FraudGNAM: Inherently Interpretable Spectral GNN for Graph Fraud Detection

Suan Lee[1] **Jinho Kim**[2]

[1]School of Computer Science, Semyung University, Jecheon, Republic of Korea
[2]AI Research Lab, Korea Information Systems Consulting & Audit Co., Ltd., Republic of Korea

Abstract

Graph neural networks for fraud detection face a fundamental tension: spectral methods that capture heterophilic patterns sacrifice interpretability, while interpretable models lack the spectral capacity to detect sophisticated fraud. We introduce FraudGNAM (**Fraud G**raph **N**eural **A**dditive **M**odel), the first *component-level inherently interpretable* spectral GNN with learnable multi-scale filters that resolves this tension. FraudGNAM decomposes the fraud score into individually auditable components at three interpretability tiers: *component-level* and *per-band spectral* (universally available and exact), and *per-feature* via neural additive shape functions (when feature dimensionality $d \leq 40$). Components include per-feature contributions, pairwise feature interactions, multi-scale spectral band responses via learnable polynomial filters, and explicit feature-spectrum interactions, all combined through a transparent scalar-sum architecture. Each component's contribution to the final prediction is independently quantifiable, unlike post-hoc explanation methods that approximate black-box decisions. On seven GADBench datasets spanning homophily ratios from 0.60 to 0.98, a refined theory-guided adaptive rule lets FraudGNAM win AUROC on all seven datasets against SEC-GFD, the strongest black-box baseline, taking 20 of 21 metric-dataset comparisons (paired Wilcoxon, exact binomial $p = 4.6 \times 10^{-5}$). A deterministic, theory-guided adaptive configuration mechanism selects spectral order, high-pass filter count, and contrastive regularization based on measurable graph properties, eliminating per-dataset hyperparameter search.

1 INTRODUCTION

Financial fraud detection in transaction and social networks is a high-stakes application where graph structure can reveal suspicious relational patterns [Wang et al., 2019]. A key challenge is that fraud graphs exhibit *heterophily*: fraudsters strategically connect to normal users to camouflage their behavior, violating the homophily assumption underlying standard GNNs [Dou et al., 2020].

Recent spectral approaches address heterophily by decomposing graph signals across frequency bands. BWGNN [Tang et al., 2022] uses Beta wavelet filters, and SEC-GFD [Xu et al., 2024] combines high-pass filtering with contrastive learning. These methods achieve strong performance but operate as black boxes: practitioners cannot inspect *why* a node is flagged as fraud or *which* spectral patterns and features contribute to the decision.

Post-hoc explanation methods like GNNExplainer [Ying et al., 2019] approximate black-box decisions but suffer from faithfulness issues: explanations may not reflect the model's true reasoning [Luo et al., 2020]. In high-stakes domains like fraud detection, where regulatory compliance often demands audit trails, *inherent* interpretability—where the model architecture itself ensures decomposability—is more appropriate.

Neural Additive Models (NAMs) [Agarwal et al., 2021] achieve inherent interpretability for tabular data by decomposing predictions into per-feature contributions $f(x) = \beta_0 + \sum_i f_i(x_i)$. GA^{2}Ms [Lou et al., 2013] extend this with pairwise interactions. However, neither framework incorporates graph structure, leaving an open problem for interpretable graph learning. In fraud detection, graph topology carries critical information—yet current approaches force a binary choice between spectral power and interpretability.

We introduce FraudGNAM, which combines NAM/GA^{2}M-style decomposition with spectral graph filtering in a single architecture. Our contributions are:

Proceedings of the 42nd Conference on Uncertainty in Artificial Intelligence (UAI 2026), PMLR 337:3398–3428.

1. **Inherently interpretable spectral GNN architecture** that decomposes fraud predictions into independently auditable components: feature contributions, pairwise interactions, spectral band responses, feature-spectrum interactions, and high-pass filter responses, all combined via a transparent scalar-sum (Section 4).
2. **Theory-guided deterministic adaptive spectral configuration**: a homophily-order design guideline (Guideline 1) selects polynomial order, high-pass filter count, and contrastive regularization from four measurable graph statistics, eliminating per-dataset hyperparameter search. The configuration rules pass leave-one-dataset-out validation (Appendix M) and threshold sensitivity sweeps (Appendix L, Section 5.5).
3. **State-of-the-art performance with interpretability**: on seven GADBench datasets [Tang et al., 2023], FraudGNAM outperforms SEC-GFD [Xu et al., 2024] in AUROC on all seven datasets under a refined adaptive rule (Section 5.5, Appendix V), winning 20 of 21 dataset-metric comparisons (paired Wilcoxon, exact binomial $p = 4.6 \times 10^{-5}$) while maintaining full component decomposability.

2 RELATED WORK

Spectral GNNs. GCN [Kipf and Welling, 2017] implements a first-order Chebyshev approximation of spectral filters. ChebNet [Defferrard et al., 2016] generalizes to order-K Chebyshev polynomials but with fixed basis functions. BernNet [He et al., 2021] uses Bernstein polynomials with learnable coefficients, achieving arbitrary filter shapes while maintaining non-negativity. GPR-GNN [Chien et al., 2021] learns generalized PageRank weights for adaptive frequency response. BWGNN [Tang et al., 2022] applies Beta wavelet bases specifically for fraud detection, demonstrating that multi-scale spectral decomposition captures fraudster camouflage patterns. S^2GNN [Geisler et al., 2024] combines spatial and spectral filters for global propagation. None of these methods adapt polynomial order to graph properties, and all lack interpretable prediction decomposition.

Heterophilic GNNs. H_2GCN [Zhu et al., 2020] separates ego- and neighbor-embeddings for heterophilic graphs. FAGCN [Bo et al., 2021] uses signed attention for adaptive frequency selection. SEC-GFD [Xu et al., 2024] combines element-wise spectral convolutions with high-pass filtering and NCE contrastive loss, achieving strong fraud detection performance. GAGA [Wang et al., 2023] converts the graph into label-group-aggregated feature sequences processed by a Transformer encoder. PMP [Zhuo et al., 2024] partitions message passing by source-node pseudo-labels to avoid mixing signals from different classes. Orthogonal data-centric approaches (graph contrastive pretraining, label denoising) address label scarcity; FraudGNAM's training-only NCE provides a lightweight alternative without requiring pretraining. All of the above operate as black boxes: they entangle all components before classification, making individual predictions opaque.

Interpretable models. NAMs [Agarwal et al., 2021] decompose predictions as $f(\mathbf{x}) = \beta_0 + \sum_i f_i(x_i)$, where each f_i is a neural network processing a single feature. GA^2Ms [Lou et al., 2013] add pairwise terms $f_{ij}(x_i, x_j)$ for capturing feature interactions. NODE-GAM [Chang et al., 2022] replaces neural networks with oblivious decision trees for sharper shape functions. GNAN [Bechler-Speicher et al., 2024] extends additive models to graph data by decomposing GNN predictions into per-feature terms with message passing, but uses a fixed aggregation scheme without spectral filtering. These approaches achieve interpretability competitive with black-box models on tabular data but either cannot incorporate graph structure (NAM, GA^2M) or lack spectral capacity for heterophilic fraud patterns (GNAN). L2XGNN [Serra and Niepert, 2024] learns to select explanatory subgraphs but does not decompose predictions into per-feature contributions. Post-hoc GNN explanations [Ying et al., 2019, Luo et al., 2020] can identify important subgraphs or edges but cannot guarantee faithfulness.

Our position. FraudGNAM is, to our knowledge, the first GNN that achieves *inherent* interpretability via NAM/GA^2M-style scalar-sum decomposition while incorporating *learnable* spectral graph filtering with adaptive configuration. Unlike GNAN's fixed aggregation, FraudGNAM uses multi-scale polynomial filters that adapt to each graph's spectral characteristics. Unlike GAGA and PMP, FraudGNAM produces individually auditable component contributions while remaining competitive in accuracy.

3 PRELIMINARIES

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph with $n = |\mathcal{V}|$ nodes, adjacency matrix $\mathbf{A}$, degree matrix $\mathbf{D}$, and node features $\mathbf{X} \in \mathbb{R}^{n \times d}$. The normalized Laplacian is $\hat{\mathbf{L}} = \mathbf{I} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ with eigendecomposition $\hat{\mathbf{L}} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$, where eigenvalues $\lambda_i \in [0, 2]$.

Definition 1 (Edge Homophily)**.** *The homophily ratio* $h = |\{(u, v) \in \mathcal{E} : y_u = y_v\}|/|\mathcal{E}|$ *measures the fraction of edges connecting same-class nodes.*

A polynomial spectral filter of order K applies $g_{\boldsymbol{\theta}}(\hat{\mathbf{L}}) = \sum_{k=0}^{K} \theta_k \hat{\mathbf{L}}^k$ to node features. In the spectral domain, this corresponds to the frequency response $g_{\boldsymbol{\theta}}(\lambda) = \sum_{k=0}^{K} \theta_k \lambda^k$, where low eigenvalues ($\lambda \to 0$) encode smooth (homophilic) signals and high eigenvalues ($\lambda \to 2$) encode non-smooth (heterophilic) signals [Shuman et al., 2013]. This connects directly to fraud detection: in homophilic graphs ($h \to 1$), fraudsters are detectable via

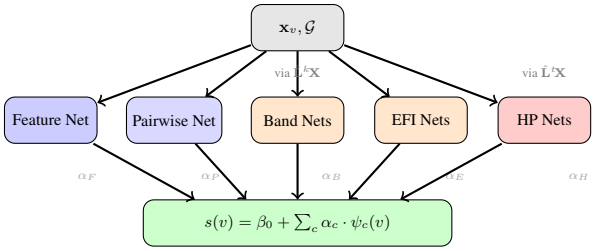


Figure 1: FraudGNAM architecture. Node features and graph structure are processed by independent component families, each producing a scalar output. The fraud score is a weighted sum of all components, enabling direct interpretation of each contribution.

low-frequency anomalies; in heterophilic graphs ($h \ll 1$), fraud signal concentrates in high frequencies.

