High-Pass Matters: Theoretical Insights and Sheaflet-Based Design for Hypergraph Neural Networks

Anonymous submission

Abstract

Hypergraph neural networks (HGNNs) have shown great potential in modeling higher-order relationships among multiple entities. However, most existing HGNNs primarily emphasize low-pass filtering while neglecting the role of high-frequency information. In this work, we present a theoretical investigation into the spectral behavior of HGNNs and prove that combining both low-pass and high-pass components leads to more expressive and effective models. Notably, our analysis highlights that high-pass signals play a crucial role in capturing local discriminative structures within hypergraphs. Guided by these insights, we propose a novel sheaflet-based HNNs that integrates cellular sheaf theory and framelet transforms to preserve higher-order dependencies while enabling multi-scale spectral decomposition. This framework explicitly emphasizes high-pass components, aligning with our theoretical findings. Extensive experiments on benchmark datasets demonstrate the superiority of our approach over existing methods, validating the importance of high-frequency information in hypergraph learning.

1 Introduction

Unlike traditional graphs that capture only pairwise relationships, hypergraphs provide a more expressive framework for modeling higher-order interactions among multiple entities, as supported by both theoretical and empirical evidence in (Wang and Kleinberg 2024; Millán et al. 2025). This richer representational capacity allows hypergraphs to naturally encode complex and multi-way dependencies, making them especially suitable for real-world applications characterized by intricate relational structures (Antelmi et al. 2023).

Recent advances in hypergraph neural networks (HGNNs) have extended spectral and message-passing techniques from graphs to hypergraphs, enabling effective learning over higher-order structures (Kim et al. 2024; Gao et al. 2024). Despite these developments, the spectral design of HGNNs remains largely underexplored, particularly in terms of frequency components. While graph neural networks have begun to incorporate both low-pass and high-pass filtering mechanisms (Bo et al. 2021; Zheng et al. 2021; Li et al. 2024), analogous efforts in the hypergraph setting remain sparse. Most existing HGNNs either implicitly favor low-frequency propagation or neglect the role of high-frequency signals altogether. This gap leaves open a fundamental question: ***To what extent do high-pass components influence the expressive power and learning performance of HGNNs?***

In this work, we present a theoretical and practical investigation into this question. We begin by providing rigorous theoretical analysis to demonstrate that incorporating both low-pass and high-pass components yields more expressive and robust hypergraph neural networks than relying on either component alone. Notably, our results reveal that high-pass information is particularly essential in capturing fine-grained structural variations and node-level distinctions that are often diluted in standard HGNN designs. Motivated by these findings, we propose a novel framework, i.e., sheaflet-based HNNs, that unifies cellular sheaf theory and framelet transforms to explicitly model both low- and high-frequency components on hypergraphs. The sheaf structure allows us to preserve the directional and functional dependencies in higher-order relations, while framelets provide a principled tool to decompose and process signals at multiple frequency bands, with a particular emphasis on high-pass signals as guided by our theoretical insights.

In summary, our primary contributions are three-fold:

- **Theoretical Perspective**: We establish a theoretical foundation that characterizes the complementary roles of low- and high-pass components in hypergraph learning, and formally prove that models leveraging both exhibit improved representational capacity.
- **Model Design**: We propose a Sheaflet-based HGNN framework that integrates the expressive advantages of cellular sheaves and the multi-resolution analysis of framelets, explicitly highlighting and utilizing high-frequency signals.
- **Experimental Study**: We conduct extensive empirical evaluations across multiple hypergraph benchmarks to validate our theoretical claims and demonstrate the consistent effectiveness of the proposed method.

2 Related Work

Recent advances in hypergraph learning have yielded various architectures that extend message passing or spectral methods to capture higher-order interactions, including HGNN (Feng et al. 2019), HyperGCN (Yadati et al. 2019), and more recent designs such as HyperND (Prokopchik, Benson, and Tudisco 2022), ED-HNN (Wang et al. 2023),

and HDS^{ode}(Yan et al. 2024). While some methods explore structural transformations or dynamic systems, the role of frequency components in terms of the spectral perspective remains underexplored. FrameHGNN (Li et al. 2025a) is among the few that incorporate both low- and high-pass signals to address oversmoothing in HNNs. Meanwhile, cellular sheaf theory has been employed to introduce geometric structure into graphs and hypergraphs, such as SheafHyperGNN (Duta et al. 2024), though most focus on single-frequency propagation. Although some works have explored framelet-based graph neural networks (Zheng et al. 2021; Luo, Mo, and Pan 2024; Li et al. 2024), they do not provide theoretical insights into how low-pass and high-pass components affect model expressivity. Furthermore, the integration of framelets into sheaf-based hypergraph models remains unexplored. While Chen et al. (2023) propose a combination of sheaf and framelet ideas, their work is limited to graphs and does not address hypergraph structures. These gaps motivate our study, in which we provide theoretical insights into the spectral behavior of hypergraph learning, highlighting the significance of high-pass components, and introduces a sheaflet-based model that unifies cellular sheaves and framelet transforms for multi-frequency hypergraph representation learning.

3 Theoretical Insights and Findings

In this section, we analyze the impact of high-frequency information on the generalization error of HGNNs for node classification. Our results show that combining low-pass and high-pass components improves the expressiveness of the model and reduces the generalization error. To begin, we revisit the node classification problem with n-labels on hypergraph and analyze its generalization error under a probabilistic framework.

Generalization Analysis of Hypergraph Node Classification. Let $\triangle_n = \{x \in \mathbb{R}^n : x_j \in [0,1] \text{ and } \sum_{j=1}^n x_j = 1\}$ and $\bar{\triangle}_n = \{x \in \mathbb{R}^n : x_j \in \{0,1\} \text{ and } \sum_{j=1}^n x_j = 1\}$. Given some observations $(x, y) \in X \times Y$ of nodes/labels, we assume a joint distribution ρ on $X \times Y$. The task is to learn a classification function $f(x) \in \triangle_n$, with minimum expected risk $R(f)$, that can predict the label $y \in \bar{\triangle}_n$ for a given node x. Let $\eta(x) \in \triangle_n$ and the jth component $\eta_j(x) = \Pr(y = e_j \mid x)$ for $(x, y) \sim \rho$. We denote the ground truth by $f_\rho = \eta$. Then the generalization error is given by $R(f) - R(f_\rho)$. Here the expected risk can be defined by 0-1 loss (i.e., $\mathbb{1}(f(x) \neq y) = 1$ if $f(x) \neq y$, and is zero otherwise), and cross-entropy loss (i.e., $\ell(f(x), y) = \sum_{j=1}^n y_j \log f_j(x)$ for $f, y \in \triangle_n$).

