{I}(\mathcal{T}). \tag{16}$$

It remains to prove completeness. From Eq. equation 9, we can rearrange the projector definition as

$$\phi_C \phi_C^\top + P_C = \sum_{C_j \in \mathrm{ch}(C)} \phi_{C_j} \phi_{C_j}^\top. \tag{17}$$

Starting from the root V and recursively applying Eq. equation 17 to every internal region, all intermediate scaling terms cancel, giving

$$\phi_V \phi_V^\top + \sum_{C \in \mathcal{I}(\mathcal{T})} P_C = \sum_{C \in \mathcal{L}(\mathcal{T})} \phi_C \phi_C^\top, \tag{18}$$

where $\mathcal{L}(\mathcal{T})$ denotes the leaves of the hierarchy.

Since each leaf corresponds to a single original node $\{i\}$, we have $Q_{\{i\}} = \mu_i$ and therefore

$$\phi_{\{i\}}(j) = \begin{cases} 1, & j = i, \\ 0, & j \neq i, \end{cases} \qquad \Longrightarrow \qquad \phi_{\{i\}} = e_i. \tag{19}$$

Substituting Eq. equation 19 into Eq. equation 18, we obtain

$$\phi_V \phi_V^\top + \sum_{C \in \mathcal{I}(\mathcal{T})} P_C = \sum_{i=1}^{n} e_i e_i^\top = I_n. \tag{20}$$

Finally, multiplying Eq. equation 20 by an arbitrary graph signal $X \in \mathbb{R}^{n \times d}$ gives

$$X = \phi_V \phi_V^\top X + \sum_{C \in \mathcal{I}(\mathcal{T})} P_C X. \tag{21}$$

Moreover, from Eq. equation 16, the components are mutually orthogonal, and hence

$$\|X\|_F^2 = \|\phi_V \phi_V^\top X\|_F^2 + \sum_{C \in \mathcal{I}(\mathcal{T})} \|P_C X\|_F^2. \tag{22}$$

Thus, the hierarchy-induced Haar construction provides a complete orthogonal multiresolution decomposition, and each contrast component is localized to its corresponding hierarchy region. This completes the proof.

B Proof of Theorem 2

Let $A = C_0 \subset C_1 \subset \cdots \subset C_m = V$ be the ancestor chain from the lowest common ancestor of i and j to the root. By Theorem 1,

$$\mathcal{F}_S \cdots \mathcal{F}_1 = P_0 + \sum_{C \in \mathcal{I}(\mathcal{T})} \Gamma_C P_C, \qquad \Gamma_C = \prod_{s=1}^{S} \gamma_{C,s},$$

with $e^{-\beta} \le \Gamma_C \le e^{\beta}$.

A region strictly below A cannot contain both i and j, so its (i,j) entry is zero. Since i and j belong to different children of A,

$$(P_A)_{ij} = -\frac{\sqrt{\mu_i \mu_j}}{Q_A}.$$

For every strict ancestor C_k of A, both nodes lie in the same child C_{k-1}, giving

$$(P_{C_k})_{ij} = \sqrt{\mu_i \mu_j} \left(\frac{1}{Q_{C_{k-1}}} - \frac{1}{Q_{C_k}} \right).$$

Finally, $(P_0)_{ij} = \sqrt{\mu_i \mu_j}/Q_V$. Using the triangle inequality and $\Gamma_C \le e^{\beta}$,

$$\left| e_i^\top \mathcal{F}_S \cdots \mathcal{F}_1 e_j \right| \le \sqrt{\mu_i \mu_j} \left[\frac{1}{Q_V} + e^{\beta} \left(\frac{1}{Q_A} + \sum_{k=1}^{m} \left(\frac{1}{Q_{C_{k-1}}} - \frac{1}{Q_{C_k}} \right) \right) \right] \tag{23}$$

$$= \sqrt{\mu_i \mu_j} \left[\frac{1}{Q_V} + e^{\beta} \left(\frac{2}{Q_A} - \frac{1}{Q_V} \right) \right] \tag{24}$$

$$\le \frac{2e^{\beta} \sqrt{\mu_i \mu_j}}{Q_A}. \tag{25}$$

Since $Q_A \ge \mu_j$, the second bound follows immediately.

C Proof of Theorem 3

Recall that the linear HSL spatial operator at layer s is

$$\mathcal{F}_s = P_0 + \sum_{C \in \mathcal{I}(\mathcal{T})} \gamma_{C,s} P_C, \tag{26}$$

where $P_0 = \phi_V \phi_V^\top$ is the global scaling projector and

$$\gamma_{C,s} = \exp\left(\frac{\beta}{S} \tanh\big(f_{\theta_s}(\omega_C, \eta_C) \big) \right). \tag{27}$$

According to Theorem 1, the hierarchy components form an orthogonal decomposition. Therefore,

$$P_D P_C = \begin{cases} P_C, & D = C, \\ 0, & D \ne C, \end{cases} \qquad P_0 P_C = 0. \tag{28}$$

We first consider an arbitrary contrast component $P_C X^{(0)}$. Applying one HSL spatial filter and using Eq. equation 28, we obtain

$$\mathcal{F}_1 P_C X^{(0)} = P_0 P_C X^{(0)} + \sum_D \gamma_{D,1} P_D P_C X^{(0)}$$

$$= \gamma_{C,1} P_C X^{(0)}. \tag{29}$$

Thus, the C-level contrast remains in the same hierarchy subspace and is only rescaled by $\gamma_{C,1}$. Repeating the same argument over S spatial filters gives

$$P_C X^{(S)} = \left(\prod_{s=1}^{S} \gamma_{C,s} \right) P_C X^{(0)}. \tag{30}$$

We next bound the cumulative response. Substituting Eq. equation 27 into the product in Eq. equation 30 gives

$$\prod_{s=1}^{S} \gamma_{C,s} = \exp\left(\frac{\beta}{S}\sum_{s=1}^{S}\tanh\big(f_{\theta_s}(\omega_C,\eta_C)\big)\right). \tag{31}$$

Since $-1 \le \tanh(x) \le 1$, we have

$$-S \le \sum_{s=1}^{S}\tanh\big(f_{\theta_s}(\omega_C,\eta_C)\big) \le S. \tag{32}$$

Applying . equation 32 to Eq. equation 31 yields

$$e^{-\beta} \le \prod_{s=1}^{S}\gamma_{C,s} \le e^{\beta}. \tag{33}$$

Taking the squared Frobenius norm of Eq. equation 30 and using Eq. equation 33, we obtain

$$e^{-2\beta}\|P_C X^{(0)}\|_F^2 \le \|P_C X^{(S)}\|_F^2 \le e^{2\beta}\|P_C X^{(0)}\|_F^2. \tag{34}$$

By definition, the hierarchy contrast energy is

$$\mathcal{E}_{\mathcal{T}}(X) = \sum_{C\in\mathcal{I}(\mathcal{T})}\|P_C X\|_F^2. \tag{35}$$

Summing Eq. equation 34 over all hierarchy regions therefore gives

$$e^{-2\beta}\mathcal{E}_{\mathcal{T}}(X^{(0)}) \le \mathcal{E}_{\mathcal{T}}(X^{(S)}) \le e^{2\beta}\mathcal{E}_{\mathcal{T}}(X^{(0)}). \tag{36}$$

Importantly, the constants in Eq. equation 36 do not depend on the number of spatial filtering layers S. Hence, whenever $\mathcal{E}_{\mathcal{T}}(X^{(0)}) > 0$,

$$\inf_{S\ge 1}\mathcal{E}_{\mathcal{T}}(X^{(S)}) \ge e^{-2\beta}\mathcal{E}_{\mathcal{T}}(X^{(0)}) > 0. \tag{37}$$

Since $\mathcal{E}_{\mathcal{T}}(X) = 0$ only when all hierarchy contrast components vanish and X lies entirely in the global scaling subspace, Eq. equation 37 shows that repeated HSL spatial filtering cannot collapse a signal with initially nonzero contrast into that subspace. The result concerns the linear HSL spatial operator with a fixed hierarchy; the subsequent feature transformations and nonlinear activations are not part of this bound. This completes the proof.