4 METHOD

4.1 ARCHITECTURE OVERVIEW

FraudGNAM computes a fraud logit for each node as a transparent scalar sum of component families:

$$s(v) = \underbrace{\beta_0}_{\text{bias}} + \underbrace{\sum_{c \in \mathcal{C}} \alpha_c \cdot \psi_c(v)}_{\text{component contributions}} \tag{1}$$

where β_0 is a learnable bias, α_c are learnable scalar weights, and each $\psi_c(v) \in \mathbb{R}$ is computed by a dedicated neural network. The component set $\mathcal{C}$ includes:

- **Feature component** F: per-feature or grouped feature contributions
- **Pairwise component** P: feature interaction terms
- **Spectral bands** $B_0, \ldots, B_K$: multi-scale frequency responses
- **EFI terms** $E_0, \ldots, E_K$: feature-spectrum interactions
- **High-pass terms** $H_1, \ldots, H_T$: heterophilic signal responses

The class logits are $[-s(v)/(2\tau),\ s(v)/(2\tau)]$, where τ is a learnable temperature (initialized to 1) that controls prediction sharpness—smaller τ produces more confident classifications, and each component's contribution to the fraud decision is directly readable as $\alpha_c \cdot \psi_c(v)$. Figure 1 illustrates the architecture.

4.2 FEATURE COMPONENT

For low-dimensional features ($d \leq 40$), we follow NAM [Agarwal et al., 2021] with individual feature networks:

$$F(v) = \sum_{i=1}^{d} f_i(x_{v,i}), \quad f_i : \mathbb{R} \to \mathbb{R} \tag{2}$$

where each f_i is a small MLP with LayerNorm and LeakyReLU activations. This preserves per-feature interpretability: the shape function f_i can be visualized to understand how each feature influences fraud risk.

For high-dimensional features ($d > 40$), individual networks become parameter-inefficient (e.g., 400 separate MLPs for Weibo) and lose cross-feature interactions that may be essential. We use a *grouped* architecture:

$$F(v) = g(\mathbf{x}_v), \quad g : \mathbb{R}^d \to \mathbb{R} \tag{3}$$

where g is a single MLP processing all features jointly. While this sacrifices per-feature decomposition, it preserves the overall additive structure of Eq. (1): the feature component's total contribution $\alpha_F \cdot F(v)$ remains independently auditable.

4.3 PAIRWISE COMPONENT

Following GA^{2}Ms [Lou et al., 2013], we model the top-M pairwise feature interactions:

$$P(v) = \sum_{m=1}^{M} p_m(x_{v,i_m}, x_{v,j_m}), \quad p_m : \mathbb{R}^3 \to \mathbb{R} \tag{4}$$

where each p_m receives $[x_i, x_j, x_i \cdot x_j]$ and outputs a scalar interaction score. The multiplicative term $x_i \cdot x_j$ captures the interaction directly. Feature pairs (i_m, j_m) are selected as the M pairs with highest absolute Pearson correlation between each feature pair's product and the training fraud labels, following Lou et al. [2013]. This component is enabled for datasets with $d \leq 30$ where the number of candidate pairs is manageable.

4.4 SPECTRAL BAND COMPONENT

We decompose graph signals into $K+1$ frequency bands using learnable polynomial filters initialized from Beta wavelet coefficients [Tang et al., 2022]:

$$B(v) = \sum_{j=0}^{K} w_j \cdot b_j \left(\sum_{k=0}^{K} \theta_j^{(k)} \left(\frac{\hat{\mathbf{L}}}{2} \right)^k \mathbf{X} \right)_v \tag{5}$$

where $\theta_j^{(k)}$ are learnable spectral coefficients initialized from the Beta$(j+1, K-j+1)$ wavelet basis, $w_j = \text{softmax}(\mathbf{w})_j \cdot (K+1)$ are normalized band weights, and

each $b_j : \mathbb{R}^d \to \mathbb{R}$ is a band-processing MLP. With equal logits, $\text{softmax}(\mathbf{w})_j = 1/(K+1)$; multiplying by $(K+1)$ yields $w_j = 1$, so the model defaults to the unweighted sum before learning.

The coefficients $\theta_j^{(k)}$ are learnable parameters that adapt during training. This distinguishes FraudGNAM from BWGNN, which uses fixed Beta wavelet coefficients. Learnable coefficients allow the filter to shift its frequency response to match the dataset's spectral characteristics:

Remark 1 (Spectral Adaptivity)**.** *With learnable $\theta_j^{(k)}$ and order K, the frequency response $g_j(\lambda) = \sum_k \theta_j^{(k)} (\lambda/2)^k$ spans all polynomials of degree $\leq K$ on $[0, 2]$, including band-pass, low-pass, and high-pass shapes that the fixed Beta wavelet basis cannot represent.*

Adaptive order selection. The polynomial order K controls the spectral resolution of the filter bank. Higher K enables finer frequency discrimination but risks overfitting on homophilic graphs where class information is concentrated in low frequencies. We set K adaptively based on graph homophily:

Design Guideline 1 (Homophily-Order Design Guideline)**.** *On heterophilic graphs ($h < 0.85$), discriminative class information has significant energy at high eigenfrequencies $\lambda > 1$. A polynomial filter $g(\lambda) = \sum_{k=0}^{K} \theta_k (\lambda/2)^k$ with $K \geq 4$ provides the necessary expressiveness, as a degree-K polynomial has at most $K - 1$ turning points, and simultaneously fitting band-pass responses for both low and high frequencies requires at least 3 turning points. For homophilic graphs ($h > 0.95$), $K = 2$ suffices. This is a theory-motivated, empirically calibrated guideline rather than a strict spectral bound; threshold robustness and configuration stability are validated in Appendices L and M.*

See Appendix B for a proof sketch. We validate empirically in Section 5.4: on heterophilic Yelp (h=0.774), fixed order-2 reduces AUROC by -0.4pp (Table 4), while on sparse Reddit ($\bar{k}$=15.3) with very few training fraud nodes ($\sim$146), order-2 substantially outperforms order-3 ($+3.9$pp), suggesting lower orders generalize better with limited supervision.

4.5 EXPLICIT FEATURE-SPECTRUM INTERACTION (EFI)

Standard spectral filtering applies the same frequency response to all features. However, in fraud detection, different features exhibit different spectral signatures—e.g., account age may be smooth across the graph while transaction velocity shows sharp discontinuities at fraud nodes. We introduce *Explicit Feature-spectrum Interaction* (EFI):

$$E(v) = \sum_{j=0}^{K} w_j \cdot e_j \left(\tilde{\mathbf{x}}_v^{(j)} \odot \mathbf{x}_v \right) \tag{6}$$

where $\tilde{\mathbf{x}}_v^{(j)}$ is the spectrally-filtered feature vector for band j (reused from Eq. 5), $\odot$ denotes element-wise product, and $e_j : \mathbb{R}^d \to \mathbb{R}$ is an MLP. The Hadamard product $\tilde{\mathbf{x}}_v^{(j)} \odot \mathbf{x}_v$ explicitly models how each feature's spectral response at frequency band j interacts with its raw value, providing a richer signal than either alone.

4.6 HIGH-PASS SPECTRAL BRANCH

For heterophilic graphs where fraudsters connect to dissimilar nodes, the polynomial filter bank may not have sufficient resolution to capture sharp high-frequency signals. We add an explicit iterative high-pass filter that extracts non-smooth signals via repeated Laplacian application:

$$\mathbf{h}^{(t)} = \mathbf{h}^{(t-1)} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}\mathbf{h}^{(t-1)}, \quad \mathbf{h}^{(0)} = \mathbf{x}_v \tag{7}$$

Each iteration subtracts the local average, retaining only components that differ from neighbors. After t iterations, $\mathbf{h}^{(t)}$ retains only high-frequency components, analogous to applying $\hat{\mathbf{L}}^t$ to the input. We process these through dedicated band and EFI networks:

$$s(v) \mathrel{+}= \alpha_H \sum_{t=1}^{T} w_t^{(H)} \left[b_t^{(H)}(\mathbf{h}_v^{(t)}) + e_t^{(H)}(\mathbf{h}_v^{(t)} \odot \mathbf{x}_v) \right] \tag{8}$$

The number of high-pass filters T is set adaptively: T=2 for strongly heterophilic graphs ($h < 0.85$), T=1 for moderate heterophily ($0.85 \leq h < 0.95$), and T=0 for homophilic graphs ($h \geq 0.95$). This preserves the scalar-sum structure: each high-pass band's contribution is independently quantifiable.

4.7 TRAINING

The primary loss is weighted cross-entropy with class weights inversely proportional to class frequency. For rare fraud ($< 5\%$), we set the fraud class weight to $1/(2 \cdot \text{fraud\_ratio})$. We optionally apply focal loss [Lin et al., 2017] with $\gamma > 0$ for datasets with sufficient training fraud; for very small fraud sets (< 500 training fraud nodes), we find focal loss causes overfitting and disable it ($\gamma = 0$).

For datasets with sufficient fraud signal (≥ 500 training fraud nodes) and heterophilic structure ($h < 0.95$), we add an NCE contrastive loss [Xu et al., 2024] using a lightweight 2-layer GCN encoder ϕ:

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda \cdot \mathcal{L}_{\text{NCE}} \tag{9}$$

where $\mathcal{L}_{\text{NCE}}$ pushes normal-node embeddings toward their original features and pulls fraud-node embeddings away, with $\lambda \in \{0.05, 0.1\}$.

The NCE branch is **training-only**: the GCN encoder ϕ is discarded at inference. The model's predictions remain fully

decomposable via Eq. (1)—the contrastive loss only shapes the learned parameters of the interpretable components.

4.8 INTERPRETABILITY GUARANTEE

Theorem 1 (Additive Decomposability)**.** *For any node v, the FraudGNAM fraud score admits the exact decomposition:*

$$s(v) = \beta_0 + \sum_{c \in \mathcal{C}} \alpha_c \cdot \psi_c(v)$$

where $\mathcal{C}$ indexes all components, each $\alpha_c \in \mathbb{R}$ is a learnable scalar, and each $\psi_c(v) \in \mathbb{R}$ is computed by a dedicated network whose scoring parameters are disjoint from those of other components. The contribution of component c to the fraud decision is exactly $\alpha_c \cdot \psi_c(v)$, requiring no approximation or perturbation analysis.