Consider the case $n = 2$. Let

$$R(f) = \int_{X\times Y} \mathbb{1}(f(x) \neq y) d\rho$$

and

$$R_\ell(f) = \int_{X\times Y} \ell(f(x), y) d\rho.$$

Theorem 3.1. *Suppose that there exits $s \in (0.5, 1]$ such that $\max\{\eta_1(x), \eta_2(x)\} \geq s$ for all $x \in X$, then*

$$R(f) - R(f_\rho) \leq \frac{1}{s - 0.5}[R_\ell(f) - R_\ell(f_\rho)].$$

Proof. Let $A_j = \{x \in X \mid \text{Label}(f_\rho(x)) = e_j\}$ and $B_j = \{x \in X \mid \text{Label}(f(x)) = e_j\}$.

$$\begin{aligned}
&R(f) - R(f_\rho) \\
=&\sum_{i=1,2} \int_X \eta_i(x)[\mathbb{1}(f(x) \neq e_i) - \mathbb{1}(f_\rho(x) \neq e_i)]dx \\
=&\sum_{i\neq j} \int_X \mathbb{1}(x \in A_i \cap B_j)[\eta_i(x) - \eta_j(x)]dx.
\end{aligned}$$

Note that for any $x \in A_1 \cap B_2$, $\eta_1(x) > \eta_2(x)$ but $f_1(x) \leq f_2(x)$, which yields

$$-\eta_1(x) \log f_1(x) - \eta_2(x) \log f_2(x) \geq \log 2.$$

Furthermore, we have that

$$\begin{aligned}
&R_\ell(f) - R_\ell(f_\rho) \\
=&\int_X -\eta_1(x) \log f_1(x) - \eta_2(x) \log f_2(x) \\
&+ \eta_1(x) \log \eta_1(x) + \eta_2(x) \log \eta_2(x) dx \\
\geq&\int_X [\mathbb{1}(x \in A_1 \cap B_2) + \mathbb{1}(x \in A_2 \cap B_1)] \cdot \\
&[\log 2 + \eta_1(x) \log \eta_1(x) + \eta_2(x) \log \eta_2(x)]dx.
\end{aligned}$$

Notice that for any $t \in (0, 1)$,

$$\log 2 + t \log t + (1 - t) \log(1 - t) \geq (2t - 1)^2/2,$$

which can be verified by checking $h'(t) \geq 0$ and $h''(t) \geq 0$ with $h(t) = \log 2 + t \log t + (1-t) \log(1-t) - (2t-1)^2/2$ for $t \geq 0.5$. Then, by taking $t = \eta_1(x)$ it implies that

$$\begin{aligned}
&2\left[R_\ell(f) - R_\ell(f_\rho)\right] \\
\geq&\sum_{i\neq j} \int_X [\mathbb{1}(x \in A_i \cap B_j)][\eta_i(x) - \eta_j(x)]^2 dx \\
\geq&(2s - 1) \sum_{i\neq j} \int_X [\mathbb{1}(x \in A_i \cap B_j)]|\eta_i(x) - \eta_j(x)|dx,
\end{aligned}$$

where the last inequality is due to the condition $\max\{\eta_1(x), \eta_2(x)\} \geq s > 0.5$. Therefore, rearranging the above inequality we have

$$R(f) - R(f_\rho) \leq \frac{1}{s - 0.5}\left[R_\ell(f) - R_\ell(f_\rho)\right]. \qquad \square$$

Corollary 1. *Given an encoder f, let $\eta(x; f) = \eta(f(x))$ and the jth component $\eta_j(x; f) = \Pr(y = e_j \mid f(x))$, for $(f(x), y) \sim \tilde{\rho}$. Then for a decoder g trained with cross entropy loss,*

$$R(g(f)) - R(f_{\tilde{\rho}}) \leq \frac{1}{s - 0.5}[R_\ell(g(f)) - R_\ell(f_{\tilde{\rho}})],$$

where $f_{\tilde{\rho}} = \eta(x; f)$.

Remark 1. *The proof of the aforementioned corollary can be accomplished by substituting $\eta(x)$ with $\eta(x; f)$.*

Remark 2. *The component $\eta_j(x; f)$ is essentially a conditional probability and should continuous with respect to f. Given two node sets $\{x_m\}$ and $\{x_n\}$, if we have*

$$\eta_j(x_m; f_1) > \eta_j(x_m; f_2) > 0.5$$

and

$$\eta_j(x_n; f_1) < \eta_j(x_n; f_2) < 0.5,$$

then f_1 will exhibit relatively more oscillation than f_2. Consequently, it is logical to expect that $w_r(f_1, t) > w_r(f_2, t)$ for $t > 0$, since the modulus w_r serves as a means of quantifying oscillation.

In the following theorem, we will demonstrate that a highpass filter has the capacity to augment the oscillation of feature expressions.

Theorem 3.2. *If H is a highpass filter with all nonzero eigenvalues having lower bound $\beta > 1$, then*

$$\omega_r(Hf, t) \geq \beta \omega_r(f, t).$$

Proof. Let $H = U \operatorname{diag}(h_1, h_2, \ldots, h_N) U^*$. When H is a highpass filter with $\beta > 1$, we have $0 = h_1 < \beta < h_2 \leq \cdots \leq h_N$. By definition of $\omega_r(f, t)$, for any $s \in \mathbb{R}$,

$$\begin{aligned}
\|(T_s - I)^r Hf\|_2^2 &= \sum_{j=2}^{N} |e^{is\sqrt{\lambda_j}} - 1|^{2r} |h_j \hat{f}(j)|^2 \\
&\geq \beta^2 \sum_{j=2}^{N} |e^{is\sqrt{\lambda_j}} - 1|^{2r} |\hat{f}(j)|^2 \\
&= \beta^2 \|(T_s - I)^r f\|_2^2.
\end{aligned}$$

This proves $\|(T_s - I)^r Hf\|_2 \geq \beta \|(T_s - I)^r f\|_2$. Due to the arbitrary of s,

$$\omega_r(Hf, t) \geq \beta \omega_r(f, t)\,. \qquad \square$$

Our result shows that representations with larger values of s lead to tighter generalization bounds. This insight motivates a principled approach to design representations that explicitly maximize s, which we address next through framelet analysis on hypergraph.

Framelet-Based Representation Learning for Optimal Generalization. Consider a hypergraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with N nodes and hypergraph Laplacian $\mathcal{L}$. Let $\mathbf{U} = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_N]$ denote the matrix of eigenvectors of $\mathcal{L}$, and $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_N)$ be the diagonal matrix of the eigenvalues. We define the Fourier transform for a signal $x \in \mathbb{R}^N$ on hypergraph as $\widehat{x} = \mathbf{U}^\top x$, and the inverse Fourier transform as $x = \mathbf{U}\widehat{x}$. Given a set of filters $\{a_r : 0 \leq r \leq R\}$, the discrete J-level tight wavelet frame decomposition of x is defined as $\{\mathcal{W}_{r,j} x : (r, j) \in \Gamma\}$ with $\Gamma = \{(1,1), (2,1), \ldots, (R,1), (1,2), \ldots, (R,J)\} \cup \{(0, J)\}$ and