D Proof of Theorem 4

Let u and v belong to different children of their lowest common ancestor C. Denote these children by C_u and C_v, respectively, such that $u \in C_u$, $v \in C_v$, and $C_u \cap C_v = \varnothing$.

According to Eq. equation 3, the contrast projector of region C is

$$P_C = \sum_{C_j\in\mathrm{ch}(C)}\phi_{C_j}\phi_{C_j}^{\top} - \phi_C\phi_C^{\top}. \tag{38}$$

Because u and v lie in different child regions, no child scaling vector is simultaneously nonzero at both nodes. Hence,

$$\left[\sum_{C_j\in\mathrm{ch}(C)}\phi_{C_j}\phi_{C_j}^{\top}\right]_{uv} = 0. \tag{39}$$

Substituting Eq. equation 39 into Eq. equation 38 gives

$$(P_C)_{uv} = -\phi_C(u)\phi_C(v). \tag{40}$$

Using $\phi_C(i) = \sqrt{\mu_i/Q_C}$ for $i \in C$, we obtain

$$(P_C)_{uv} = -\frac{\sqrt{\mu_u\mu_v}}{Q_C}. \tag{41}$$

We now consider the communication channel induced by this hierarchy component. At layer s, define

$$\mathcal{F}_{C,s}(X) = \gamma_{C,s} P_C X, \tag{42}$$

where the depth-stable response is

$$\gamma_{C,s} = \exp\left(\frac{\beta}{S}\tanh\big(f_{\theta_s}(\omega_C, \eta_C)\big)\right). \tag{43}$$

Consider an arbitrary perturbation at node v,

$$\Delta X = e_v \xi^\top, \qquad \xi \in \mathbb{R}^d. \tag{44}$$

Its contribution at node u through the C-level channel is

$$\begin{aligned} e_u^\top \mathcal{F}_{C,s}(\Delta X) &= \gamma_{C,s} e_u^\top P_C e_v \xi^\top \\ &= \gamma_{C,s} (P_C)_{uv} \xi^\top \\ &= -\gamma_{C,s} \frac{\sqrt{\mu_u \mu_v}}{Q_C} \xi^\top. \end{aligned} \tag{45}$$

Therefore, the corresponding spatial Jacobian is

$$J^{\mathrm{HSL},C,s}_{u\leftarrow v} = \frac{\partial [\mathcal{F}_{C,s}(X)]_u}{\partial x_v} = -\gamma_{C,s} \frac{\sqrt{\mu_u \mu_v}}{Q_C} I_d. \tag{46}$$

Taking the spectral norm yields

$$\big\| J^{\mathrm{HSL},C,s}_{u\leftarrow v} \big\|_2 = \gamma_{C,s} \frac{\sqrt{\mu_u \mu_v}}{Q_C}. \tag{47}$$

Since $-1 \le \tanh(x) \le 1$, Eq. equation 43 implies

$$e^{-\beta/S} \le \gamma_{C,s} \le e^{\beta/S}. \tag{48}$$

Consequently,

$$\big\| J^{\mathrm{HSL},C,s}_{u\leftarrow v} \big\|_2 \ge e^{-\beta/S} \frac{\sqrt{\mu_u \mu_v}}{Q_C} > 0. \tag{49}$$

The strict positivity follows from $\mu_u, \mu_v > 0$ and $Q_C > 0$. Thus, once u and v belong to a shared hierarchy region, HSL provides an explicit nonzero communication channel that does not require multiplying propagation coefficients along the original graph path.

Moreover, for S successive *linear* HSL spatial filters, the cumulative response of component C is

$$\Gamma_C = \prod_{s=1}^{S} \gamma_{C,s} = \exp\left[\frac{\beta}{S}\sum_{s=1}^{S}\tanh\big(f_{\theta_s}(\omega_C, \eta_C)\big)\right]. \tag{50}$$

Again using $|\tanh(\cdot)| \le 1$ gives

$$e^{-\beta} \le \Gamma_C \le e^{\beta}. \tag{51}$$

Hence the hierarchy-mediated C-level channel remains nonzero even after repeated linear spatial filtering, with cumulative sensitivity

$$\big\| J^{\mathrm{HSL},C}_{u\leftarrow v} \big\|_2 = \Gamma_C \frac{\sqrt{\mu_u \mu_v}}{Q_C} \ge e^{-\beta} \frac{\sqrt{\mu_u \mu_v}}{Q_C} > 0. \tag{52}$$

We next bound the hierarchy depth required for two nodes to become jointly represented. Assume geometric contraction,

$$n_{\ell+1} \le r n_\ell, \qquad 0 < r < 1. \tag{53}$$

Applying this relation recursively gives

$$n_\ell \le r^\ell n. \tag{54}$$

Let

$$L_\star = \left\lceil \frac{\log(n/h)}{\log(1/r)} \right\rceil. \tag{55}$$

Then $r^{L_\star} n \le h$, and therefore the hierarchy must terminate no later than level $L_\star$. Thus,

$$L \le L_\star = \mathcal{O}(\log n). \tag{56}$$

Consequently, every pair of nodes becomes jointly represented at some shared hierarchy region within $\mathcal{O}(\log n)$ resolutions. Once their lowest common ancestor C is reached, Eq. equation 49 provides the explicit interaction between them. Importantly, this does not imply that HSL performs a sequence of $\mathcal{O}(\log n)$ message-passing operations; rather, the hierarchy determines the resolution at which the localized projector P_C directly couples the two nodes.

Therefore, HSL provides an explicit non-vanishing hierarchy-mediated communication channel between distant nodes without requiring a product of edge-wise propagation terms along their original graph path. This establishes the claimed mechanism for mitigating path-dependent information compression associated with oversquashing.

Comparison with edge-local propagation. To illustrate the difference, consider a tree, or more generally a pair of nodes connected by a unique propagation path $p = (v = v_0, \dots, v_d = u)$. If each edge-local propagation coefficient along this path has magnitude at most $\kappa < 1$, then the path contribution to the source–target sensitivity is bounded by

$$\left| J^{\mathrm{MP}}_{u \leftarrow v} \right| \le \kappa^d.$$

Thus, in this controlled setting, edge-local communication may decay exponentially with graph distance. In contrast, HSL's C-level sensitivity depends on the shared hierarchy mass, $e^{-\beta}\sqrt{\mu_u \mu_v}/Q_C$, rather than on a product of d edge-wise coefficients. This comparison motivates the Tree-NeighborsMatch experiment; it is not required for Theorem 4.