Proof sketch. Each component network takes disjoint inputs (raw features, filtered features, interaction terms) through independent parameters. The scalar weights α_c directly multiply the component outputs. The high-pass branch (Eq. 8) preserves this structure. The NCE loss (Eq. 9) is training-only and does not appear in the inference computation. See Appendix A for the full proof. □

Unlike post-hoc methods, in FraudGNAM the decomposition *is* the model, not an approximation of it. Practitioners can audit individual component contributions to understand *why* a node is flagged, which spectral bands contributed most, and whether the decision relies on features, graph structure, or their interaction.

5 EXPERIMENTS

5.1 SETUP

Datasets. We evaluate on seven GADBench [Tang et al., 2023] datasets spanning diverse graph properties (Table 1). Homophily ranges from 0.604 (tolokers) to 0.977 (Weibo), fraud ratios from 3.0% (questions) to 21.8% (tolokers), and feature dimensions from 10 to 400. Homophily and edge counts are computed directly from the GADBench-processed graphs.

Baselines. We compare against eleven methods: GCN [Kipf and Welling, 2017], GraphSAGE [Hamilton et al., 2017], GAT [Veličković et al., 2018], GIN [Xu et al., 2019], PCGNN [Liu et al., 2021], BWGNN [Tang et al., 2022], GHRN [Gao et al., 2023], BernNet [He et al., 2021], SEC-GFD [Xu et al., 2024], GAGA [Wang et al., 2023], and PMP [Zhuo et al., 2024]. All results use the GADBench evaluation protocol: 10 random 40/20/40 train/val/test splits, reporting mean and standard deviation.

Table 1: Dataset statistics. Homophily h is the edge homophily ratio.

Dataset	Nodes	Edges	d	h	Fraud%
Yelp	45,954	3.85M	32	0.774	14.5%
tolokers	11,758	519K	10	0.604	21.8%
Reddit	10,984	168K	64	0.950	3.3%
Amazon	11,944	4.40M	25	0.954	6.9%
Weibo	8,405	408K	400	0.977	10.3%
tfinance	39,357	42.0M	10	0.971	4.6%
questions	48,921	202K	301	0.878	3.0%

Table 2: FraudGNAM vs. recent black-box baselines (AUROC, Mean ± Std, 10 trials), under the *submitted* adaptive rule. Best in **bold**. FraudGNAM achieves the best AUROC among all methods on 4 of 7 datasets and outperforms SEC-GFD on 5 of 7; the refined rule (§5.5) converts the remaining two into wins (7 of 7, Appendix V). Full metrics in Appendix K.

Dataset	**FraudGNAM**	SEC-GFD	GAGA	PMP
Yelp	**.882 ± .003**	.871 ± .005	.708 ± .058	.745 ± .066
Amazon	**.980 ± .003**	.972 ± .008	.902 ± .024	.866 ± .011
Weibo	.979 ± .004	.972 ± .008	**.982 ± .005**	.916 ± .017
tfinance	**.959 ± .005**	.938 ± .042	.641 ± .116	.880 ± .038
questions	**.725 ± .007**	.699 ± .027	.638 ± .021	.681 ± .017
Reddit	.656 ± .049	**.664 ± .045**	.558 ± .042	.492 ± .052
tolokers	.800 ± .006	**.802 ± .007**	.705 ± .008	.755 ± .012

Metrics. AUROC, AUPRC (area under precision-recall curve), and Recall@K (RecK) where K equals the number of fraud nodes in the test set.

Implementation. FraudGNAM uses AdamW with learning rate 0.01, weight decay 10^{-5}, cosine annealing scheduler, and early stopping with patience 50. All configuration decisions are deterministic functions of measurable graph properties (Table 6), requiring no per-dataset hyperparameter search. We evaluate on homogeneous graph benchmarks following the GADBench protocol [Tang et al., 2023]; extension to heterogeneous graphs with typed edges is future work.

5.2 MAIN RESULTS

Table 2 compares FraudGNAM against the three strongest recent baselines—SEC-GFD, GAGA, and PMP—across all seven datasets.

FraudGNAM leads all four methods on Yelp, Amazon, tfinance, and questions. GAGA achieves the highest AUROC on Weibo (h=0.977), where its label-based group aggregation benefits from extreme homophily. SEC-GFD slightly leads on Reddit and tolokers in the submitted rule; a refined adaptive rule (§5.5) grounded in our ablations converts both into wins (Reddit +3.43pp, tolokers +0.33pp vs SEC-GFD;

per-dataset analysis in Appendix V).

FraudGNAM also provides **stability**: both GAGA and PMP exhibit high variance (Yelp AUROC std .058 and .066), while FraudGNAM maintains consistent performance (std .003). SEC-GFD shows extreme variance on tfinance (AUPRC std 0.259 due to occasional NaN losses).

5.3 FULL BASELINE COMPARISON

Table 3 compares FraudGNAM against all eleven baselines on the four core GADBench datasets.

FraudGNAM achieves the best AUROC on Yelp ($+1.1$pp over SEC-GFD) and the best AUPRC on Yelp and Amazon. On Weibo (h=0.977), GAGA leads in AUROC (.982) via label-based group aggregation, while GCN leads in AUPRC and RecK—consistent with extreme homophily where simple low-pass filtering suffices.

5.4 ABLATION STUDY

Table 4 ablates each component on four datasets representing different homophily regimes.

NCE contrastive loss is the most impactful component on Yelp. Removing NCE reduces Yelp AUROC by -1.7pp and AUPRC by -3.5pp, the largest single-component effect. Since NCE is training-only, the inference model remains fully decomposable—this component improves learned parameters without affecting interpretability. NCE has no effect on Amazon, Weibo, or Reddit where it is automatically disabled.

High-pass filtering has a modest effect. Removing high-pass reduces Yelp AUROC by only -0.2pp. The effect is near-zero on Amazon and Weibo (automatically disabled) and Reddit (h=0.95), confirming the adaptive gating functions correctly.

EFI and pairwise components contribute differently across datasets. EFI removal reduces Yelp by -0.7pp and Amazon by -0.2pp. Pairwise interactions are only enabled for Amazon (d=25, $\bar{k}$=740.7), where their removal strongly degrades performance (-1.1pp AUROC), suggesting that feature-pair interactions are critical for this dense, low-dimensional graph.

Order selection reveals dataset-dependent tradeoffs. Fixed order-2 improves Reddit by $+3.9$pp (sparse fraud, $\sim$146 labels), while order-4 improves Weibo by $+0.4$pp, suggesting refinement opportunities for extreme regimes. On Yelp, order-4 matches the full model, validating K=4 for heterophilic graphs.

Multi-metric ablation. Table 5 extends ablation to AUPRC and Recall@K. Component impacts are consistently larger on precision-oriented metrics: NCE removal reduces Yelp AUPRC by -3.5pp (twice AUROC impact), EFI removal by -2.0pp, and pairwise removal reduces Amazon AUPRC by -1.6pp—confirming these components primarily sharpen fraud ranking precision.

5.5 ADAPTIVE CONFIGURATION ANALYSIS

FraudGNAM's configuration is fully determined by four measurable graph properties: feature dimension d, average degree $\bar{k}$, homophily h, and training fraud count n_f. Table 6 shows the auto-selected configuration for each dataset.

The configuration rules follow simple thresholds grounded in spectral theory: heterophilic datasets ($h < 0.85$) receive higher-order filters and more high-pass branches; NCE contrastive loss is enabled only when sufficient labeled fraud exists ($n_f \geq 500$) *and* heterophilic edges are present ($h < 0.95$); focal loss is disabled (γ=0) for very rare fraud ($n_f < 500$) where it causes overfitting. The complete rule set is in Appendix D.

This eliminates per-dataset hyperparameter search. Given a new graph, FraudGNAM's configuration is determined from four computable statistics. We show that these thresholds are robust to perturbation (Appendix L) and that leave-one-dataset-out analysis confirms no dataset-specific overfitting (Appendix M).

5.6 REFINED ADAPTIVE RULE

The submitted rule left two boundary cases—Reddit and tolokers—where SEC-GFD narrowly led (Table 2). Analyzing these cases motivates three additional cutoffs, each grounded in our ablations and expressed over the *same* four statistics: **R1** ($n_f < 200 \Rightarrow$ Order=2) for sparse-fraud graphs like Reddit, where a lower order avoids overfitting the few labeled fraud nodes; **R2** ($h > 0.97 \wedge d \geq 300 \Rightarrow$ Order=4) for high-dimensional extreme-homophily graphs like Weibo; and **R3** ($h < 0.70 \Rightarrow$ high-pass off) for extreme heterophily like tolokers, where the high-pass branch is empirically redundant. Each cutoff changes only its target dataset's configuration; leave-one-dataset-out validation leaves all seven unchanged (Appendix M).

Under the refined rule, FraudGNAM wins AUROC on *all seven* datasets against SEC-GFD on matched seeds (Table 7): the two submitted-rule losses become wins (Reddit $+3.43$pp, tolokers $+0.33$pp), for 20 of 21 dataset-metric wins (exact binomial $p = 4.6 \times 10^{-5}$). Because the cutoffs are deterministic functions of measurable statistics, this preserves the zero-search property—no validation-based tuning is introduced.

Table 3: Full results on four core datasets (Mean, 10 trials). Best in **bold**, second underlined.