$$\begin{aligned}
\mathcal{W}_{0,J} &= \mathbf{U}\widehat{a}_0^*\left(2^{-S+J-1}\Lambda\right) \cdots \widehat{a}_0^*\left(2^{-S}\Lambda\right) \mathbf{U}^\top, \\
\mathcal{W}_{r,1} &= \mathbf{U}\widehat{a}_r^*\left(2^{-S}\Lambda\right) \mathbf{U}^\top, \\
\mathcal{W}_{r,j} &= \mathbf{U}\widehat{a}_r^*\left(2^{-S+j-1}\Lambda\right) \widehat{a}_0^*\left(2^{-S+j-2}\Lambda\right) \cdots \widehat{a}_0^*\left(2^{-S}\Lambda\right) \mathbf{U}^\top
\end{aligned}$$

where h^* denotes the complex conjugate of h. Here, S is chosen to be sufficiently large such that the largest eigenvalue $\lambda_{\max}$ of the hypergraph Laplacian satisfies $\lambda_{\max} \leq 2^S \pi$. The band of the transform is indicated by index r, where $r = 0$ corresponds to the low frequency component, while $1 \leq r \leq R$ represent the high-frequency components. The index j denotes the level of the transform. The tightness of the framelet system can be guaranteed by the condition, $\sum_{r=1}^{R} |\widehat{a}_r(\xi)|^2 = 1$. This ensures that framelet decomposition and reconstruction are invertible, i.e., $\mathcal{W}_{0,J}^\top \mathcal{W}_{0,J} x + \sum_{r,j} \mathcal{W}_{r,j}^\top \mathcal{W}_{r,j} x = x$. Next we present a Gaussian denoising model that incorporates a framelet-based sparse prior.

Theorem 3.3. *Consider the additive noise model $x = z + \sigma n$, with $\sigma > 0$ and $n \sim \mathcal{N}(0, I)$. Let $g(u; \gamma) = \sum_i \gamma_i |u_i|$ denote the weighted ℓ^1-norm of u with non-negative parameter γ. We impose a sparsity enforcing prior on z with the tight framelet transform $\{\mathcal{W}_{r,j} : (r, j) \in \Gamma\}$, i.e., $p(z) \propto \exp[-\sum_{r,j} g(\mathcal{W}_{r,j} z; \gamma_{r,j})]$. Then the MAP estimate is given by $z^* = \sum_{r,j} \mathcal{W}_{r,j}^\top \Theta_{r,j} \mathcal{W}_{r,j} x$, where $\Theta_{r,j}$ are shrinkage-thresholding matrices depending on $\sigma, \gamma_{r,j}$ and $\mathcal{W}_{r,j}$.*

Proof. The MAP estimate maximizes the posterior $p(z|x) \propto p(x|z)p(z)$. Then

$$z^* = \operatorname*{argmax}_z \left[\log p(x|z) + \log p(z)\right].$$

Substituting the likelihood and prior gives

$$z^* = \operatorname*{argmin}_z \left[\frac{1}{2\sigma^2}\|x - z\|_2^2 + \sum_{r,j} g(\mathcal{W}_{r,j} z; \gamma_{r,j})\right].$$

By optimality condition, we have

$$\frac{1}{\sigma^2}(z^* - x) + \sum_{r,j} \partial g(\mathcal{W}_{r,j} z^*; \gamma_{r,j}) \ni 0.$$

Note that the subdifferential of the weighted ℓ^1-norm is explicitly given by

$$\partial g(\mathcal{W}_{r,j} z^*; \gamma_{r,j}) = \mathcal{W}_{r,j}^\top \left.\partial g(u; \gamma_{r,j})\right|_{u = \mathcal{W}_{r,j} z^*},$$

where

$$\partial g(u; \gamma_{r,j}) = \left\{\gamma_{r,j} \odot s : \|s\|_\infty \leq 1, s^\top u = \|u\|_1\right\}.$$

Due to the tight framelet condition, we can decouple the above problem in the transform domain. For each $(r, j) \in \Gamma$ we impose that

$$\frac{1}{\sigma^2}\mathcal{W}_{r,j}(z^* - x) + \left.\partial g(u; \gamma_{r,j})\right|_{u = \mathcal{W}_{r,j} z^*} \ni 0.$$

The above inclusion is equivalent to the proximal operator of $g(\cdot; \sigma^2 \gamma_{r,j})$, i.e.,

$$\mathcal{W}_{r,j} z^* = \operatorname{prox}_g(\mathcal{W}_{r,j} x),$$

such that

$$(\mathcal{W}_{r,j} z^*)_i = \begin{cases} (\mathcal{W}_{r,j} x)_i - \sigma^2 (\gamma_{r,j})_i & \text{if } (\mathcal{W}_{r,j} x)_i > \sigma^2 (\gamma_{r,j})_i, \\ (\mathcal{W}_{r,j} x)_i + \sigma^2 (\gamma_{r,j})_i & \text{if } (\mathcal{W}_{r,j} x)_i < -\sigma^2 (\gamma_{r,j})_i, \\ 0 & \text{otherwise.} \end{cases}$$

That is

$$\mathcal{W}_{r,j} z^* = \Theta_{r,j} \mathcal{W}_{r,j} x,$$

where $\Theta_{r,j} = \mathrm{diag}(\theta_{r,j})$ and each element of $\theta_{r,j}$ is defined as

$$(\theta_{r,j})_i = \begin{cases} 1 - \frac{\sigma^2(\gamma_{r,j})_i}{|(\mathcal{W}_{r,j}x)_i|} & \text{if } |(\mathcal{W}_{r,j}x)_i| > \sigma^2(\gamma_{r,j})_i, \\ 0 & \text{otherwise.} \end{cases}$$

Again, we apply the tight framelet condition to derive z^*

$$z^* = \sum_{r,j} \mathcal{W}_{r,j}^\top \Theta_{r,j} \mathcal{W}_{r,j} x. \qquad \square$$

Remark 3. *Suppose the regularization parameters satisfy that $\gamma_{r,j} \to +\infty$ elementwise for all $(r,j) \in \Gamma$ except the index of the low-pass filter $(0,J)$, while $\lambda_{0,J}$ remains fixed. Then the MAP estimate z^* converges to the low-pass only form, $z_L = \mathcal{W}_{0,J}^\top \Theta_{0,J} \mathcal{W}_{0,J} x$. We can also derive a similar result for the high-pass only estimate, $z_H = \sum_{(r,j)\neq(0,J)} \mathcal{W}_{r,j}^\top \Theta_{r,j} \mathcal{W}_{r,j} x$.*

Remark 4. *The MAP estimate z^* maximizes the oscillation measure s under mild conditions. Let $s(z) = \max_j p(y = e_j, z)$ and k^* be the corresponding maximizer. Assume that $\Pr(y = e_j|x) = 1$ if $j = k^*$ and is zero otherwise. Then $s(z) = \Pr(y = e_{k^*}|x)p(x|z) \propto p(x,z)$, aligning the maximization of $s(z)$ with the joint likelihood. Further we can apply Theorem 3.3 to evaluate, in terms of s, the representations learned via low-pass and high-pass framelets respectively.*

Theorem 3.4. *In the setting of Theorem 3.3, let $s(z) = \log p(x,z)$ be the log-likelihood function, and define the low-pass and high-pass estimates:*

$$z_L = \mathcal{W}_{0,J}^\top \Theta_{0,J} \mathcal{W}_{0,J} x,$$
$$z_H = \sum_{(r,j)\neq(0,J)} \mathcal{W}_{r,j}^\top \Theta_{r,j} \mathcal{W}_{r,j} x.$$