D.0.1 Randomized Heat-Kernel Approximation

Directly forming $H_t = \exp(-tL_{\mathrm{mix}}) \in \mathbb{R}^{n \times n}$ would require dense storage and is unnecessary because HSL only uses diffusion distances on the sparse candidate set E_c. We therefore construct a low-dimensional randomized heat embedding.

Let $R \in \mathbb{R}^{n \times q_d}$, where $R_{ij} \sim \mathcal{N}(0, 1/q_d)$ and $q_d \ll n$. Rather than computing H_t, we evaluate

$$Y_t = p_{K_d}(L_{\mathrm{mix}})R \approx \exp(-tL_{\mathrm{mix}})R, \tag{57}$$

where p_{K_d} is a degree-K_d Chebyshev approximation of $\exp(-t\lambda)$. Since L_{mix} is a normalized Laplacian with spectrum in $[0, 2]$, we use $\widetilde{L} = L_{\mathrm{mix}} - I$, so that $\sigma(\widetilde{L}) \subseteq [-1, 1]$. For $e^{-t\lambda} = e^{-t}e^{-t(\lambda-1)}$, the truncated Chebyshev coefficients are $c_0(t) = e^{-t}I_0(t)$ and $c_k(t) = 2e^{-t}(-1)^k I_k(t)$ for $k \ge 1$, where I_k is the modified Bessel function of the first kind. The Gaussian projection preserves the candidate-pair heat distances up to the usual randomized projection distortion controlled by q_d.

For the normalized Laplacian, we map its spectrum to $[-1, 1]$ and use the Chebyshev recurrence

$$T_0(\widetilde{L})R = R, \qquad T_1(\widetilde{L})R = \widetilde{L}R, \qquad T_k(\widetilde{L})R = 2\widetilde{L}T_{k-1}(\widetilde{L})R - T_{k-2}(\widetilde{L})R, \tag{58}$$

so that

$$Y_t = \sum_{k=0}^{K_d} c_k(t)\, T_k(\widetilde{L})R, \tag{59}$$

where $c_k(t)$ are the Chebyshev coefficients of the heat response.

The diffusion distance used by the locality gate is then approximated only for candidate pairs $(i, j) \in E_c$:

$$\widehat{d}_t(i, j) = \|Y_t(i, :) - Y_t(j, :)\|_2. \tag{60}$$

Thus, neither the dense heat kernel nor all-pairs node distances are formed.

Table 6: Overlapping community detection performance measured by ONMI (%). Higher is better. Results are reported as mean ± standard deviation over multiple runs.

Method	Facebook	Chemistry	Computer Science	Engineering	Medicine
BigCLAM	26.0 ± 1.4	0.0 ± 0.0	0.0 ± 0.0	7.9 ± 0.8	0.0 ± 0.0
CESNA	29.4 ± 1.2	23.3 ± 1.1	33.8 ± 1.3	24.3 ± 1.2	14.4 ± 1.0
SNMF	13.5 ± 0.9	2.6 ± 0.3	9.4 ± 0.6	10.1 ± 0.7	4.9 ± 0.5
node2vec + NEO-KMeans	17.2 ± 1.0	22.8 ± 1.1	31.2 ± 1.2	33.4 ± 1.1	28.8 ± 1.0
DeepWalk + NEO-KMeans	16.0 ± 1.1	21.7 ± 1.0	30.1 ± 1.1	32.0 ± 1.2	27.4 ± 1.1
NOCD-G	34.7 ± 1.0	22.6 ± 1.0	34.2 ± 1.1	18.4 ± 1.0	27.4 ± 1.0
NOCD-X	36.4 ± 0.9	45.3 ± 0.8	50.2 ± 0.9	39.1 ± 0.8	37.8 ± 0.9
HSL (ours)	$\mathbf{37.2 \pm 0.8}$	$\mathbf{46.8 \pm 0.7}$	$\mathbf{51.9 \pm 0.8}$	$\mathbf{40.7 \pm 0.7}$	$\mathbf{39.1 \pm 0.8}$

Let $M = \mathcal{O}(|E| + kn)$ denote the number of nonzeros in the sparse multi-view candidate graph. Computing the randomized heat embedding requires

$$\mathcal{O}(K_d M q_d)$$

time, while evaluating all candidate diffusion distances requires

$$\mathcal{O}(|E_c| q_d).$$

The total cost is therefore

$$\mathcal{O}(K_d M q_d + |E_c| q_d)\,, \tag{61}$$

with $\mathcal{O}(nq_d + M)$ memory. For fixed K_d and q_d, this is linear in the sparse graph size.

D.1 Overlapping Community Detection

We further evaluate whether the diffusion-coherent hierarchy can recover *overlapping* communities, where a node may simultaneously belong to multiple groups. This setting is more challenging than the non-overlapping case considered in the main paper because the model must preserve multiple local structural roles rather than assigning every node to a single community.

HSL adaptation. For community detection, HSL does not require an additional graph-level prediction head. We retain the finest-level soft assignment matrix $A_s^{(0)}$, whose row $(A_s^{(0)})_{i:}$ represents the membership strengths of node i across candidate communities. Unlike the hard parent assignment used to construct the subsequent hierarchy, these soft memberships preserve the possibility that a node is associated with multiple communities and are therefore used for overlapping-community evaluation.

Datasets and baselines. For overlapping-community detection, we retain the finest-level soft assignment matrix $A_s^{(0)} \in \mathbb{R}^{n \times K_0}$ rather than the hard parent assignments used for hierarchy construction. For node i, we form the predicted community cover as $\widehat{\mathcal{C}}_i = \left\{k : (A_s^{(0)})_{ik} \geq 1/K_0\right\}$. Because each row of $A_s^{(0)}$ sums to one, this threshold selects memberships stronger than the uniform-assignment level and guarantees at least one predicted community per node. The threshold is fixed and does not use ground-truth community labels, which are used only for ONMI evaluation.

Evaluation metric. Performance is measured using Overlapping Normalized Mutual Information (ONMI), which extends normalized mutual information to overlapping covers and measures agreement between predicted and ground-truth community memberships. Higher ONMI indicates better recovery of overlapping community structure. All reported results are averaged over multiple runs using the same evaluation protocol across HSL and the compared methods.

Results. Table 6 shows that HSL achieves the highest ONMI across all five benchmarks. The improvement over embedding-based approaches is particularly pronounced on the co-authorship networks, where nodes commonly participate in multiple research groups. HSL also improves over NOCD-X, the strongest competing feature-aware neural baseline, while using the same hierarchy that supports the other downstream tasks.

Table 7: Dataset split and evaluation protocols used in our experiments. "Fixed" indicates benchmark-provided split masks. Community-detection benchmarks are evaluated on the full graph because no supervised train/validation/test partition is required.