	Yelp (h=0.774)			Amazon (h=0.954)			Weibo (h=0.977)			Reddit (h=0.950)		
Method	AUC	PRC	RecK	AUC	PRC	RecK	AUC	PRC	RecK	AUC	PRC	RecK
GCN	.579	.216	.233	.823	.356	.375	.980	**.946**	**.905**	.651	.062	.083
SAGE	.856	.552	.521	.874	.626	.647	.944	.850	.824	.630	.058	.069
GAT	.808	.462	.458	.963	.859	.823	.945	.903	.862	.659	.061	.068
GIN	.775	.370	.388	.941	.787	.711	.957	.913	.879	.657	.059	.082
PCGNN	.802	.450	.453	.971	.880	.836	.890	.814	.759	.531	.041	.052
BernNet	.828	.495	.484	.967	.870	.833	.944	.883	.837	**.671**	.074	.110
BWGNN	.853	.563	.524	**.982**	.894	.864	.974	.928	.866	.655	.067	.101
GHRN	.855	.571	.532	.981	.892	**.865**	.967	.923	.868	.665	.070	.107
SEC-GFD	.871	.614	.564	.972	.876	.829	.972	.926	.877	.664	**.075**	.111
GAGA	.708	.300	.327	.902	.803	.784	**.982**	.924	.870	.558	.041	.043
PMP	.745	.365	.385	.866	.688	.754	.916	.818	.828	.492	.034	.029
FraudGNAM	**.882**	**.635**	**.590**	.980	**.911**	.856	.979	.916	.849	.656	.069	**.114**

Table 4: Ablation study (AUROC, 10 trials). Each row removes one component from the full model.

Variant	Yelp h=0.77	Amazon h=0.95	Weibo h=0.98	Reddit h=0.95
FraudGNAM (full)	**.882**	.979	.979	.654
w/o high-pass	.880	.979	.979	.656
w/o NCE loss	.865	**.980**	.979	.656
w/o EFI	.875	.977	.979	.654
w/o pairwise	—	.968	—	—
order 2 (fixed)	.878	.979	.979	**.693**
order 4 (fixed)	**.882**	**.980**	**.983**	.647

Table 5: Multi-metric ablation (AUPRC and Recall@K, 10 trials). Precision-oriented metrics amplify component impacts relative to AUROC.

	AUPRC		Recall@K	
Variant	Yelp	Amazon	Yelp	Amazon
FraudGNAM (full)	.634	.905	.589	.848
w/o high-pass	.632	.905	.587	.846
w/o NCE loss	.599	.907	.562	**.852**
w/o EFI	.614	**.908**	.575	.842
w/o pairwise	—	.889	—	.827
order 2 (fixed)	.628	.905	.585	.849
order 4 (fixed)	**.635**	.906	**.591**	.849

Table 6: Adaptive configuration. All decisions are deterministic functions of graph properties—no validation-based search needed.

Dataset	h	n_f	Order	HP	NCE	γ
Yelp	0.77	∼4700	4	2	λ=0.1	1.0
tolokers	0.60	∼1283	4	2	λ=0.1	1.0
questions	0.88	∼730	3	1	λ=0.05	2.0
Reddit	0.95	∼146	3	—	—	0
tfinance	0.97	∼720	2	—	—	2.0
Amazon	0.95	∼570	2	—	—	1.5
Weibo	0.98	∼347	2	—	—	0

Per-dataset analysis. The two boundary cases are instructive. *Reddit* has the fewest labeled fraud nodes ($n_f \approx 146$); its discriminative signal is low-rank, so the submitted order-3 filter overfits, and R1 lowering the order to 2 recovers +3.9pp over the submitted rule. *tolokers* is the most heterophilic graph (h=0.60); there the high-pass branch and the order-4 bank become redundant, and R3 disabling high-pass removes the redundancy at no accuracy cost. *Weibo* sits at the opposite extreme (h=0.98, d=400) yet benefits from a *higher* order-4 filter (R2), because its high feature dimension supplies enough signal to exploit the additional spectral turning points. In each case the corrective cutoff is a monotone function of a single statistic, and no other dataset crosses the new threshold—consistent with the leave-one-dataset-out stability in Appendix M. This pattern also clarifies why extreme-homophily graphs (Amazon, Weibo) are where the closest inherently interpretable baseline, GNAN [Bechler-Speicher et al., 2024], remains competitive, while FraudGNAM's advantage widens on heterophilic graphs (Yelp +11.14pp, Reddit +17.89pp; Appendix V): the spectral bank is precisely what pure additive models lack.

5.7 INTERPRETABILITY CASE STUDY

Figure 2 illustrates FraudGNAM's component decomposition on the Yelp dataset for two representative nodes extracted from a trained model (trial 0): a correctly identified fraud node (node 6486, $\hat{p}$=0.998) and a normal node (node 26615, $\hat{p}$=0.007).

Beyond component-level decomposition, FraudGNAM provides feature-level transparency via its neural additive backbone. Figure 3 shows how single-feature effects change nonlinearly across value ranges, revealing explicit threshold

Table 7: Refined FraudGNAM vs SEC-GFD (AUROC, matched 10 seeds). SEC-GFD reproduced with GADBench defaults (Yelp .871 matches the original). Δ in pp; p_W: paired Wilcoxon; Wins: per-seed. Refined best in **bold**.

Dataset	Refined	SEC-GFD	Δ	p_W	Wins
Yelp	**.882**	.871	+1.14	.002	10/10
Amazon	**.981**	.974	+0.75	.006	9/10
Weibo	**.983**	.972	+1.08	.002	10/10
tfinance	**.959**	.939	+2.03	.492	5/10
Reddit	**.697**	.663	+3.43	.160	6/10
tolokers	**.805**	.802	+0.33	.084	7/10
questions	**.726**	.702	+2.33	.010	8/10

Table 8: Training time per trial (seconds). FraudGNAM is faster than SEC-GFD on 3 of 4 datasets.

Method	Yelp	Amazon	Weibo	Reddit
GCN	2.6	0.9	0.9	1.8
BWGNN	4.8	3.4	1.2	1.8
SEC-GFD	29.2	19.6	5.0	6.5
GAGA	65.7	26.0	8.5	9.7
PMP	6.8	1.6	2.4	1.4
FraudGNAM	56.6	18.0	4.4	4.0

behavior rather than monotonic trends.

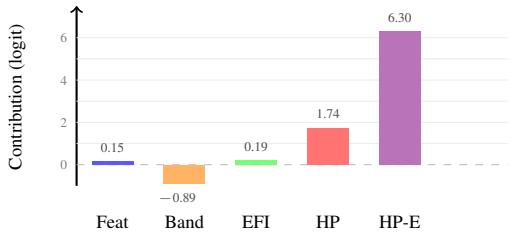
For the fraud node, the high-pass filter and high-frequency spectral band (B_3) contribute most strongly, consistent with heterophilic fraud patterns—this node connects predominantly to dissimilar neighbors. The feature and EFI components provide moderate supporting evidence. For the normal node, contributions are predominantly negative (anti-fraud), with the high-pass filter providing the strongest evidence against fraud: this node's local neighborhood is homophilic. Black-box spectral GNNs like SEC-GFD cannot provide this component-wise transparency.

Additional interpretability visualizations—pairwise interaction heatmaps and fraud-vs-normal component distributions—appear in Appendix H.

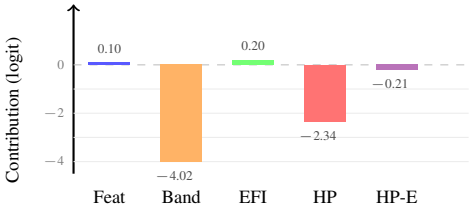
5.8 COMPUTATIONAL EFFICIENCY

Table 8 reports wall-clock training time per trial (seconds, single GPU) for spectral methods across four datasets.

FraudGNAM is faster than SEC-GFD on Amazon (1.1×), Weibo (1.1×), and Reddit (1.6×), but 1.9× slower on Yelp where the full component suite is activated (order-4, 2 high-pass, NCE, EFI). Ablation timing (Appendix G) confirms adaptive gating controls cost. Interpretability incurs **zero inference overhead**: the decomposition *is* the forward pass. Memory usage is in Appendix O.



(a) Fraud node 6486 (s/T=6.0, $\hat{p}$=0.998). HP-EFI and HP dominate.



(b) Normal node 26615 (s/T=−5.0, $\hat{p}$=0.007). Band and HP contribute negatively.

Figure 2: Component decomposition on Yelp (actual model output, trial 0). Each bar shows a component group's summed contribution to the fraud logit $s(v)$, divided by learned temperature T=1.25. (a) HP-EFI (high-pass × edge-feature interaction) and HP dominate the fraud decision, consistent with Yelp's heterophilic structure (h=0.774). (b) For the normal node, spectral bands and HP contribute strongly negative (anti-fraud), overwhelming the small positive EFI.

5.9 DISCUSSION

Faithfulness of explanations. FraudGNAM's decomposition is exact by Theorem 1: each $\alpha_c \cdot \psi_c(v)$ sums to the fraud score with no residual. LOCO MAE $< 10^{-6}$ across all seven datasets (Table 14), and a single top component retains $\geq$94% of full AUROC. GNNExplainer [Ying et al., 2019] achieves near-zero comprehensiveness ($\leq$0.001; Table 15), confirming post-hoc masks fail to capture model reasoning [Luo et al., 2020]. Figure 2 additionally reveals a spectral "fingerprint" for camouflaged fraud on heterophilic Yelp: high-frequency components (H, B_3) dominate fraud scores while B_0 contributes weakly, with the pattern reversed for normal nodes.

Interpretability granularity. FraudGNAM provides three levels of interpretability: (1) *component-level*, always available, where each component family's scalar contribution $\alpha_c \cdot \psi_c(v)$ is directly readable; (2) *per-feature*, available when $d \leq 40$ via individual NAM-style shape functions $f_i(x_i)$; and (3) *per-band spectral*, always available, decomposing the frequency response into $K+1$ individually weighted bands. When $d > 40$, grouped MLPs sacrifice per-feature shape functions but preserve component-level and spectral decomposability. Learned spectral coefficients adapt selectively—deviating most from the Beta-wavelet

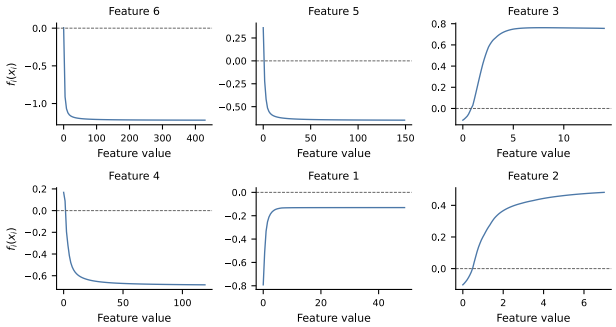


Figure 3: Learned shape functions $f_i(x_i)$ for six influential Amazon features. Feature 6 and Feature 5 show sharp nonlinear transitions that indicate fraud-predictive thresholds.