Let $\lambda_{\min}$ denote the smallest non-zero eigenvalue of the hypergraph Laplacian. Suppose the low-pass filter a_0 satisfies $\widehat{a}_0(0) = 1$, $a = \prod_{j=0}^{J-1} |\widehat{a}_0\left(2^{-S+j}\lambda_{\min}\right)|^2 < \frac{1}{2}$. If the high-frequency components of x dominate in the sense that:

$$(1-2a)\left[\sum_{\lambda_k \geq \lambda_{\min}} |\widehat{x}_k|^2\right]^{\frac{1}{2}} \geq \left[\sum_{\lambda_k < \lambda_{\min}} |\widehat{x}_k|^2\right]^{\frac{1}{2}} + \sum_{(r,j)\in\Gamma} \|I - \Theta_{r,j}\|_2 \|\widehat{x}\|_2 + \sqrt{2}\sigma\varepsilon,$$

where

$$\varepsilon = \max\left\{\sum_{(r,j)\in\Gamma} \left[g(\mathcal{W}_{r,j} z_H; \gamma_{r,j}) - g(\mathcal{W}_{r,j} z_L; \gamma_{r,j})\right], 0\right\}^{\frac{1}{2}}.$$

Then we have that $s(z_L) \leq s(z_H)$.

The proof of Theorem 3.4 is provided in the **Appendix**. Our analysis shows that the MAP estimate z^* under this model not only admits a closed-form framelet convolution but also favors high-frequency components. These results provide a theoretical foundation for the design of our proposed architecture, which leverages both low-frequency and high-frequency framelet coefficients to improve generalization, as detailed in the next section.

4 HyperSheaflets

In this section, we present the framework of designing sheaflet-based hypergraph neural networks, termed **HyperSheaflets**, which integrates both low-pass and high-pass filtering by combining cellular sheaf theory with framelet transforms on hypergraphs. To lay the foundation for our model design, we first revisit the definition of cellular sheaves on hypergraphs and the associated linear sheaf hypergraph Laplacian Duta et al. (2024).

Basics of Sheaves on Hypergraphs. A *cellular sheaf $\mathcal{F}$ associated with a hypergraph $\mathcal{H}$* is defined as a triple $\langle \mathcal{F}(v), \mathcal{F}(e), \mathcal{F}_{v \trianglelefteq e} \rangle$, where: i) $\mathcal{F}(v)$ are *vertex stalks:* vector spaces associated with each node v; ii) $\mathcal{F}(e)$ are *hyperedge stalks:* vector spaces associated with each hyperedge e; iii) $\mathcal{F}_{v \trianglelefteq e} : \mathcal{F}(v) \to \mathcal{F}(e)$ are *restriction maps:* linear maps between each pair $v \trianglelefteq e$, if hyperedge e contains node v.

Then, the linear sheaf hypergraph Laplacian is defined as:

$$(\mathcal{L}_\mathcal{F})_{vv} = \sum_{e; v\in e} \frac{1}{\delta_e} \mathcal{F}_{v\trianglelefteq e}^T \mathcal{F}_{v\trianglelefteq e} \in \mathbb{R}^{d\times d}$$

and

$$(\mathcal{L}_\mathcal{F})_{uv} = -\sum_{e; u,v\in e} \frac{1}{\delta_e} \mathcal{F}_{u\trianglelefteq e}^T \mathcal{F}_{v\trianglelefteq e} \in \mathbb{R}^{d\times d},$$

where d is the dimension of the sheaf, $\mathcal{F}_{v\trianglelefteq e} : \mathbb{R}^d \to \mathbb{R}^d$ represents the linear restriction maps guiding the flow of information from node v to hyperedge e.

In particular, the linear sheaf Laplacian operator for node v applied on a signal $x \in \mathbb{R}^{N\times d}$ can be rewritten as:

$$\mathcal{L}_\mathcal{F}(x)_v = \sum_{e; v\in e} \frac{1}{\delta_e} \mathcal{F}_{v\trianglelefteq e}^T \Big(\sum_{\substack{u\in e \\ u\neq v}} (\mathcal{F}_{v\trianglelefteq e} x_v - \mathcal{F}_{u\trianglelefteq e} x_u)\Big). \quad (1)$$

Construction of Sheaflets on Hypergraphs. Following the general principles of constructing framelet systems on graphs (Chen et al. 2023), we extend this methodology to define sheaflets on hypergraphs. Let $\{(u_l, \lambda_l)\}_{l=1}^{Nd}$ denote the eigenpairs of the linear sheaf hypergraph Laplacian $L_\mathcal{F}$. For $j \in \mathbb{Z}$ and $p \in V$, we define the undecimated sheaflets $\phi_{j,p}(v)$ and $\psi_{j,p}^r(v), v \in V$ at scale j as follows:

$$\begin{aligned} \phi_{j,p}(v) &:= \sum_{l=1}^{Nd} \widehat{\alpha}\left(\frac{\lambda_l}{2^j}\right) \overline{u_l(p)} u_l(v), \\ \psi_{j,p}^r(v) &:= \sum_{l=1}^{Nd} \widehat{\beta}\left(\frac{\lambda_l}{2^j}\right) \overline{u_l(p)} u_l(v), \quad r = 1, \ldots, n. \end{aligned} \quad (2)$$

Here, the scaling functions $\{\alpha; \beta^{(1)}, \ldots, \beta^{(n)}\}$, are associated with a filter bank $\eta = \{a; b^{(1)}, \ldots, b^{(n)}\}$, satisfying

$$\widehat{\alpha}(2\xi) = \widehat{a}(\xi)\widehat{\alpha}(\xi), \quad \widehat{\beta}^{(r)}(2\xi) = \widehat{b}^{(r)}(\xi)\widehat{\alpha}(\xi), \quad \forall \xi \in \mathbb{R},$$

where $\widehat{h}(\xi)$ denotes the Fourier transform of a function h, defined by $\widehat{h}(\xi) := \sum_{k\in\mathbb{Z}} h(k)\, e^{-2\pi i k\xi}$. Here, α corresponds to the low-pass scaling function, while $\{\beta^{(r)}\}_{r=1}^n$ represent the high-pass functions, and n is the number of high-pass channels in the filter bank.

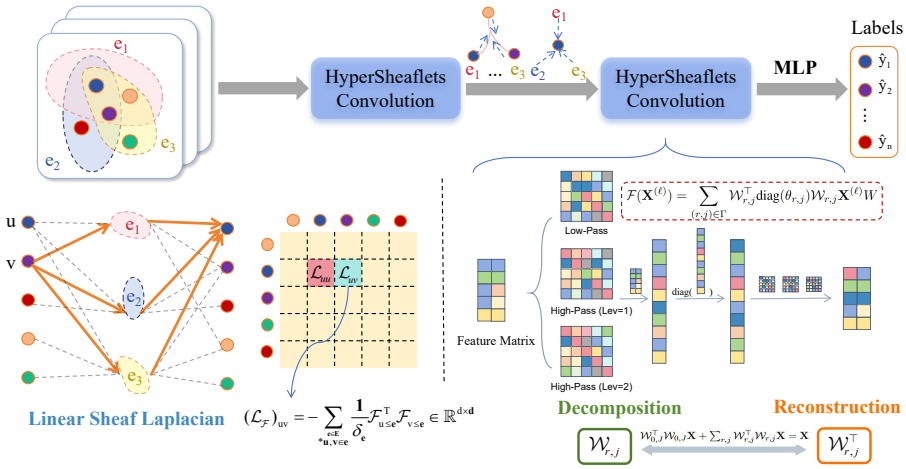

Figure 1: An overview of HyperSheaflets.