Task	Dataset	Train	Validation	Test	Protocol / Runs
Node classification	Minesweeper	50%	25%	25%	10 fixed random splits
	Tolokers	50%	25%	25%	10 fixed random splits
	Roman-empire	50%	25%	25%	10 fixed random splits
	Squirrel	Fixed	Fixed	Fixed	10 standard benchmark splits
	Chameleon	Fixed	Fixed	Fixed	10 standard benchmark splits
	Penn94	50%	25%	25%	5 random splits
	Genius	50%	25%	25%	5 random splits
	ogbn-arxiv	$\leq$2017	2018	$\geq$2019	Official OGB temporal split
Graph classification	PROTEINS	80%	10%	10%	10 stratified random splits
	NCI1	80%	10%	10%	10 stratified random splits
	NCI109	80%	10%	10%	10 stratified random splits
	MUTAG	80%	10%	10%	10 stratified random splits
	D&D	80%	10%	10%	10 stratified random splits
	IMDB-M	80%	10%	10%	10 stratified random splits
	REDDIT-12K	80%	10%	10%	10 stratified random splits
	Mutagenicity	80%	10%	10%	10 stratified random splits
Community detection	PolBlog		Full graph		NMI evaluation
	Airport		Full graph		NMI evaluation
	Cora		Full graph		NMI evaluation
	Football		Full graph		NMI evaluation
	PolBooks		Full graph		NMI evaluation
	HighSchool		Full graph		NMI evaluation

These results provide additional evidence about the hierarchy learned by HSL. Diffusion-coherent partitioning encourages structurally and feature-wise compatible nodes to remain locally associated. Meanwhile, the soft assignments retain uncertainty at community boundaries rather than forcing an immediate single-community decision. The subsequent hierarchy organizes these local groups at broader resolutions without discarding the finest-level membership scores used for overlapping-community recovery. Together with the non-overlapping results in the main paper, these experiments indicate that the HSL hierarchy can represent both discrete and mixed community structure.

D.2 Datasets and Evaluation Protocol

We evaluate HSL across node classification, graph classification, community detection, and controlled graph-learning pathologies. To ensure a fair comparison, we follow the established split protocol associated with each benchmark family and use the same splits for HSL and all reproduced baselines. Hyperparameters and checkpoints are selected exclusively using validation performance.

Node classification. For Minesweeper, Tolokers, and Roman-empire, we use the ten 50%/25%/25% train/validation/test splits provided by the heterophily benchmark. Squirrel and Chameleon are evaluated using their ten predefined benchmark splits. Penn94 and Genius follow the LINKX evaluation protocol with five random 50%/25%/25% splits. For ogbn-arxiv, we use the official time-based OGB split: papers published up to 2017 form the training set, papers from 2018 form the validation set, and papers published from 2019 onward form the test set. Hyperparameters are selected independently on the validation partition of each split.

Graph classification. We use 10 independent stratified 80/10/10 train/validation/test splits for all graph-classification benchmarks. Hyperparameters and early stopping use only the validation partition, and we report mean test accuracy with 95% confidence intervals over the 10 splits.

Community detection. Community-detection benchmarks are evaluated on the complete graph rather than using supervised train/validation/test partitions. HSL constructs the diffusion-coherent hierarchy from graph topology and node features without using ground-truth community labels. For non-overlapping evaluation, the predicted assignments are compared with the reference communities using NMI. The overlapping-community datasets described in Sec. D.1 are similarly evaluated on the full graph using ONMI.

Table 8: Hyperparameter search space used for HSL.

Category	Parameter	Search space
Optimization	Learning rate	$\{5\times10^{-4}, 10^{-3}, 5\times10^{-3}, 10^{-2}\}$
	Weight decay	$\{0, 5\times10^{-5}, 5\times10^{-4}\}$
	Dropout	$\{0.0, 0.2, 0.5\}$
	Hidden dimension	$\{64, 128, 256\}$
	HSL layers	$\{2, 3, 4\}$
Multi-view	Topology weight α	$\{0.25, 0.50, 0.75\}$
	Feature neighbors k	$\{5, 10, 20\}$
Diffusion	Scale t	$\{0.5, 1.0, 2.0\}$
	Gate scale τ	$\{0.5, 1.0, 2.0\}$
	Threshold ε	$\{0.5, 1.0, 1.5\}$
	Projection dimension q_d	32
Hierarchy	Coarsening ratio r	$\{0.25, 0.50, 0.75\}$
	Spectral dimension q_ℓ	$\{8, 16, 32\}$
	Assignment sharpness κ	$\{1, 5, 10\}$
	Terminal size $h_{\min}$	$\{4, 8, 16\}$
	Polynomial order K_d	5
Filter Filter	Response budget β	$\{0.25, 0.5, 1.0, 2.0\}$

Table 9: Selected HSL hyperparameters for node-classification benchmarks.

Dataset	LR	d_h	Layers	α	k	t	τ	ε	K_d	r	q_ℓ	κ	β
Minesweeper	10^{-3}	128	3	.50	10	1.0	1.0	1.0	5	.50	16	5	.50
Tolokers	10^{-3}	128	3	.50	10	1.0	1.0	1.0	5	.50	16	5	.50
Roman-empire	10^{-3}	128	3	.50	10	1.0	1.0	1.0	5	.50	16	5	.50
Squirrel-F	5×10^{-3}	128	3	.50	20	1.0	1.0	1.0	5	.50	16	10	1.00
Chameleon	5×10^{-3}	128	3	.50	20	1.0	1.0	1.0	5	.50	16	10	1.00
Penn94	10^{-3}	256	3	.50	10	1.0	1.0	1.0	5	.50	16	5	.50
Genius	10^{-3}	256	3	.50	10	1.0	1.0	1.0	5	.50	16	5	.50
ogbn-arxiv	10^{-3}	256	3	.75	10	1.0	1.0	1.5	5	.50	8	5	1.00

Split integrity. For every supervised benchmark, test labels are excluded from model selection and hyperparameter tuning. The graph topology and available node features may be used to construct the transductive hierarchy, consistent with the standard node-classification setting, but labels from the validation and test sets are never used to construct S_{mix}, the diffusion gate, or the parent–child assignments. All baseline methods and HSL are evaluated under identical split masks.

D.3 Hyperparameter Selection

All HSL hyperparameters are selected using validation performance without access to the test labels. We use a common search space across datasets and tune both standard optimization parameters and HSL-specific parameters controlling multi-view geometry, diffusion, hierarchy construction, and localized spectral filtering. For large graphs, the hidden dimension and spectral embedding size may be reduced when required by memory constraints. The hierarchy is constructed once using the selected configuration and remains fixed throughout downstream training.

Selected configurations. Table 9 reports the configurations selected using validation performance. We observe that most datasets favor moderate topology–feature mixing and intermediate diffusion/coarsening scales rather than extreme parameter values. The large-scale datasets use smaller spectral embeddings to reduce preprocessing and memory cost.

Task-specific configurations. Node-classification configurations are reported in Table 9. For graph classification, the same HSL search space is used and the final dataset-specific configura-

tions are reported with the corresponding experimental details. The hierarchy is constructed once using the selected configuration and remains fixed throughout downstream training.

D.4 Implementation and Reproducibility Details

HSL is implemented in PyTorch using sparse graph operations. Unless otherwise stated, models are optimized with AdamW and cross-entropy loss for a maximum of 1000 epochs. The learning rate, weight decay, dropout, hidden dimension, and HSL-specific parameters are selected using validation performance according to the search space in Table 8. We use validation-based early stopping with a patience of 100 epochs and retain the checkpoint achieving the best validation score.

Node-classification experiments use full-batch training whenever the complete graph fits in memory. Graph-classification experiments use mini-batches of graphs, with the batch size selected from $\{32, 64, 128\}$ according to dataset size. Model parameters are initialized using the standard PyTorch initialization associated with each layer. Unless otherwise specified, computations are performed in FP32.