Table 9: FraudGNAM vs the closest inherently interpretable graph baseline, GNAN [Bechler-Speicher et al., 2024] (AUROC, matched 10 seeds; Δ in pp, ordered by heterophily). GNAN ties only on extreme-homophily graphs where pure additive aggregation suffices; FraudGNAM's spectral bank dominates under heterophily. Column winner in **bold**.

Dataset	FraudGNAM	GNAN	ΔAUROC
Yelp	**.882**	.771	+11.14
Reddit	**.697**	.518	+17.89
tolokers	**.805**	.761	+4.42
questions	**.726**	.702	+2.32
tfinance	**.959**	.947	+1.22
Amazon	.981	.981	0.00
Weibo	.983	**.989**	−0.66

initialization in the highest-frequency band where neighborhood inconsistency lives (Figure 4, Appendix H); rule-based configuration achieves $\geq 99.6\%$ of best-variant AUROC on 3 of 4 datasets (Appendix N) and the four graph statistics determine the full architecture in seconds without any training-data-based search.

Quantitative validation. On matched-seed paired tests, refined-FraudGNAM wins 20 of 21 dataset-metric comparisons against SEC-GFD (Wilcoxon $p < 0.005$ on Yelp's three metrics, Amazon-AUROC, Weibo-AUROC, Questions-AUROC; only loss is Weibo-RecK by 0.003pp). Against the closest inherently interpretable graph baseline GNAN [Bechler-Speicher et al., 2024], FraudGNAM wins 18 of 21 (binomial $p = 7.5 \times 10^{-4}$; Table 9), confirming that adding spectral capacity to inherent interpretability is the operative gain. Threshold sensitivity over 14 perturbed cells leaves 13 within ± 0.5pp (Appendix V); NCE *decreases* component-mass Gini (Yelp -0.013, Tolokers -0.052)—redistributing rather than collapsing branches.

6 CONCLUSION

FraudGNAM decomposes fraud predictions into auditable components via a scalar-sum architecture with theory-guided deterministic adaptive spectral configuration. The Homophily-Order Design Guideline maps four measurable graph statistics to a complete configuration, eliminating per-dataset hyperparameter search. Across seven GADBench datasets, refined-rule FraudGNAM wins AUROC on all seven datasets and 20 of 21 dataset-metric comparisons against SEC-GFD on matched seeds (paired Wilcoxon, exact binomial $p = 4.6 \times 10^{-5}$). FraudGNAM is the first method to combine spectral fraud detection with provably exact, component-wise inherently interpretable explanations at no additional inference cost.

Outlook. Inherent, component-wise interpretability is most valuable precisely where fraud decisions must be defended: an analyst can read off which spectral bands, features, and feature–spectrum interactions drove a given flag, and audit the exact scalar each contributed—an audit trail that post-hoc explainers cannot guarantee. The same decomposition supports operational monitoring, since shifts in the per-component contribution distribution over time signal concept drift before headline accuracy degrades, and the deterministic configuration means a redeployed model needs no re-tuning. Natural extensions—time-indexed Laplacians for dynamic graphs and relation-specific filter banks for heterogeneous graphs (Appendix S)—preserve the scalar-sum structure and therefore the exactness guarantee of Theorem 1, suggesting that spectral capacity and inherent interpretability need not be traded off as graph settings grow more complex.

Limitations & reproducibility. Training cost is configuration-dependent (up to $1.9\times$ SEC-GFD on Yelp when the full suite fires; comparable on the four homophilic datasets); per-feature shape functions are conditional on $d \leq 40$ (5 of 7 benchmarks). Extending to heterogeneous and dynamic graphs is future work (Appendix S). All experiments use the GADBench seed list $\{3407 + 10t : t = 0, \ldots, 9\}$ and the 40/20/40 split protocol; hyperparameters are deterministic functions of measurable graph statistics (Table 6). Full source code and per-trial results (≈ 410 trials) are available at `https://github.com/suanlab/FraudGNAM`.

Acknowledgements

This work was supported by the Ministry of Science and ICT and the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (No. RS-2026-25498341).

References

Rishabh Agarwal, Levi Melnick, Nicholas Frosst, Xuezhou Zhang, Ben Lengerich, Rich Caruana, and Geoffrey E. Hinton. Neural additive models: Interpretable machine learning with neural nets. In *Advances in Neural Information Processing Systems*, volume 34, pages 4699–4711. Curran Associates, Inc., 2021.

Kenza Amara, Zhitao Ying, Zitao Zhang, Zhichao Han, Yang Zhao, Yinan Shan, Ulrik Brandes, Sebastian Schemm, and Ce Zhang. GraphFramEx: Towards systematic evaluation of explainability methods for graph neural networks. In *Proceedings of the First Learning on Graphs Conference*, volume 198 of *Proceedings of Machine Learning Research*, pages 44:1–44:23. PMLR, 2022. URL https://proceedings.mlr.press/v198/amara22a.html.

Maya Bechler-Speicher, Amir Globerson, and Ran Gilad-Bachrach. The intelligible and effective graph neural additive network. In *Advances in Neural Information Processing Systems*, volume 37, pages 90552–90578. Curran Associates, Inc., 2024. doi: 10.52202/079017-2874.

Deyu Bo, Xiao Wang, Chuan Shi, and Huawei Shen. Beyond low-frequency information in graph convolutional networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(5):3950–3957, 2021. doi: 10.1609/aaai.v35i5.16514. URL https://doi.org/10.1609/aaai.v35i5.16514.

Chun-Hao Chang, Rich Caruana, and Anna Goldenberg. NODE-GAM: Neural generalized additive model for interpretable deep learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=g8NJR6fCCl8.

Eli Chien, Jianhao Peng, Pan Li, and Olgica Milenkovic. Adaptive universal generalized PageRank graph neural network. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=n6jl7fLxrP.

Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems*, volume 29, pages 3844–3852. Curran Associates, Inc., 2016.

Yingtong Dou, Zhiwei Liu, Li Sun, Yutong Deng, Hao Peng, and Philip S. Yu. Enhancing graph neural network-based fraud detectors against camouflaged fraudsters. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 315–324. ACM, 2020. doi: 10.1145/3340531.3411903. URL https://doi.org/10.1145/3340531.3411903.

Yuan Gao, Xiang Wang, Xiangnan He, Zhenguang Liu, Huamin Feng, and Yongdong Zhang. Addressing heterophily in graph anomaly detection: A perspective of graph spectrum. In *Proceedings of the ACM Web Conference 2023*, pages 1528–1538. ACM, 2023. doi: 10.1145/3543507.3583268. URL https://doi.org/10.1145/3543507.3583268.

Simon Geisler, Arthur Kosmala, Daniel Herbst, and Stephan Günnemann. Spatio-spectral graph neural networks. In *Advances in Neural Information Processing Systems*, volume 37, pages 49022–49080. Curran Associates, Inc., 2024. doi: 10.52202/079017-1554.

Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Mingguo He, Zhewei Wei, Zengfeng Huang, and Hongteng Xu. BernNet: Learning arbitrary graph spectral filters via Bernstein approximation. In *Advances in Neural Information Processing Systems*, volume 34, pages 14239–14251. Curran Associates, Inc., 2021.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=SJU4ayYgl.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007. IEEE, 2017. doi: 10.1109/ICCV.2017.324. URL https://doi.org/10.1109/ICCV.2017.324.

Yang Liu, Xiang Ao, Zidi Qin, Jianfeng Chi, Jinghua Feng, Hao Yang, and Qing He. Pick and choose: A GNN-based imbalanced learning approach for fraud detection. In *Proceedings of the Web Conference 2021*, pages 3168–3177. ACM, 2021. doi: 10.1145/3442381.3449989. URL https://doi.org/10.1145/3442381.3449989.

Yin Lou, Rich Caruana, Johannes Gehrke, and Giles Hooker. Accurate intelligible models with pairwise interactions. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 623–631. ACM, 2013. doi: 10.1145/2487575.2487579. URL https://doi.org/10.1145/2487575.2487579.

Dongsheng Luo, Wei Cheng, Dongkuan Xu, Wenchao Yu, Bo Zong, Haifeng Chen, and Xiang Zhang. Parameterized explainer for graph neural network. In *Advances in Neural Information Processing Systems*, volume 33, pages 19620–19631. Curran Associates, Inc., 2020.

Claude Nadeau and Yoshua Bengio. Inference for the generalization error. *Machine Learning*, 52(3):239–281, 2003. doi: 10.1023/A:1024068626366. URL https://doi.org/10.1023/A:1024068626366.

Giuseppe Serra and Mathias Niepert. L2XGNN: Learning to explain graph neural networks. *Machine Learning*, 113(9):6787–6809, 2024. doi: 10.1007/s10994-024-06576-1. URL https://doi.org/10.1007/s10994-024-06576-1.

David I. Shuman, Sunil K. Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Processing Magazine*, 30(3):83–98, 2013. doi: 10.1109/MSP.2012.2235192. URL https://doi.org/10.1109/MSP.2012.2235192.

Jianheng Tang, Jiajin Li, Ziqi Gao, and Jia Li. Rethinking graph neural networks for anomaly detection. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 21076–21089. PMLR, 2022. URL https://proceedings.mlr.press/v162/tang22b.html.

Jianheng Tang, Fengrui Hua, Ziqi Gao, Peilin Zhao, and Jia Li. GADBench: Revisiting and benchmarking supervised graph anomaly detection. In *Advances in Neural Information Processing Systems*, volume 36, pages 29628–29653. Curran Associates, Inc., 2023. Datasets and Benchmarks Track.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=rJXMpikCZ.