Sheaflet coefficients $V_0, W_j^r \in \mathbb{R}^{Nd\times m}$ are defined as:

$$V_0 = \langle \phi_{0,\cdot}, X\rangle = \mathbf{U}^\top \widehat{\alpha}\left(\frac{\Lambda}{2}\right)\mathbf{U}X,$$
$$W_j^r = \left\langle \psi_{j,\cdot}^r, X\right\rangle = \mathbf{U}^\top \widehat{\beta^{(r)}}\left(\frac{\Lambda}{2j+1}\right)\mathbf{U}X, \tag{3}$$

where $X \in \mathbb{R}^{Nd\times m}$ denotes the sheaf signal, m is the feature dimension.

Let $\mathcal{W}_{r,j}$ denote the decomposition operators given by $V_0 = \mathcal{W}_{0,J}X$ and $W_j^r = \mathcal{W}_{r,j}X$. To avoid the computational burden of directly computing the eigendecomposition of sheaf Laplacian $\mathcal{L}_F$. Given Chebyshev polynomials $\mathcal{T}_0, \cdots, \mathcal{T}_n$ of fixed degree t, we can approximate the filter bank as $a \approx \mathcal{T}_0$ and $b^{(r)} \approx \mathcal{T}_r$, then the decomposition operators $\mathcal{W}_{r,j}$ can be approximated

$$\begin{aligned}\mathcal{W}_{0,J} &\approx \mathbf{U}^\top \mathcal{T}_0(2^{-K+J-1}\Lambda)\cdots\mathcal{T}_0(2^{-K}\Lambda)\mathbf{U}\\ &= \mathcal{T}_0(2^{K+J-2}L_{\mathcal{F}})\cdots\mathcal{T}_0(2^{-K}L_{\mathcal{F}}),\\ \mathcal{W}_{r,1} &\approx \mathbf{U}^\top \mathcal{T}_r(2^{-K}\Lambda)\mathbf{U} = \mathcal{T}_r(2^{-K}L_{\mathcal{F}}),\\ \mathcal{W}_{r,j} &\approx \mathbf{U}^\top \mathcal{T}_r(2^{-K+j-1}\Lambda)\mathcal{T}_0(2^{-K+j-2}\Lambda)\cdots\mathcal{T}_0(2^{-K}\Lambda)\mathbf{U}\\ &= \mathcal{T}_r(2^{K+j-1}L_{\mathcal{F}})\mathcal{T}_0(2^{K+j-2}L_{\mathcal{F}})\cdots\mathcal{T}_0(2^{-K}L_{\mathcal{F}}).\end{aligned}$$

Hypergraph Neural Networks with Sheaflets. Given a hypergraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where each node is associated with a feature representation $\mathbf{X} \in \mathbb{R}^{N\times m}$, we begin by applying a linear projection to map the input features into a higher-dimensional space $\tilde{X} \in \mathbb{R}^{N\times(dm)}$. We then reshape $\tilde{X}$ into $\mathbb{R}^{Nd\times m}$ to obtain a structure compatible with the vertex stalk representation. As a result, each node is embedded as a matrix in $\mathbb{R}^{d\times m}$, where d denotes the dimension of the vertex stalk and m corresponds to the number of feature channels. Based on the constructed sheaflet operators on hypergraphs, i.e., $\mathcal{W}_{0,J}, W_{r,j}$ as defined above, we formulate a hypergraph neural network consisting of two layers of sheaflet-based spectral convolution. Specifically, the forward propagation is defined as:

$$\begin{aligned}\tilde{X}^{(\ell+1)} =& \sigma\Big(\mathcal{W}_{0,J}^\top \Theta_{0,J}\mathcal{W}_{0,J}\tilde{X}^{(\ell)}W_{0,J}\\ &+ \sum_{r,j}\mathcal{W}_{r,j}^\top\Theta_{r,j}\mathcal{W}_{r,j}\tilde{X}^{(\ell)}W_{r,j}\Big),\end{aligned}$$

where $\ell = 0, 1$ denotes the first and second layers, respectively, and $\tilde{X}^{(0)} := \tilde{X}$ is the initial input feature matrix. The diagonal matrices $\Theta_{0,J} = \mathrm{diag}(\theta_{0,J})$, $\Theta_{r,j} = \mathrm{diag}(\theta_{r,j})$ contain learnable spectral filter coefficients for the low- and high-frequency components, respectively. The matrices with $W_{0,J}$, $W_{r,j}$ are trainable transformation weights applied to the corresponding frequency responses. The nonlinearity $\sigma(\cdot)$ denotes an activation function such as ReLU.

Remark 5. The overall architecture of HyperSheaflets is illustrated in Figure 1, where we adopt a two-layer design, consistent with our experimental setup. Technically, the framework can be extended to deeper architectures by incorporating residual connections and identity mappings, following techniques introduced in (Chen et al. 2020). These mechanisms help preserve the initial node features and facilitate stable training by mitigating oversmoothing (as demonstrated in our experiments on deep HNNs equipped with sheaflets). A generalized propagation rule for such deeper variants can be expressed as:

$$\begin{aligned}\tilde{X}^{(\ell+1)} =& \sigma\Big(\Big((1-\alpha_\ell)\big(\mathcal{W}_{0,J}^\top\Theta_{0,J}\mathcal{W}_{0,J}\tilde{X}^{(\ell)} + \sum_{r,j}\mathcal{W}_{r,j}^\top\Theta_{r,j}\\ &\cdot\mathcal{W}_{r,j}\tilde{X}^{(\ell)}\big) + \alpha_\ell\tilde{X}\Big)\Big((1-\beta_\ell)\mathbf{I} + \beta_\ell\mathbf{\Theta}^{(\ell)}\Big)\Big),\end{aligned}$$

where α_ℓ, β_ℓ are two hyperparameters, $\mathbf{\Theta}^{(\ell)}$ is the trainable parameter matrix.

5 Experiments

5.1 Experimental Setups

Datasets. We evaluate HyperSheaflets on 12 benchmark datasets, including Cora, Citeseer, Pubmed, Cora-CA, DBLP-CA (Yadati et al. 2019), House (Chodrow, Veldt, and Benson 2021), Senate, and Congress (Fowler 2006), as well as four recently introduced heterophilic hypergraph datasets: Actor, Twitch, Amazon, and Pokec (Li et al. 2025b). For the first eight datasets, we adopt a 50%/25%/25% split for training, validation, and testing, while the heterophilic datasets

Table 1: Accuracy (%) comparison across 12 datasets, including six *homophilic* and six *heterophilic* ones. Results are reported as mean and standard deviation over 10 runs. Best results are in **bold**; second-best are underlined. 'OOM' denotes out-of-memory.