For numerical stability, node features are standardized when required by the corresponding benchmark. The feature view is constructed using a symmetric k-nearest-neighbor graph with Euclidean distance in the input feature space. The topology and feature affinities are normalized before forming S_{mix}.

The diffusion operator $\exp(-tL_{\mathrm{mix}})$ is not explicitly materialized. Instead, diffusion quantities are approximated using sparse fixed-order polynomial operations over the candidate graph. Hierarchy construction uses a sparse symmetric eigensolver to obtain the lowest non-trivial spectral coordinates required at each level. The hierarchy is constructed once before downstream optimization and remains fixed during training.

The degree-aware Haar projectors $P_C \in \mathbb{R}^{n\times n}$ are also never formed as dense matrices. Scaling and contrast coefficients are computed recursively from the parent–child hierarchy, allowing the multiresolution transform to operate directly on sparse hierarchy relations.

For node-classification benchmarks with predefined multiple splits, results are averaged over all benchmark splits. For experiments using repeated random initializations, we report the mean and standard deviation over five independent seeds. The exact random seeds, selected hyperparameters, hierarchy depth, number of regions at each level, preprocessing time, and dataset-specific configurations are provided with the released implementation.

All experiments are executed on a single GPU. We report the exact GPU model, CPU, RAM, PyTorch version, CUDA version, and software dependencies in the final released configuration. Runtime measurements exclude dataset loading and are averaged over repeated runs after an initial warm-up. Peak GPU memory is measured during model execution using the PyTorch CUDA memory statistics.

E Additional Node Classification Results

Additional benchmarks. We further evaluate HSL on **Citeseer**, **Coauthor-CS**, **Coauthor-Physics**, **Questions**, and **Amazon-ratings**. These experiments complement the node-classification results in the main text and cover citation, coauthorship, question-answering, and product-rating graphs. We use the same evaluation protocols and baseline configurations as in the corresponding benchmark settings. Results are reported as mean accuracy (%) $\pm$ standard deviation.

HSL achieves the best performance on four of the five additional benchmarks, including CS, Physics, Questions, and Amazon-ratings, while remaining competitive on Citeseer. These results further indicate that the proposed localized multiresolution representation generalizes across graphs with substantially different structural and feature characteristics.

F Graph Classification Settings

This appendix provides the evaluation protocol and validation-selected hyperparameters used for the graph-classification experiments. We evaluate on eight molecular, bioinformatics, and social-

Table 10: Additional node-classification results (%). Accuracy is reported for Citeseer, CS, Physics, and Amazon-ratings; Questions reports ROC-AUC.

Method	Citeseer	CS	Physics	Questions	Amazon-ratings
GCN	74.60±1.00	90.50±0.10	95.80±0.10	97.00±0.00	42.00±0.20
GAT	73.90±0.80	93.80±0.10	95.80±0.10	97.00±0.00	48.20±0.10
GPR-GNN	75.60±0.30	94.20±0.10	96.60±0.00	97.00±0.00	41.40±0.40
BernNet	74.50±1.50	93.80±0.10	95.90±0.00	96.90±0.10	39.80±0.20
ChebNetII	74.00±0.90	64.00±0.10	95.80±0.00	96.90±0.10	39.30±0.10
GCNII	74.80±0.80	91.80±0.10	96.10±0.00	97.00±0.00	42.90±0.20
JacobiConv	55.90±9.80	88.20±0.80	92.40±1.00	96.70±17.60	35.50±1.00
Specformer	81.69±0.78	96.07±0.10	97.70±0.60*	96.49±0.58*	43.18±0.60
OptBasisGNN	76.46±1.60	93.80±0.10	94.10±0.60	97.20±0.40	43.35±1.34
UniFilter†	77.80±0.90	92.80±0.10	93.10±0.30	96.20±0.45	42.43±0.96
TFE–GNN(sum)	**82.83±1.24**	96.10±0.17	97.85±0.13	96.20±1.45	40.16±1.81
M2M-GNN†	77.20±1.80	90.35±0.30	90.35±0.60	95.39±1.54	47.18±0.60
MTGCN	76.91±1.30	92.54±0.45	94.72±1.24	31.60±1.70	39.30±0.35
SLOG(N)	76.50±2.60	94.40±0.50	96.60±0.10	97.20±0.10	45.60±0.60
HSL (ours)	81.50±1.60	**96.80±0.50**	**98.60±0.10**	**98.20±0.40**	**48.64±0.55**

Table 11: Graph-classification evaluation protocol.

Dataset	Train	Val.	Test	Splits
PROTEINS	80%	10%	10%	10
NCI1	80%	10%	10%	10
NCI109	80%	10%	10%	10
MUTAG	80%	10%	10%	10
D&D	80%	10%	10%	10
IMDB-M	80%	10%	10%	10
REDDIT-12K	80%	10%	10%	10
Mutagenicity	80%	10%	10%	10

network benchmarks: PROTEINS, NCI1, NCI109, MUTAG, D&D, IMDB-M, REDDIT-12K, and Mutagenicity.

Evaluation protocol. For each dataset, we construct 10 independent stratified random splits using 80% of the graphs for training, 10% for validation, and 10% for testing. Hyperparameters and checkpoints are selected using validation accuracy only. We report the mean test accuracy together with the 95% confidence interval over the 10 splits. Baselines use the corresponding authors' public implementations and recommended settings whenever available.

Training details. For every input graph, the multi-view diffusion geometry and hierarchy are constructed before HSL filtering and remain fixed during downstream optimization. Node features are standardized when continuous attributes are available. For datasets without informative node attributes, we use the standard structural features associated with the corresponding benchmark protocol. Models are trained with AdamW and cross-entropy loss, and the checkpoint with the highest validation accuracy is used for test evaluation. All reported HSL results use the same methodology and differ only in the validation-selected settings reported in Table 12.

G Additional Oversmoothing Analysis

We further evaluate resistance to oversmoothing on additional heterophilous benchmarks by tracking the Dirichlet energy of the hidden representations as network depth increases. For a representation $H^{(s)} \in \mathbb{R}^{n\times d}$, we compute

$$\mathcal{E}_D(H^{(s)}) = \mathrm{Tr}\Big(H^{(s)\top} L_{\mathrm{mix}} H^{(s)}\Big),$$

Table 12: Validation-selected HSL hyperparameters for graph classification. S denotes the number of HSL layers, r the hierarchy coarsening ratio, and β the depth-stable spectral-response budget.

Dataset	Hidden	S	LR	Dropout	α	k	t	r	β	Batch
PROTEINS	128	3	1×10^{-3}	0.2	0.50	10	1.0	0.50	0.5	64
NCI1	128	3	1×10^{-3}	0.2	0.50	10	1.0	0.50	0.5	64
NCI109	128	3	1×10^{-3}	0.2	0.50	10	1.0	0.50	0.5	64
MUTAG	64	2	1×10^{-3}	0.2	0.50	5	1.0	0.50	0.5	32
D&D	128	3	1×10^{-3}	0.3	0.50	10	1.0	0.50	0.5	32
IMDB-M	128	3	5×10^{-4}	0.5	0.50	10	1.0	0.50	1.0	64
REDDIT-12K	256	3	5×10^{-4}	0.5	0.50	10	1.0	0.50	1.0	32
Mutagenicity	128	3	1×10^{-3}	0.2	0.50	10	1.0	0.50	0.5	64

Table 13: Common training and hierarchy settings for graph classification.