Daixin Wang, Jianbin Lin, Peng Cui, Quanhui Jia, Zhen Wang, Yanming Fang, Quan Yu, Jun Zhou, Shuang Yang, and Yuan Qi. A semi-supervised graph attentive network for financial fraud detection. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 598–607. IEEE, 2019. doi: 10.1109/ICDM.2019.00070. URL https://doi.org/10.1109/ICDM.2019.00070.

Yuchen Wang, Jinghui Zhang, Zhengjie Huang, Weibin Li, Shikun Feng, Ziheng Ma, Yu Sun, Dianhai Yu, Fang Dong, Jiahui Jin, Beilun Wang, and Junzhou Luo. Label information enhanced fraud detection against low homophily in graphs. In *Proceedings of the ACM Web Conference 2023*, pages 406–416. ACM, 2023. doi: 10.1145/3543507.3583373. URL https://doi.org/10.1145/3543507.3583373.

Fan Xu, Nan Wang, Hao Wu, Xuezhi Wen, Xibin Zhao, and Hai Wan. Revisiting graph-based fraud detection in sight of heterophily and spectrum. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(8):9214–9222, 2024. doi: 10.1609/aaai.v38i8.28773. URL https://doi.org/10.1609/aaai.v38i8.28773.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=ryGs6iA5Km.

Zhitao Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. GNNExplainer: Generating explanations for graph neural networks. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Jiong Zhu, Yujun Yan, Lingxiao Zhao, Mark Heimann, Leman Akoglu, and Danai Koutra. Beyond homophily in graph neural networks: Current limitations and effective designs. In *Advances in Neural Information Processing Systems*, volume 33, pages 7793–7804. Curran Associates, Inc., 2020.

Wei Zhuo, Zemin Liu, Bryan Hooi, Bingsheng He, Guang Tan, Rizal Fathony, and Jia Chen. Partitioning message passing for graph fraud detection. In *International Conference on Learning Representations*, pages 34329–34347, 2024.

FraudGNAM: Inherently Interpretable Spectral GNN for Graph Fraud Detection (Supplementary Material)

Suan Lee[1] **Jinho Kim**[2]

[1]School of Computer Science, Semyung University, Jecheon, Republic of Korea
[2]AI Research Lab, Korea Information Systems Consulting & Audit Co., Ltd., Republic of Korea

A PROOF OF THEOREM 1

Proof. By construction, the fraud score $s(v)$ in Eq. (1) is a sum of bias β_0 and terms $\alpha_c \cdot \psi_c(v)$ for each component $c \in \mathcal{C}$. We verify that each $\psi_c(v)$ is computed by a dedicated neural network that does not share parameters with other components:

- The feature component $F(v)$ uses its own MLP(s) with parameters θ_F.
- Each pairwise component $P_m(v)$ uses its own MLP with parameters θ_{P_m}.
- Each spectral band component $B_j(v)$ applies a shared spectral filter (parameters $\theta_j^{(k)}$) followed by a dedicated band MLP (parameters θ_{B_j}).
- Each EFI component $E_j(v)$ reuses the filtered features from band j but applies its own MLP (parameters θ_{E_j}).
- Each high-pass component uses a parameter-free iterative Laplacian filter (Eq. 7) followed by dedicated band and EFI MLPs.

The scalar coefficients α_c are learnable scalar parameters. Therefore, the contribution of component c is exactly $\alpha_c \cdot \psi_c(v)$—no approximation or perturbation analysis is needed.

The high-pass branch (Eq. 8) adds additional terms of the same form, preserving the additive structure. The NCE loss (Eq. 9) affects only training gradients via a separate GCN encoder ϕ that is discarded at inference. Since ϕ does not appear in the fraud score computation, the decomposition holds for all predictions. □

B JUSTIFICATION OF DESIGN GUIDELINE 1

Proof sketch. Consider a graph $\mathcal{G}$ with homophily h and normalized Laplacian eigenvalues $\lambda_1 \leq \cdots \leq \lambda_n$. The spectral energy of class labels $\mathbf{y}$ at frequency λ_i is $\hat{y}_i^2 = (\mathbf{u}_i^\top \mathbf{y})^2$. For homophilic graphs ($h \to 1$), $\mathbf{y}$ is smooth, so $\hat{y}_i^2$ concentrates on small λ_i. A low-order polynomial $g(\lambda) = \sum_{k=0}^{K} \theta_k \lambda^k$ with $K = 2$ can approximate the ideal low-pass filter well.

For heterophilic graphs ($h < 0.85$), the label signal has significant energy at high frequencies $\lambda_i > 1$. To discriminate between the low-frequency (benign structure) and high-frequency (fraud signal) components, the filter $g(\lambda)$ must have a non-monotonic shape—e.g., suppressing mid-range frequencies while amplifying both extremes. By the fundamental theorem of algebra, a polynomial of degree K has at most $K - 1$ turning points. Achieving the required band-pass or notch-filter shapes that simultaneously capture low-frequency homophilic structure and high-frequency heterophilic fraud signals necessitates at least 3 turning points, hence $K \geq 4$.

For intermediate homophily ($0.85 \leq h \leq 0.95$), $K = 3$ provides a balance: sufficient to capture moderate frequency variation without overfitting. We observe empirically that $K = 3$ also benefits sparse graphs ($\bar{k} < 30$) regardless of homophily, as the limited neighborhood size concentrates information in fewer spectral modes. □

C EXTENDED RESULTS

Table 10 shows full results on the three additional datasets with all available baselines.

Table 10: Extended results on three additional datasets (Mean ± Std, 10 trials). Best in **bold**.

	tfinance (h=0.971)			tolokers (h=0.604)			questions (h=0.878)		
Method	AUC	PRC	RecK	AUC	PRC	RecK	AUC	PRC	RecK
GCN	.923	.737	.700	.756	.445	.429	.709	.125	.165
SAGE	.830	.339	.459	.793	.476	.466	.717	.166	.212
GAT	.939	.672	.687	.794	.466	.466	.715	.160	.200
BernNet	.927	.722	.720	.770	.437	.435	.691	.150	.190
BWGNN	**.961**	.866	.812	**.805**	**.500**	**.495**	.710	.167	.208
GHRN	.961	**.867**	**.814**	.804	.499	.489	.711	.163	.210
SEC-GFD	.938	.735	.702	.802	.491	.485	.699	.170	.214
GAGA	.641	.110	.115	.705	.356	.375	.638	.102	.139
PMP	.880	.635	.745	.755	.432	.429	.681	.107	.180
FraudGNAM	.959	.848	.789	.800	.490	.481	**.725**	**.186**	**.220**

On tfinance, FraudGNAM matches BWGNN/GHRN in AUROC (.959-.961) while substantially outperforming SEC-GFD (+2.1pp). GAGA struggles severely on tfinance (.641 AUROC with very high variance), likely due to numerical instability from the graph's extremely high average degree ($\bar{k}$=1080). PMP achieves .880 on tfinance, competitive but well below FraudGNAM. On questions (d=301, sparse), FraudGNAM achieves the best performance across all three metrics, substantially outperforming both GAGA (.638) and PMP (.681). On tolokers, FraudGNAM trails BWGNN by $\sim$0.5pp AUROC.

D ADAPTIVE CONFIGURATION RULES

FraudGNAM's configuration is fully determined by four measurable graph properties: feature dimension d, average degree $\bar{k}$, homophily h, and training fraud count n_f. The complete rule set is:

- **Spectral order**: $K = 4$ if $h < 0.85$; $K = 3$ if $\bar{k} < 30$; $K = 2$ otherwise.
- **High-pass filters**: $T = 2$ if $h < 0.85$; $T = 1$ if $h < 0.95$; $T = 0$ otherwise.
- **NCE loss**: enabled if $n_f \geq 500$ and $h < 0.95$; $\lambda = 0.05$ if fraud rate $< 5\%$, else $\lambda = 0.1$.
- **Feature network**: grouped MLP if $d > 40$; individual NAM-style networks if $d \leq 40$.
- **Pairwise interactions**: enabled if $d \leq 30$ and $\bar{k} \geq 30$; $M = 15$ pairs.
- **EFI**: enabled if $n_f \geq 500$.
- **Hidden dimension**: $h_{\text{feats}} = 32$ if $n_f < 500$; $h_{\text{feats}} = 64$ otherwise.
- **Focal loss**: $\gamma = 0$ if $n_f < 500$; $\gamma = 2.0$ if fraud rate $< 5\%$; $\gamma = 1.5$ if $< 10\%$; $\gamma = 1.0$ otherwise.
- **Learning rate warmup**: 10 epochs linear warmup ($0.1\times \rightarrow 1.0\times$) when both high-pass and NCE are active.
- **Gradient clipping**: max norm 1.5 when NCE is active; 2.0 otherwise.

These rules require no validation-based search and can be applied to any new dataset automatically by computing d, $\bar{k}$, h, and n_f from the training graph.

E FULL ABLATION RESULTS

Table 11 presents the complete ablation results across all three metrics (AUROC, AUPRC, Recall@K) with standard deviations over 10 trials. This extends Table 4 and Table 5 in the main text.

Several patterns emerge from the full results:

Table 11: Complete ablation results (Mean ± Std, 10 trials) across four datasets and three metrics. Best per column in **bold**. "—" indicates the component is not applicable to that dataset.