Datasets	Cora	Citeseer	Pubmed	Cora-CA	DBLP-CA	Congress
HGNN	79.39 ± 1.36	72.45 ± 1.16	86.44 ± 0.44	82.64 ± 1.65	91.03 ± 0.20	91.26 ± 1.15
HyperGCN	78.45 ± 1.26	71.28 ± 0.82	82.84 ± 8.67	79.48 ± 2.08	89.38 ± 0.25	55.12 ± 1.96
UniGCNII	78.81 ± 1.05	73.05 ± 2.21	88.25 ± 0.40	83.60 ± 1.14	91.69 ± 0.19	94.81 ± 0.81
HyperND	79.20 ± 1.14	72.62 ± 1.49	86.68 ± 1.32	80.62 ± 1.32	90.35 ± 0.26	74.63 ± 3.62
AllDeepSets	76.88 ± 1.80	70.83 ± 1.63	88.75 ± 0.33	81.97 ± 1.50	91.27 ± 0.27	91.80 ± 1.53
AllSetTransformer	78.58 ± 1.47	73.08 ± 1.20	88.72 ± 0.37	83.63 ± 1.47	91.53 ± 0.23	92.16 ± 1.05
ED-HNN	80.31 ± 1.35	73.70 ± 1.38	**89.03 ± 0.53**	83.97 ± 1.55	**91.90 ± 0.19**	**95.00 ± 0.99**
SheafHyperGNN	81.30 ± 1.70	74.71 ± 1.23	87.68 ± 0.60	85.52 ± 1.28	91.59 ± 0.24	91.81 ± 1.60
HyperUFG	81.51 ± 0.99	74.72 ± 2.10	88.73 ± 0.42	85.18 ± 0.69	91.67 ± 0.31	OOM
HyperSheaflets(Ours)	**81.60 ± 1.92**	**75.19 ± 1.80**	87.19 ± 0.45	**85.85 ± 0.92**	91.58 ± 0.27	92.07 ± 1.22

Datasets	Senate	House	Actor	Amazon	Twitch	Pokec	Rank(↑)
HGNN	48.59 ± 4.52	61.39 ± 2.96	74.47 ± 0.32	23.79 ± 0.24	51.88 ± 0.26	49.82 ± 0.27	8
HyperGCN	42.45 ± 3.67	48.32 ± 2.93	68.67 ± 4.38	22.53 ± 3.94	51.32 ± 1.02	52.43 ± 3.68	10
UniGCNII	49.30 ± 4.25	67.25 ± 2.57	80.48 ± 1.13	26.63 ± 1.32	50.84 ± 0.76	54.25 ± 2.70	5
HyperND	52.82 ± 3.20	51.70 ± 3.37	**92.52 ± 0.81**	26.08 ± 0.33	51.44 ± 0.67	55.94 ± 0.45	7
AllDeepSets	48.17 ± 5.67	67.82 ± 2.40	82.00 ± 2.33	18.60 ± 0.17	50.72 ± 0.96	51.11 ± 1.04	9
AllSetTransformer	51.83 ± 5.22	69.33 ± 2.20	83.39 ± 1.73	18.60 ± 0.17	50.45 ± 0.76	58.40 ± 0.42	6
ED-HNN	64.79 ± 5.14	72.45 ± 2.28	91.86 ± 0.43	26.21 ± 0.36	50.86 ± 0.88	59.11 ± 0.57	3rd
SheafHyperGNN	68.73 ± 4.68	73.84 ± 2.30	80.09 ± 2.45	26.93 ± 3.04	51.03 ± 0.76	55.34 ± 4.39	4
HyperUFG	67.61 ± 7.00	72.82 ± 2.22	89.32 ± 0.75	**40.53 ± 2.25**	**52.35 ± 0.04**	**62.30 ± 0.12**	2nd
HyperSheaflets(Ours)	**69.01 ± 5.39**	**74.49 ± 1.21**	84.77 ± 0.53	27.13 ± 0.48	52.29 ± 0.59	59.81 ± 0.55	1st

follow the 40%/20%/40% protocol from (Li et al. 2025b) to ensure fair comparison. Further details on the dataset statistics are summarized in **Appendix**, where we also report node and hyperedge homophily levels, denoted as $\mathcal{H}_{\text{node}}$ and $\mathcal{H}_{\text{edge}}$, respectively. Based on homophily ratios, the datasets can be broadly categorized into eight homophilic and six heterophilic datasets. All models are trained for up to 1,000 epochs with early stopping (patience = 200), and results are averaged over 10 random splits to report mean accuracy and standard deviation.

Baselines. We compare HyperSheaflets against 9 existing models, including HGNN (Feng et al. 2019), HyperGCN (Yadati et al. 2019), UniGCNII (Huang and Yang 2021), HyperND (Prokopchik, Benson, and Tudisco 2022), AllDeepSets (Chien et al. 2022), AllSetTransformer (Chien et al. 2022), ED-HNN (Wang et al. 2023), SheafHyperGNN (Duta et al. 2024), HyperUFG (Li et al. 2025b) .

5.2 Overall Performance Comparison

Table 1 summarizes the performance of HyperSheaflets on node classification tasks across eight widely used benchmarks and four recently introduced heterophilic datasets. The results demonstrate that our model consistently performs well across all datasets and achieves state-of-the-art performance on the majority of them. Notably, HyperSheaflets shows clear advantages on challenging datasets such as Senate, House, Twitch, and Pokec, highlighting its strong capacity to model complex higher-order relationships. These results underscore the model's robustness and its effectiveness in handling both homophilic and heterophilic hypergraph structures.

Potential for Preventing Oversmoothing. We conduct a set of experiments to examine whether HyperSheaflets can maintain stable performance as the network depth increases, i.e., a key indicator of resistance to oversmoothing, which is a well-known limitation in deep GNNs and HNNs. As shown in Figure 2, HyperSheaflets exhibits stable accuracy across a wide range of layer depths (from 1 to 32) on Citeseer and House datasets, which are representative of homophilic and heterophilic hypergraphs, respectively. These results suggest that the proposed model is less prone to oversmoothing, likely due to its multi-frequency design and sheaf-based formulation. While this issue is not the primary focus of our work, the findings highlight the model's potential for enabling deeper architectures without substantial performance degradation.

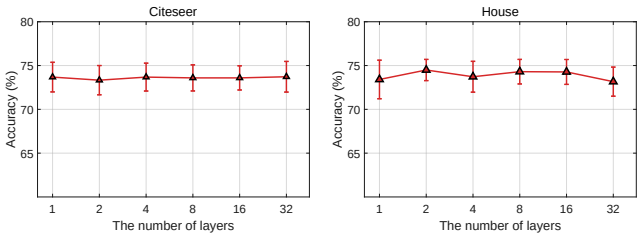

Figure 2: Demonstration of how HyperSheaflets alleviate oversmoothing.