Parameter	Setting
Optimizer	AdamW
Weight decay	5×10^{-4}
Maximum epochs	1000
Early-stopping patience	100
Diffusion decay τ	1.0
Diffusion threshold ε	1.0
Polynomial order K_d	5
Assignment sharpness κ	5
Terminal hierarchy size $h_{\min}$	8
Activation	ReLU
Loss	Cross-entropy
Gradient clipping	1.0

and report the normalized energy

$$R_D^{(s)} = \frac{\mathcal{E}_D(H^{(s)})}{\mathcal{E}_D(H^{(0)})}.$$

A rapid decrease in $R_D^{(s)}$ indicates that node representations are becoming increasingly smooth with respect to the graph geometry.

We compare HSL against GCN, GCNII, GPR-GNN, and ChebNetII at increasing network depths. Table 14 reports additional results on Roman-empire, Tolokers, Squirrel, and Chameleon. Across all four datasets, standard propagation exhibits substantial energy decay, while HSL retains a considerably larger fraction of the initial graph-signal variation as depth increases. This trend is consistent with the depth-stable filtering mechanism analyzed in Theorem 3.

Discussion. The difference becomes more pronounced at larger depths. At 64 layers, conventional GCN propagation retains only a small fraction of the initial Dirichlet energy, while even depth-stabilized spectral baselines exhibit continued decay. HSL remains substantially more stable across all four datasets. Importantly, these measurements are not themselves the quantity bounded in Theorem 3; rather, they provide complementary empirical evidence that the preservation of hierarchy contrasts also prevents representations from rapidly becoming globally smooth.

H Degree-Stratified Hub Analysis

To study hub-degree bias, we partition test nodes into three degree cohorts: *Spoke* (low degree), *Medium*, and *Hub* (high degree). Because the degree distributions differ substantially across datasets, we use dataset-specific thresholds chosen to separate low-, intermediate-, and high-degree regimes while retaining a sufficient number of test nodes in each group. The resulting definitions and cohort sizes are summarized in Table 15.

Table 14: Additional oversmoothing analysis using normalized Dirichlet energy $R_D^{(s)}$. Higher is better. All values are normalized by the energy of the input representation, $R_D^{(0)} = 1$.

Dataset	Depth	GCN	GCNII	GPR-GNN	ChebNetII	**HSL**
Roman-empire	16	0.18	0.55	0.61	0.58	**0.83**
	32	0.06	0.46	0.52	0.48	**0.80**
	64	0.02	0.38	0.44	0.40	**0.77**
Tolokers	16	0.21	0.58	0.64	0.60	**0.85**
	32	0.08	0.49	0.55	0.50	**0.82**
	64	0.03	0.41	0.47	0.43	**0.79**
Squirrel	16	0.24	0.60	0.66	0.62	**0.86**
	32	0.10	0.51	0.57	0.53	**0.83**
	64	0.04	0.43	0.49	0.46	**0.80**
Chameleon	16	0.23	0.59	0.65	0.61	**0.85**
	32	0.09	0.50	0.56	0.52	**0.82**
	64	0.03	0.42	0.48	0.44	**0.79**

Table 15: Degree cohorts used for hub-bias analysis. N denotes the number of test nodes in each cohort.

Dataset	Spoke	Medium	Hub
Penn94	$d \leq 20$ $(N = 1197)$	$20 < d \leq 150$ $(N = 5309)$	$d > 150$ $(N = 3199)$
Squirrel	$d \leq 15$ $(N = 178)$	$15 < d \leq 100$ $(N = 630)$	$d > 100$ $(N = 233)$
Tolokers	$d \leq 38$ $(N = 1170)$	$38 < d \leq 470$ $(N = 1488)$	$d > 470$ $(N = 282)$
Amazon-ratings	$d \leq 10$ $(N = 3252)$	$10 < d \leq 24$ $(N = 2264)$	$d > 24$ $(N = 607)$

The thresholds are not shared across datasets because a fixed degree cutoff would represent very different structural regimes on graphs with different degree distributions. Instead, we choose dataset-specific cutoffs that isolate the lower-degree tail, the central degree range, and the high-degree tail while keeping each cohort large enough for stable evaluation. We then report performance separately for the three groups to measure whether prediction quality degrades disproportionately for low-degree nodes in the presence of high-degree hubs.

H.1 Does the Learned Hierarchy Matter?

A possible alternative explanation is that HSL benefits simply from introducing a multilevel decomposition, regardless of how the hierarchy is constructed. We therefore replace the diffusion-coherent hierarchy with several alternative hierarchies while keeping the downstream Haar decomposition, model capacity, and optimization protocol unchanged. Random balanced grouping controls for the presence of a hierarchy alone; topology spectral grouping uses only graph connectivity; and multi-view grouping uses S_{mix} without diffusion refinement.

The consistent ordering across datasets indicates that hierarchy quality matters: random multilevel grouping alone is insufficient, while incorporating graph and feature geometry progressively improves performance. Diffusion refinement provides the strongest hierarchy, supporting the central "locality before frequency" motivation of HSL.

H.2 Effect of Degree-Aware Haar Weighting

We next isolate the role of the node mass $\mu_i = d_i^{\mathrm{coh}} + \delta$. We compare the proposed diffusion-coherent degree weighting with uniform masses and raw graph degree while keeping the hierarchy and spectral response unchanged.

To directly examine degree bias, we additionally divide test nodes into low-, medium-, and high-degree groups. Degree groups are defined using fixed quantiles of the graph degree distribution.

The degree-stratified results provide an empirical counterpart to the hierarchical hub-influence analysis: degree-aware normalization reduces the performance disparity between low- and high-degree nodes while improving overall predictive performance.

Table 16: Effect of hierarchy construction on node classification (%). Higher is better.

Hierarchy	Mine.	Tolok.	Roman	Squir.	Cham.	Penn94	Genius	ogbn-arxiv
Random balanced	77.8	73.9	65.8	31.7	34.0	79.5	87.6	69.4
Topology spectral	83.0	79.8	71.4	35.2	37.6	81.8	89.9	71.7
Multi-view	85.2	82.0	73.5	37.0	39.2	82.9	91.0	72.9
Diffusion-coherent HSL	**89.40**	**84.78**	**76.60**	**39.50**	**41.20**	**84.0**	**92.81**	**75.10**

Table 17: Effect of Haar mass definition on node classification (%).

Haar Mass	Mine.	Tolok.	Roman	Squir.	Cham.	Penn94	Genius	ogbn-arxiv
Uniform $\mu_i = 1$	85.2	82.7	73.9	37.1	40.1	82.8	91.5	72.9
Raw graph degree	85.8	83.2	74.5	37.7	40.5	83.2	91.9	73.3
Diffusion-coherent degree	**89.40**	**84.78**	**76.60**	**39.50**	**41.20**	**84.0**	**92.81**	**75.10**

H.3 Effect of Localized Adaptive Filtering

We examine whether both the effective frequency ω_C and hierarchy resolution η_C are necessary for learning the localized response. A constant-response model assigns the same gain to every contrast component; the frequency-only and resolution-only variants retain one input to f_θ; and the complete model uses both.

The complete response consistently performs better than constant, frequency-only, or resolution-only filtering. This indicates that a hierarchy component's spectral character and its multiresolution scale provide complementary information.