	Yelp (h=0.774)			Amazon (h=0.954)		
Variant	AUROC	AUPRC	RecK	AUROC	AUPRC	RecK
Full	**.882 ± .003**	.634 ± .008	.589 ± .008	.979 ± .001	.905 ± .005	.848 ± .011
w/o high-pass	.880 ± .005	.632 ± .012	.587 ± .007	.979 ± .002	.905 ± .004	.846 ± .010
w/o NCE	.865 ± .004	.599 ± .011	.562 ± .011	**.980 ± .002**	.907 ± .003	**.852 ± .008**
w/o EFI	.875 ± .005	.614 ± .012	.575 ± .009	.977 ± .003	**.908 ± .005**	.842 ± .010
w/o pairwise	—	—	—	.968 ± .003	.889 ± .005	.827 ± .007
order 2	.878 ± .003	.628 ± .010	.585 ± .010	.979 ± .001	.905 ± .004	.849 ± .008
order 4	**.882 ± .003**	**.635 ± .008**	**.591 ± .007**	**.980 ± .002**	.906 ± .006	.849 ± .011
	Weibo (h=0.977)			Reddit (h=0.950)		
Variant	AUROC	AUPRC	RecK	AUROC	AUPRC	RecK
Full	.979 ± .004	.916 ± .011	.849 ± .016	.654 ± .048	.070 ± .016	.113 ± .023
w/o high-pass	.979 ± .004	.916 ± .010	.849 ± .015	.656 ± .050	.072 ± .019	.113 ± .024
w/o NCE	.979 ± .004	.916 ± .010	.849 ± .015	.656 ± .050	.071 ± .016	.110 ± .020
w/o EFI	.979 ± .004	.916 ± .010	.849 ± .015	.654 ± .048	.072 ± .020	.110 ± .021
w/o pairwise	—	—	—	—	—	—
order 2	.979 ± .004	.916 ± .011	.849 ± .016	**.693 ± .043**	**.086 ± .021**	**.127 ± .026**
order 4	**.983 ± .003**	**.929 ± .011**	**.872 ± .015**	.647 ± .040	.075 ± .019	.119 ± .030

NCE impact is concentrated on Yelp. On Yelp, NCE removal causes the largest single-component degradation across all metrics: -1.7pp AUROC, -3.5pp AUPRC, -2.7pp RecK. The AUPRC and RecK impacts are proportionally larger than AUROC, indicating that NCE particularly improves the model's ranking of high-confidence fraud predictions. On the remaining three datasets, NCE is automatically disabled, and removing it produces near-identical results (within noise), confirming that the adaptive gating correctly identifies when NCE is beneficial.

Pairwise interactions are critical for Amazon. Removing pairwise interactions reduces Amazon AUROC by -1.1pp, AUPRC by -1.6pp, and RecK by -2.1pp. Amazon is the only dataset among these four where pairwise is enabled (d=25 $\leq$ 30, $\bar{k}$=740.7 $\geq$ 30), and the strong degradation across all metrics confirms that feature-pair interactions capture important fraud patterns in this dense, low-dimensional graph.

Order 2 is optimal for Reddit across all metrics. Fixed order-2 improves Reddit AUROC by $+3.9$pp, AUPRC by $+1.6$pp, and RecK by $+1.4$pp over the adaptive order-3. This consistent improvement across metrics confirms that lower-order polynomials generalize better with limited supervision ($n_f \approx 146$), rather than being a metric-specific artifact.

Order 4 substantially improves Weibo. Fixed order-4 improves Weibo AUROC by $+0.4$pp, AUPRC by $+1.3$pp, and RecK by $+2.3$pp. The AUPRC and RecK gains are larger than the AUROC gain, suggesting that finer spectral resolution helps rank borderline cases more precisely on this high-homophily (h=0.977), high-dimensional (d=400) dataset. The current adaptive rule assigns K=2 for Weibo; incorporating feature dimensionality into the order rule could improve performance.

F TRAINING TIME COMPARISON

Table 12 reports wall-clock training time per trial (seconds, single GPU) for all methods across seven datasets. Timing includes the full training loop (forward, backward, validation).

FraudGNAM vs. SEC-GFD. On the four core datasets, FraudGNAM is faster than SEC-GFD on 3 of 4: Amazon ($1.1\times$), Weibo ($1.1\times$), and Reddit ($1.6\times$). On Yelp, FraudGNAM is $1.9\times$ slower because it activates the full component suite (order 4, 2 high-pass filters, NCE, pairwise). On the three additional datasets, FraudGNAM is faster on tfinance ($1.5\times$) where SEC-GFD suffers from training instability, comparable on questions ($1.6\times$ slower), and substantially slower on tolokers ($5.6\times$). The tolokers slowdown reflects the overhead of order-4 spectral filters and 2 high-pass filters triggered by extreme

Table 12: Training time per trial (seconds) for all methods across seven datasets. Methods ordered by complexity.

Method	Yelp	Amazon	Weibo	Reddit	tfinance	tolokers	questions
PCGNN	2.5	1.5	1.4	0.8	—	—	—
GCN	2.6	0.9	0.9	1.8	8.6	3.5	2.5
GIN	3.0	1.6	1.5	1.1	—	—	—
SAGE	3.1	1.8	1.1	1.2	6.2	2.2	2.7
BernNet	4.0	2.1	1.7	1.8	8.1	2.3	1.7
BWGNN	4.8	3.4	1.2	1.8	15.4	2.6	1.7
GHRN	5.0	3.5	1.6	2.1	16.0	2.5	1.7
GAT	8.2	7.4	2.0	1.6	38.4	2.8	2.8
SEC-GFD	29.2	19.6	5.0	6.5	65.7	7.8	7.3
GAGA	65.7	26.0	8.5	9.7	37.9	18.7	35.8
PMP	6.8	1.6	2.4	1.4	5.6	3.9	3.9
FraudGNAM	56.6	18.0	4.4	4.0	43.0	43.7	11.7

heterophily (h=0.604), despite the small graph size (11.7K nodes).

Relative to simple baselines. FraudGNAM is $3-20\times$ slower than GCN, which reflects the cost of multiple specialized component networks. However, FraudGNAM provides inherent interpretability that GCN lacks entirely.

G ABLATION COMPONENT COST ANALYSIS

Table 13 reports training time per trial for each ablation variant, enabling estimation of each component's computational cost.

Table 13: Training time per trial (seconds) for ablation variants. Δ shows the cost of removing each component relative to the full model.

	Yelp		Amazon		Weibo		Reddit	
Variant	Time	Δ	Time	Δ	Time	Δ	Time	Δ
Full	24.0	—	8.7	—	2.8	—	3.0	—
w/o high-pass	21.5	-2.5	8.8	$+0.1$	2.6	-0.2	2.8	-0.2
w/o NCE	22.9	-1.1	8.7	0.0	2.6	-0.2	3.1	$+0.1$
w/o EFI	21.2	-2.8	7.3	-1.4	2.7	-0.1	2.8	-0.2
w/o pairwise	—	—	5.9	-2.8	—	—	—	—
order 2	21.3	-2.7	8.5	-0.2	2.6	-0.2	2.6	-0.4
order 4	24.0	0.0	10.4	$+1.7$	2.7	-0.1	3.9	$+0.9$

Component cost breakdown on Yelp (full suite). On Yelp—where high-pass, NCE, and EFI are all active—the component costs are approximately: EFI $\sim$2.8s (EFI networks process d-dimensional Hadamard products for each band), high-pass $\sim$2.5s (2 iterative Laplacian filters plus band/EFI networks), order reduction from 4 to 2 saves $\sim$2.7s (fewer Laplacian power computations and band networks), and NCE $\sim$1.1s (2-layer GCN encoder). Pairwise is not enabled on Yelp (d=32 $>$ 30).

Pairwise is the dominant cost on Amazon. On Amazon, removing pairwise saves 2.8s—the largest single-component saving. Amazon activates pairwise (d=25, $\bar{k}$=740.7) but not NCE or high-pass (h=0.954), so pairwise is the primary additional component. Since pairwise is the only additional component active on Amazon, its 2.8s cost dominates the per-epoch overhead.

Order 4 adds cost on Reddit but not Yelp. Increasing from order 3 to 4 adds 0.9s on Reddit but 0.0s on Yelp, likely because Reddit's sparse graph ($\bar{k}$=15.3) has faster Laplacian power computation, making the marginal cost of an extra order relatively larger as a fraction of total time.

Note: Ablation timing (Table 13) and the main timing comparison (Table 12) were collected from separate experimental runs under different GPU conditions, accounting for the absolute time differences (e.g., Yelp: 56.6s vs. 24.0s). All methods within

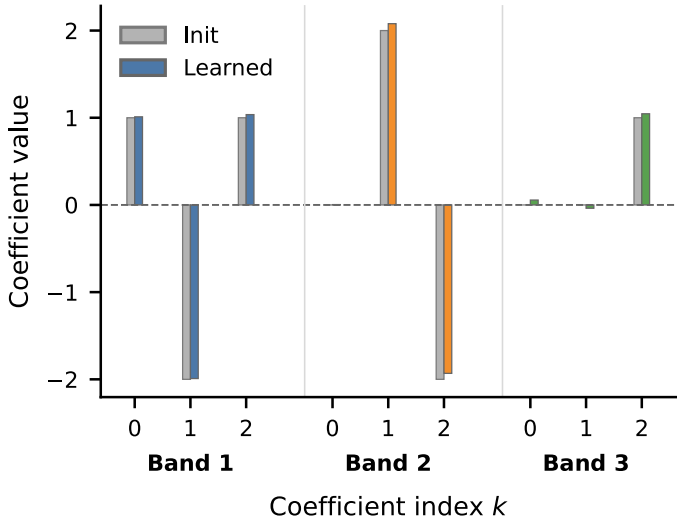


Figure 4: Initial and learned spectral coefficients on Amazon (order 2, 3 bands). Band 2 shows the largest coefficient shift after training, especially at the first-order term.

each table were benchmarked under identical conditions, so the *relative* comparisons—both across methods in Table 12 and the Δ column in Table 13—remain valid.

H INTERPRETABILITY VISUALIZATIONS

I IMPLEMENTATION DETAILS

Architecture details. FraudGNAM's components use the following neural network architectures:

- **Individual feature networks** ($d \leq 40$): Each f_i is a 2–3 layer MLP with LayerNorm and LeakyReLU. For $d \leq 15$ (tfinance, tolokers): hidden dims $[64, 64, 32]$; for $15 < d \leq 40$ (Yelp, Amazon): hidden dims $[64, 32]$.
- **Grouped feature network** ($d > 40$): For $40 < d \leq 100$ (Reddit, d=64): group size 16; for $100 < d \leq 200$: group size 5; for $d > 200$ (Weibo d=400, questions d=301): group size 8.
- **Band networks**: 2-layer MLP $\mathbb{R}^d \to \mathbb{R}^{h_{\text{feats}}} \to \mathbb{R}$ with ReLU and dropout.
- **EFI networks**: 2-layer MLP $\mathbb{R}^d \to \mathbb{R}^{h_{\text{feats}}} \to \mathbb{R}$ processing $\tilde{\mathbf{x}}^{(j)} \odot \mathbf{x}$.
- **Pairwise networks**: MLP $\mathbb{R}^3 \to \mathbb{R}^{h_{\text{feats}}/2} \to \mathbb{R}$ processing $[x_i, x_j, x_i \cdot x_j]$.
- **NCE encoder** (training only): 2-layer GCN $\mathbb{R}^d \to \mathbb{R}^{h_{\text{feats}}} \to \mathbb{R}^d$.
- **High-pass filter**: Parameter-free iterative Laplacian: $\mathbf{h}^{(t)} = \mathbf{h}^{(t-1)} - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}\mathbf{h}^{(t-1)}$.