5.3 Parameter Sensitivity Analysis

The scale level in HyperSheaflets controls the number of hierarchical resolutions used for multi-scale spectral decomposition, ranging from the coarsest scale (capturing global structures) to the finest scale (capturing localized variations). We examine its influence by varying the scale level from 1 to 6 on the Citeseer and House datasets. As shown in Figure 3, the model exhibits stable performance across differ-

Table 2: Ablation study on the contributions of low-pass and high-pass components.

Datasets	Cora	Citeseer	Pubmed	Cora-CA	DBLP-CA	Congress
Full model	81.60 ± 1.92	75.19 ± 1.80	87.19 ± 0.45	85.85 ± 0.92	91.58 ± 0.27	92.07 ± 1.22
w/o low pass	81.08 ± 1.68	74.23 ± 1.59	86.66 ± 0.53	85.35 ± 1.03	91.42 ± 0.21	91.70 ± 1.54
w/o high pass	30.77 ± 1.85	51.10 ± 1.47	40.19 ± 2.02	22.98 ± 1.93	26.70 ± 0.54	51.67 ± 1.77
Datasets	**Senate**	**House**	**Actor**	**Amazon**	**Twitch**	**Pokec**
Full model	69.01 ± 5.39	74.49 ± 1.21	84.77 ± 0.53	27.13 ± 0.48	52.29 ± 0.59	59.81 ± 0.55
w/o low pass	65.63 ± 5.48	73.68 ± 2.12	84.53 ± 0.45	26.99 ± 0.22	51.93 ± 0.55	59.62 ± 0.45
w/o high pass	64.51 ± 7.51	54.58 ± 3.63	62.41 ± 0.81	26.39 ± 0.70	50.79 ± 0.84	50.41 ± 0.75

ent settings, with slightly better accuracy achieved at lower levels (e.g., 1 or 2). These results suggest that a small number of scales is sufficient for capturing meaningful multi-frequency representations, while higher scale levels may introduce redundancy and impose additional computational overhead without significant performance improvement.

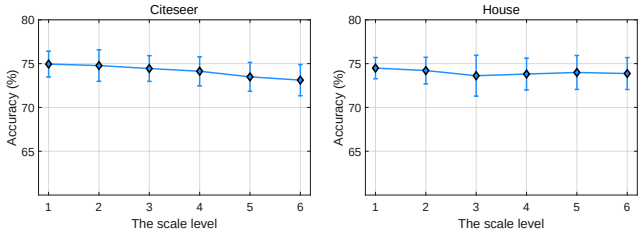

Figure 3: Impact of scale level on the overall performance.

5.4 Ablation Study

To validate the necessity of incorporating both low-pass and high-pass components in HyperSheaflets, we conduct an ablation study by selectively removing each frequency component and evaluating the resulting performance. Specifically, we consider three variants: the full model (with both low- and high-pass), a variant without low-pass components, and a variant without high-pass components.

As shown in Table 2, removing the high-pass component consistently leads to significant performance degradation across all datasets. For instance, on the Cora, Pubmed, and Actor datasets, the accuracy drops from 81.60% to 30.77%, 87.19% to 40.19%, and 84.77% to 62.41%, respectively. This sharp decline highlights the essential role of high-pass signals in capturing local variations and preserving node-level discriminability, which are especially critical in non-homophilic or structurally complex settings. In contrast, removing the low-pass component results in only marginal decreases in performance on most datasets. The slight performance drop indicats that while low-pass signals contribute to smoothing and global consistency, they are less critical than high-pass signals in our model. The relatively minor impact of removing low-pass filtering further corroborates our theoretical finding that high-frequency components play a dominant role in enhancing the expressivity of hypergraph neural networks. Figure 4 further illustrates this observation through a visualization of node embeddings on the Cora dataset. The full model yields well-separated clusters, while the removal of high-pass components leads to severe mixing of class distributions. The model without low-pass filtering still maintains clear boundaries among classes, though the clusters are slightly less compact. Overall, the ablation results validate our spectral design and demonstrate that the high-pass component is indispensable for effective hypergraph learning, which aligns well with our theoretical results and insights.

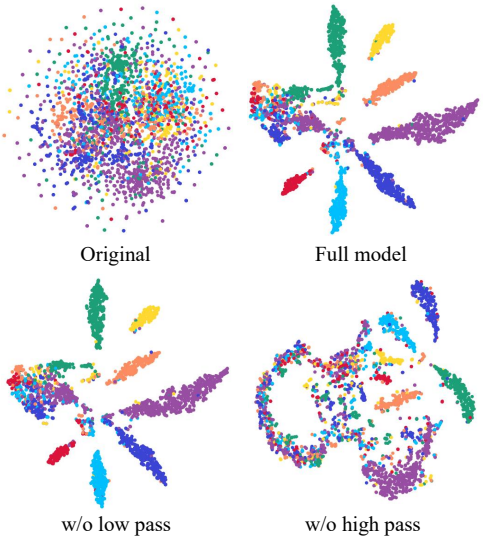

Figure 4: Visualization of node embeddings on Cora for the full HyperSheaflets model and its ablated variants (w/o low-pass, w/o high-pass).

6 Conclusion

This work provides a theoretical and empirical investigation into the role of spectral components in hypergraph neural networks. We prove that combining both low-pass and high-pass signals enhances the expressive power of HGNNs, with high-pass components playing a particularly critical role in capturing fine-grained relational structures. Motivated by these insights, we propose HyperSheaflets, a novel sheaflet-based HNNs that integrates cellular sheaf theory and framelet transforms to perform multi-frequency signal processing on hypergraphs. Our model effectively preserves higher-order relational dependencies while emphasizing high-frequency information. Extensive experiments across benchmark datasets validate the theoretical claims and demonstrate the superior performance of the proposed method. Inspired by our theoretical results and analysis, future work is expected to explore more advanced hypergraph neural networks with well-designed multi-frequency filters in the context of complex real-world applications.

References

Antelmi, A.; Cordasco, G.; Polato, M.; Scarano, V.; Spagnuolo, C.; and Yang, D. 2023. A survey on hypergraph representation learning. *ACM Computing Surveys*, 56(1): 1–38.

Bo, D.; Wang, X.; Shi, C.; and Shen, H. 2021. Beyond low-frequency information in graph convolutional networks. In *AAAI*, 3950–3957.

Chen, J.; Wang, Y.; Bodnar, C.; Ying, R.; Lio, P.; and Wang, Y. G. 2023. Dirichlet energy enhancement of graph neural networks by framelet augmentation. *arXiv preprint arXiv:2311.05767*.

Chen, M.; Wei, Z.; Huang, Z.; Ding, B.; and Li, Y. 2020. Simple and deep graph convolutional networks. In *ICML*, 1725–1735. PMLR.

Chien, E.; Pan, C.; Peng, J.; and Milenkovic, O. 2022. You are AllSet: a multiset function framework for hypergraph neural networks. In *ICLR*.

Chodrow, P. S.; Veldt, N.; and Benson, A. R. 2021. Generative hypergraph clustering: from blockmodels to modularity. *Science Advances*, 7(28): eabh1303.