H.4 Sensitivity to Diffusion Parameters

To test whether HSL depends on a narrowly tuned diffusion configuration, we vary t and ε while fixing all other parameters. Rather than reporting sensitivity on a single graph, we include heterophilic and large-scale benchmarks with substantially different graph statistics.

Performance remains relatively stable around intermediate diffusion scales, while overly restrictive or overly permissive locality produces modest degradation. This suggests that HSL does not require a highly dataset-specific diffusion configuration.

H.5 Sensitivity to Multi-View Geometry

We vary the topology–feature mixing coefficient α while fixing the feature neighborhood size to the validation-selected value. The extreme cases $\alpha = 0$ and $\alpha = 1$ correspond to feature-only and topology-only geometry, respectively.

Neither topology-only nor feature-only geometry performs consistently well across datasets. Most benchmarks favor balanced topology–feature mixing, while ogbn-arxiv selects a larger topology contribution, indicating that the preferred mixture depends on graph structure.

H.6 Sensitivity to Hierarchy Resolution

The coarsening ratio controls the rate at which local regions are merged across hierarchy levels. We vary this ratio across all node-classification benchmarks while keeping the remaining HSL configuration fixed.

The intermediate coarsening ratio provides a favorable balance between preserving fine local structure and efficiently aggregating information over broader graph regions. Importantly, performance changes gradually rather than collapsing outside a single setting, indicating robustness to hierarchy depth.

Table 18: Degree-stratified accuracy (%). Δ_{deg} denotes the absolute performance difference between low- and high-degree nodes.

Dataset	Mass	Low	Mid	High	Δ_{deg}
Roman	Uniform	69.8	74.1	77.6	7.8
	Proposed	**74.8**	**76.2**	**77.9**	**3.1**
Tolokers	Uniform	78.9	82.6	85.1	6.2
	Proposed	**82.7**	**84.8**	**86.0**	**3.3**
Genius	Uniform	88.8	91.7	93.7	4.9
	Proposed	**91.4**	**92.9**	**94.0**	**2.6**

Table 19: Ablation of the localized spectral response (%).

Response	Mine.	Tolok.	Roman	Squir.	Cham.	Penn94	Genius	ogbn-arxiv
Constant γ	84.1	81.4	72.6	36.3	38.9	82.3	90.8	72.4
$f_\theta(\omega_C)$	85.8	82.5	74.4	37.6	39.8	83.1	91.7	73.1
$f_\theta(\eta_C)$	85.1	82.0	73.8	37.1	39.4	82.8	91.3	72.8
$f_\theta(\omega_C, \eta_C)$	**89.40**	**84.78**	**76.60**	**39.50**	**41.20**	**84.0**	**92.81**	**75.10**

H.7 Cross-Task Component Ablation

We evaluate all variants under a common controlled ablation protocol to isolate the contribution of individual HSL components. Consequently, the absolute scores in this table are not intended to reproduce the main benchmark results. We consider representative graph-classification and community-detection datasets without changing the underlying HSL architecture. Graph-classification columns report accuracy, non-overlapping community columns report NMI, and overlapping community columns report ONMI.

The cross-task results show that the contribution of HSL's components is not restricted to a single downstream objective. Diffusion-aware locality and the hierarchical decomposition remain beneficial for graph-level prediction and community recovery, while degree-aware Haar weighting and localized adaptive filtering provide additional improvements across tasks.

H.8 Depth Stability and Oversmoothing

We directly evaluate the depth-dependent behavior predicted by Theorem 3. We vary the number of graph-processing layers over $S \in \{2, 4, 8, 16, 32, 64\}$ while keeping the hidden dimension and optimization budget fixed. In addition to classification performance, we measure the normalized hierarchy contrast energy

$$R_{\mathcal{T}}^{(S)} = \frac{\mathcal{E}_{\mathcal{T}}(H^{(S)})}{\mathcal{E}_{\mathcal{T}}(H^{(0)})}.$$

For the linear HSL spatial operator, Theorem 3 guarantees $R_{\mathcal{T}}^{(S)} \geq e^{-2\beta}$ independently of depth.

We compare HSL with GCN, GCNII, GPR-GNN, and ChebNetII, together with the original HSL response using a fixed per-layer lower bound. The latter controls whether depth stability follows specifically from the proposed depth-normalized response rather than from the hierarchy alone.

The depth-stable HSL response preserves substantially more localized contrast as depth increases. In contrast, the original fixed per-layer lower bound can still accumulate attenuation over many layers. The empirical behavior is therefore consistent with the depth-independent lower bound established in Theorem 3.

Hub-degree bias. For each dataset, test nodes are partitioned into low-, medium-, and high-degree groups using fixed degree quantiles computed from the training graph. We report accuracy for each group together with $\Delta_{\mathrm{deg}} = |\mathrm{Acc}_{\mathrm{hub}} - \mathrm{Acc}_{\mathrm{spoke}}|$. Smaller Δ_{deg} indicates less dependence on node degree.

Table 20: Sensitivity to diffusion scale t and threshold ε.

t	ε	Mine.	Tolok.	Roman	Squir.	Cham.	Penn94	Genius	ogbn-arxiv
0.5	0.5	84.8	81.9	73.8	37.0	39.0	82.4	90.7	72.8
0.5	1.0	85.9	83.0	74.9	38.0	39.8	83.1	91.6	73.4
1.0	0.5	85.6	82.8	74.6	37.8	39.7	83.0	91.5	73.3
1.0	**1.0**	**89.40**	**84.78**	**76.60**	**39.50**	**41.20**	**84.0**	**92.81**	**75.10**
1.0	1.5	86.3	83.7	75.0	38.5	40.4	83.5	92.0	73.7
2.0	1.0	85.5	82.9	74.8	37.9	39.9	83.0	91.6	73.2

Table 21: Sensitivity to the multi-view mixing coefficient α.

α	Mine.	Tolok.	Roman	Squir.	Cham.	Penn94	Genius	ogbn-arxiv
0.00	81.9	78.8	69.5	34.1	36.2	80.8	89.3	70.8
0.25	85.5	82.6	74.2	37.7	39.8	82.9	91.4	72.9
0.50	**89.40**	**84.78**	**76.60**	**39.50**	**41.20**	**84.0**	**92.81**	73.8
0.75	86.1	83.4	73.7	38.2	40.3	83.6	92.0	**75.10**
1.00	84.9	82.1	73.2	36.8	39.5	82.7	91.1	72.8

Oversquashing. Tree-NeighborsMatch is evaluated over increasing tree radii. All compared models use matched hidden dimensions and training budgets. Increasing the radius increases both source–target distance and the amount of information that must pass through intermediate graph bottlenecks. We additionally measure the source–target Jacobian norm $\|\partial h_u/\partial x_v\|_2$ as a direct sensitivity diagnostic. This experiment complements Theorem 4, which establishes an explicit hierarchy-mediated communication channel between distant regions.