Duta, I.; Cassarà, G.; Silvestri, F.; and Liò, P. 2024. Sheaf hypergraph networks. *In NeurIPS*, 36: 12087–12099.

Feng, Y.; You, H.; Zhang, Z.; Ji, R.; and Gao, Y. 2019. Hypergraph neural networks. In *AAAI*, 3558–3565.

Fowler, J. H. 2006. Legislative cosponsorship networks in the US House and Senate. *Social Networks*, 28(4): 454–465.

Gao, Y.; Ji, S.; Han, X.; and Dai, Q. 2024. Hypergraph computation. *Engineering*, 40: 188–201.

Huang, J.; and Yang, J. 2021. UniGNN: a unified framework for graph and hypergraph neural networks. In *IJCAI*, 2563–2569.

Kim, S.; Lee, S. Y.; Gao, Y.; Antelmi, A.; Polato, M.; and Shin, K. 2024. A survey on hypergraph neural networks: An in-depth and step-by-step guide. In *KDD*, 6534–6544.

Li, J.; Zheng, R.; Feng, H.; Li, M.; and Zhuang, X. 2024. Permutation equivariant graph framelets for heterophilous graph learning. *IEEE Transactions on Neural Networks and Learning Systems*, 35(9): 11634–11648.

Li, M.; Fang, Y.; Wang, Y.; Feng, H.; Gu, Y.; Bai, L.; and Lio, P. 2025a. Deep hypergraph neural networks with tight framelets. In *AAAI*, 18385–18392.

Li, M.; Gu, Y.; Wang, Y.; Fang, Y.; Bai, L.; Zhuang, X.; and Lio, P. 2025b. When hypergraph meets heterophily: New benchmark datasets and baseline. In *AAAI*, 18377–18384.

Luo, T.; Mo, Z.; and Pan, S. J. 2024. Learning adaptive multiresolution transforms via meta-framelet-based graph convolutional network. In *ICLR*.

Millán, A. P.; Sun, H.; Giambagli, L.; Muolo, R.; Carletti, T.; Torres, J. J.; Radicchi, F.; Kurths, J.; and Bianconi, G. 2025. Topology shapes dynamics of higher-order networks. *Nature Physics*, 21: 353—361.

Prokopchik, K.; Benson, A. R.; and Tudisco, F. 2022. Nonlinear feature diffusion on hypergraphs. In *ICML*, 17945–17958.

Wang, P.; Yang, S.; Liu, Y.; Wang, Z.; and Li, P. 2023. Equivariant hypergraph diffusion neural operators. In *ICLR*.

Wang, Y.; and Kleinberg, J. 2024. From Graphs to Hypergraphs: Hypergraph Projection and its Reconstruction. In *ICLR*.

Yadati, N.; Nimishakavi, M.; Yadav, P.; Nitin, V.; Louis, A.; and Talukdar, P. 2019. HyperGCN: A new method for training graph convolutional networks on hypergraphs. In *NeurIPS*, 1511–1522.

Yan, J.; Feng, Y.; Ying, S.; and Gao, Y. 2024. Hypergraph dynamic system. In *ICLR*.

Zheng, X.; Zhou, B.; Gao, J.; Wang, Y. G.; Lió, P.; Li, M.; and Montúfar, G. 2021. How framelets enhance graph neural networks. In *ICML*, 12761–12771.

Reproducibility Checklist

1. General Paper Structure

1.1. Includes a conceptual outline and/or pseudocode description of AI methods introduced (yes/partial/no/NA) yes

1.2. Clearly delineates statements that are opinions, hypothesis, and speculation from objective facts and results (yes/no) yes

1.3. Provides well-marked pedagogical references for less-familiar readers to gain background necessary to replicate the paper (yes/no) yes

2. Theoretical Contributions

2.1. Does this paper make theoretical contributions? (yes/no) yes

If yes, please address the following points:

2.2. All assumptions and restrictions are stated clearly and formally (yes/partial/no) yes

2.3. All novel claims are stated formally (e.g., in theorem statements) (yes/partial/no) yes

2.4. Proofs of all novel claims are included (yes/partial/no) yes

2.5. Proof sketches or intuitions are given for complex and/or novel results (yes/partial/no) yes

2.6. Appropriate citations to theoretical tools used are given (yes/partial/no) yes

2.7. All theoretical claims are demonstrated empirically to hold (yes/partial/no/NA) yes

2.8. All experimental code used to eliminate or disprove claims is included (yes/no/NA) yes

3. Dataset Usage

3.1. Does this paper rely on one or more datasets? (yes/no) yes

If yes, please address the following points:

3.2. A motivation is given for why the experiments are conducted on the selected datasets (yes/partial/no/NA) yes

3.3. All novel datasets introduced in this paper are included in a data appendix (yes/partial/no/NA) yes

3.4. All novel datasets introduced in this paper will be made publicly available upon publication of the paper with a license that allows free usage for research purposes (yes/partial/no/NA) yes

3.5. All datasets drawn from the existing literature (potentially including authors' own previously published work) are accompanied by appropriate citations (yes/no/NA) yes

3.6. All datasets drawn from the existing literature (potentially including authors' own previously published work) are publicly available (yes/partial/no/NA) yes

3.7. All datasets that are not publicly available are described in detail, with explanation why publicly available alternatives are not scientifically satisficing (yes/partial/no/NA) yes

4. Computational Experiments

4.1. Does this paper include computational experiments? (yes/no) yes

If yes, please address the following points:

4.2. This paper states the number and range of values tried per (hyper-) parameter during development of the paper, along with the criterion used for selecting the final parameter setting (yes/partial/no/NA) yes

4.3. Any code required for pre-processing data is included in the appendix (yes/partial/no) yes

4.4. All source code required for conducting and analyzing the experiments is included in a code appendix (yes/partial/no) yes

4.5. All source code required for conducting and analyzing the experiments will be made publicly available upon publication of the paper with a license that allows free usage for research purposes (yes/partial/no) yes

4.6. All source code implementing new methods have comments detailing the implementation, with references to the paper where each step comes from (yes/partial/no) yes

4.7. If an algorithm depends on randomness, then the method used for setting seeds is described in a way sufficient to allow replication of results (yes/partial/no/NA) yes

4.8. This paper specifies the computing infrastructure used for running experiments (hardware and software), including GPU/CPU models; amount of memory; operating system; names and versions of relevant software libraries and frameworks (yes/partial/no) yes

4.9. This paper formally describes evaluation metrics used and explains the motivation for choosing these metrics (yes/partial/no) yes

4.10. This paper states the number of algorithm runs used to compute each reported result (yes/no) yes

4.11. Analysis of experiments goes beyond single-dimensional summaries of performance (e.g., average; median) to include measures of variation, confidence, or other distributional information (yes/no) yes

4.12. The significance of any improvement or decrease in performance is judged using appropriate statistical tests (e.g., Wilcoxon signed-rank) (yes/partial/no) partial

4.13. This paper lists all final (hyper-)parameters used for each model/algorithm in the paper's experiments (yes/partial/no/NA) partial