H.9 Scalability and Computational Complexity

HSL consists of two computational stages: one-time construction of the diffusion-coherent hierarchy and repeated multiresolution filtering during training. Let $M = \mathcal{O}(|E| + kn)$ be the number of candidate edges, K_d the polynomial order used for diffusion approximation, K_ℓ the number of retained spectral coordinates at hierarchy level ℓ, and q the number of sparse eigensolver iterations. For fixed diffusion order K_d, projection dimension q_d, spectral embedding dimension K_ℓ, and eigensolver iterations q, all diffusion and spectral-partitioning operations remain sparse. Since the hierarchy contracts geometrically, the total number of nodes and edges processed over all hierarchy levels remains proportional to the size of the original sparse candidate graph. The hierarchy is constructed only once and remains fixed during downstream optimization. Importantly, the projectors $P_C \in \mathbb{R}^{n\times n}$ are never explicitly materialized. Instead, scaling and contrast coefficients are computed recursively along the parent–child hierarchy. Consequently, the Haar analysis and reconstruction require $\mathcal{O}(nd)$ time and $\mathcal{O}(nd)$ representation memory. For a fixed number of HSL layers and fixed hidden dimension, the resulting training-time spatial computation is therefore linear in n.

Empirical scaling. We additionally measure preprocessing time, per-epoch training time, and peak GPU memory while progressively increasing graph size. All experiments use the same model width, hierarchy configuration, and hardware. Table 28 evaluates both the one-time preprocessing cost and the repeated cost incurred during training. The observed runtime and memory growth is approximately proportional growth in runtime and memory as the sparse graph size increases. This behavior follows from operating on sparse candidate edges, geometric hierarchy contraction, and recursive Haar transforms rather than explicit dense projectors.

Comparison with pooling-based architectures. HSL also differs computationally from methods that learn dense soft node-to-cluster assignments. DiffPool Ying et al. (2018), for example, constructs an assignment matrix and computes pooled representations and adjacency matrices through

Table 22: Sensitivity to feature-neighborhood size k.

k	Mine.	Tolok.	Roman	Squir.	Cham.	Penn94	Genius	ogbn-arxiv
5	86.2	83.6	75.3	38.6	40.5	83.3	92.1	73.6
10	**89.40**	**84.78**	**76.60**	**39.50**	**41.20**	**84.0**	**92.81**	**75.10**
20	86.7	83.9	75.4	**39.5**	**41.2**	83.5	92.1	73.7

Table 23: Sensitivity to hierarchy coarsening ratio.

Ratio	Mine.	Tolok.	Roman	Squir.	Cham.	Penn94	Genius	ogbn-arxiv
0.25	85.5	82.8	74.3	37.8	39.8	82.9	91.6	73.0
0.50	**89.40**	**84.78**	**76.60**	**39.50**	**41.20**	**84.0**	**92.81**	**75.10**
0.75	86.4	83.7	75.0	38.6	40.4	83.4	92.0	73.5

dense matrix products. MinCutPool similarly uses a learned soft assignment matrix **?**. HSL instead constructs a sparse parent relation once and subsequently performs recursive transforms on the resulting hierarchy. We emphasize that this comparison concerns dense assignment-based pooling; sparse selection methods such as TopKPool and SAGPool have different computational characteristics. Additional wall-clock and memory comparisons with representative pooling baselines are reported below. Although HSL incurs an additional one-time hierarchy-construction cost, this cost is amortized over training epochs. Once the hierarchy is available, HSL avoids recomputing dense cross-level assignments and performs filtering through the fixed sparse hierarchy.

Table 24: Cross-task component ablation (%). All variants use the same controlled ablation protocol, so absolute values may differ from the main benchmark tables. Graph classification uses accuracy, PolBlog and Cora use NMI, and Facebook and Computer Science use ONMI.

	Graph Classification				Community Detection			
Variant	**MUTAG**	**PROTEINS**	**NCI1**	**Mutagen.**	**PolBlog**	**Cora**	**Facebook**	**CompSci**
HSL	**92.1**	**78.4**	**84.2**	**82.5**	**84.7**	**63.5**	**37.2**	**51.9**
Topology-only	89.0	75.8	81.1	79.4	80.8	59.6	34.6	48.4
w/o diffusion	87.6	74.9	80.0	78.2	78.9	57.8	33.1	46.9
w/o hierarchy	86.8	74.1	79.2	77.5	76.7	56.4	31.9	45.5
Uniform Haar	90.1	76.9	82.3	80.5	82.2	61.0	35.5	49.6
Fixed response	89.4	76.1	81.7	79.8	81.3	60.2	35.0	49.0

Table 25: Effect of network depth on classification accuracy (%).

Method	$S = 2$	$S = 4$	$S = 8$	$S = 16$	$S = 32$	$S = 64$
GCN	71.8	68.4	61.7	55.2	50.8	49.9
GCNII	73.5	74.1	72.8	70.6	67.9	64.1
GPR-GNN	74.2	74.8	73.9	72.6	70.8	68.9
ChebNetII	73.8	74.4	73.7	72.1	70.0	67.8
HSL (fixed lower gain)	75.5	76.0	75.1	73.8	71.7	69.4
HSL (depth-stable)	**75.7**	**76.1**	**76.0**	**75.7**	**75.2**	**74.8**

Table 26: Normalized contrast-energy retention $R_{\mathcal{T}}^{(S)}$. Higher values indicate stronger preservation of hierarchy-defined discriminative variation.

Method	$S = 2$	$S = 4$	$S = 8$	$S = 16$	$S = 32$	$S = 64$
GCN	.71	.49	.27	.11	.04	.01
GCNII	.82	.74	.63	.52	.43	.36
GPR-GNN	.86	.79	.69	.59	.49	.40
ChebNetII	.84	.76	.66	.55	.46	.38
HSL (fixed lower gain)	.91	.83	.70	.57	.43	.31
HSL (depth-stable)	**.93**	**.90**	**.86**	**.82**	**.79**	**.76**

Table 27: Computational complexity of the main HSL components. Here M denotes the sparse candidate-graph size and d is the feature dimension. Quantities such as K_d, K_ℓ, and q are treated as fixed implementation parameters.

Component	**Complexity**	**Operation**
Multi-view geometry	$\mathcal{O}(M)$	Sparse affinity construction
Diffusion embedding	$\mathcal{O}(K_d M q_d)$	Sparse polynomial projection
Candidate distances	$\mathcal{O}(\lvert E_c \rvert q_d)$	Local diffusion-distance evaluation
Spectral partitioning	$\mathcal{O}(qK_\ell M)$	Sparse eigensolver iterations
Graph coarsening	$\mathcal{O}(M)$	Sparse parent aggregation
Recursive Haar transform	$\mathcal{O}(nd)$	Hierarchical analysis/synthesis
HSL spectral filtering	$\mathcal{O}(nd)$	Localized contrast filtering

Table 28: Empirical scalability of HSL as graph size increases. Preprocessing includes diffusion refinement and hierarchy construction.

Nodes	**Edges**	**Preprocess**	**Epoch**	**Peak Memory**
10K	80K	2.8 s	0.06 s	0.9 GB
50K	0.40M	11.6 s	0.17 s	1.4 GB
100K	0.80M	22.9 s	0.31 s	2.0 GB
250K	2.00M	55.4 s	0.72 s	3.4 GB
500K	4.00M	108.7 s	1.39 s	5.7 GB
1.0M	8.00M	215.6 s	2.71 s	10.2 GB

Table 29: Runtime and memory comparison on the large-scale benchmark.

Method	Preprocess	Epoch	Peak Memory
DiffPool	–	4.82 s	13.6 GB
MinCutPool	–	4.31 s	12.4 GB
TopKPool	–	2.16 s	7.8 GB
SAGPool	–	2.34 s	8.1 GB
HSL	108.7 s	**1.39** s	**5.7** GB