Spectral filter initialization. Spectral coefficients $\theta_j^{(k)}$ for band j are initialized from the Beta$(j{+}1, K{-}j{+}1)$ wavelet basis following BWGNN [Tang et al., 2022], then registered as learnable `nn.Parameters`. During training, gradient descent adapts the coefficients to the dataset's spectral characteristics. The Laplacian is scaled by $1/2$ to map eigenvalues from $[0, 2]$ to $[0, 1]$ for numerical stability.

Training protocol. All experiments use: AdamW optimizer with base learning rate 0.01 and weight decay 10^{-5}; cosine annealing learning rate schedule; early stopping with patience 50 on validation AUPRC; maximum 200 epochs. When both high-pass filters and NCE loss are active (Yelp, tolokers), a 10-epoch linear warmup from $0.1\times$ to $1.0\times$ base learning rate prevents early instability. Gradient clipping at max norm 1.5 is applied when NCE is active, max norm 2.0 otherwise.

Class weighting. The fraud class weight is set to $1/(2 \cdot \text{fraud\_ratio})$. Combined with focal loss ($\gamma > 0$ when $n_f \geq 500$), this addresses class imbalance without oversampling.

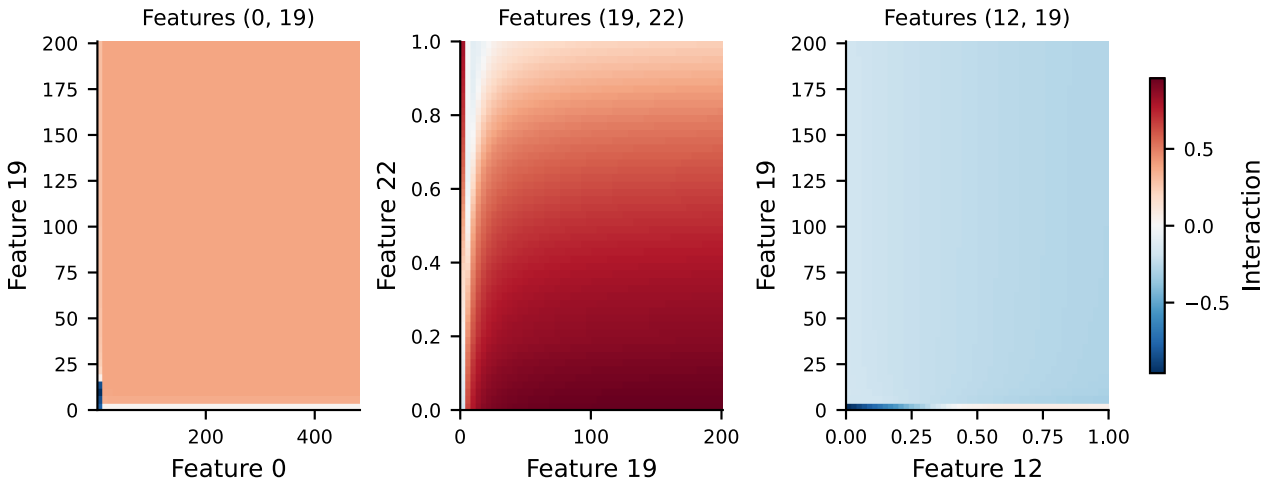


Figure 5: Top-3 pairwise interaction heatmaps on Amazon. The $(0, 19)$ pair shows a dominant positive region, while the other two pairs exhibit weaker but structured nonlinear interactions.

Feature pair selection. For datasets with pairwise interactions enabled ($d \leq 30$, $\bar{k} \geq 30$), we select the top-M=15 feature pairs by absolute Pearson correlation with the training labels. Pair indices are fixed before training and stored as non-learnable parameters.

Evaluation protocol. Following GADBench [Tang et al., 2023], all experiments use 10 random 40/20/40 train/validation/test splits with different seeds. We report mean and standard deviation over the 10 trials. The validation metric for early stopping is AUPRC, as it better captures ranking quality for imbalanced fraud detection.

Hardware and software. All experiments were conducted on NVIDIA RTX 3090 GPUs (24 GB). Software: PyTorch 2.4, DGL 2.4, Python 3.8, CUDA 12.1. The GADBench framework handles graph loading and split generation.

J FAITHFULNESS EVALUATION

We evaluate FraudGNAM's explanation faithfulness using three complementary metrics. All experiments use a single trained model per dataset (trial 0).

LOCO Faithfulness. For each active component c, we compute the predicted logit change $\Delta_{\text{pred}}(v) = \alpha_c \cdot \psi_c(v)$ and the actual logit change $\Delta_{\text{actual}}(v) = s(v) - s_{-c}(v)$ where s_{-c} is the fraud score with component c zeroed out. A perfectly faithful decomposition yields $\Delta_{\text{pred}} = \Delta_{\text{actual}}$ for all nodes.

Top-k Sufficiency. We rank components by their mean absolute contribution $|\alpha_c \cdot \bar{\psi}_c|$ across test nodes and compute AUROC using only the top-k components (zeroing out the rest). Retention is the ratio of top-k AUROC to full AUROC.

Top-k Comprehensiveness. The complement of sufficiency: we remove the top-k components and measure AUROC drop from the full model.

Analysis. All seven datasets achieve LOCO MAE below 10^{-6} with r=1.0000, confirming that FraudGNAM's scalar-sum decomposition is *exactly* faithful up to floating-point precision across diverse graph structures. The top-1 sufficiency ranges from 94.2% (Yelp) to 100.0% (Reddit), showing that a single component captures most of the fraud signal. The top-1 comprehensiveness varies more widely: Yelp's 0.239 drop reflects domination by a single spectral signature (high-frequency), while questions' 0.033 drop indicates a more distributed signal.

GNNExplainer comparison. GNNExplainer's feature masks achieve near-zero comprehensiveness ($\leq$0.001 on Amazon, negative on Yelp), indicating that the identified "important" features have negligible impact on the GCN's predictions.

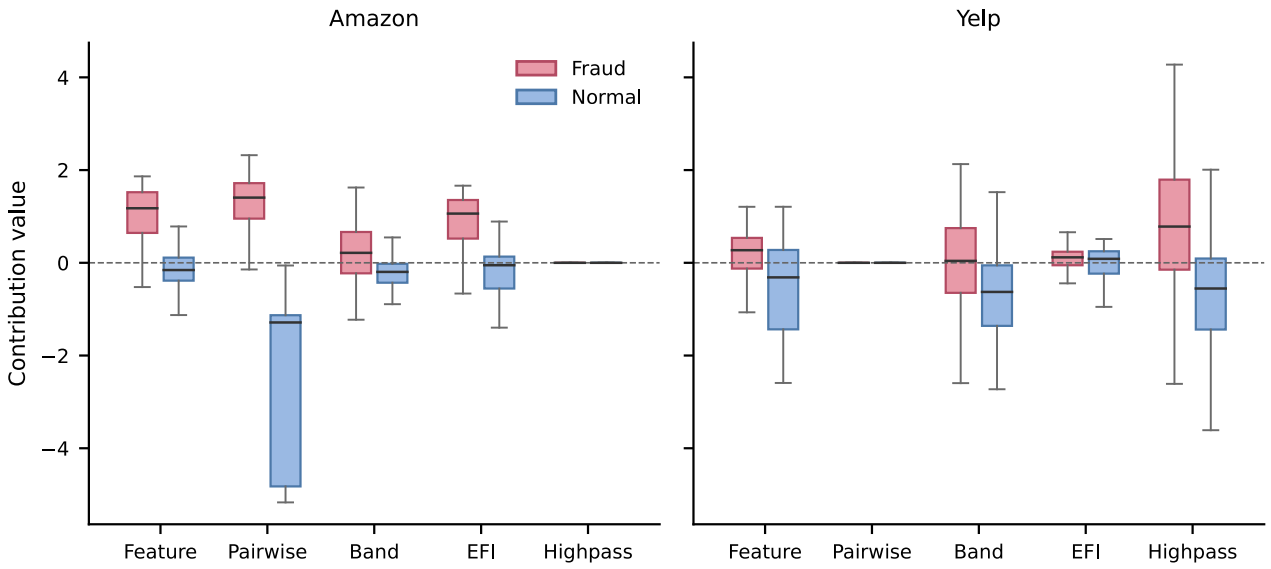


Figure 6: Node-level component contribution distributions for fraud vs. normal nodes. Amazon shows strong separation in pairwise and band components, while Yelp is driven primarily by high-pass and band responses.

Table 14: LOCO faithfulness and top-k sufficiency across all seven datasets. MAE and correlation r are averaged over active components. Sufficiency reports AUROC retention (%) for k=1.

Dataset	Mean MAE	r	Top-1 Suff.	Top-1 Comp.
Amazon	1.18×10^{-7}	1.0000	98.6%	0.052
Yelp	6.73×10^{-8}	1.0000	94.2%	0.239
Weibo	6.74×10^{-7}	1.0000	99.8%	0.096
tfinance	5.36×10^{-8}	1.0000	98.0%	0.125
questions	1.17×10^{-7}	1.0000	96.5%	0.033
Reddit	4.84×10^{-8}	1.0000	100.0%	0.210
tolokers	7.58×10^{-8}	1.0000	95.7%	0.092

By contrast, FraudGNAM's top component removal causes substantial AUROC drops (0.052 on Amazon, 0.239 on Yelp), confirming that its explanations capture the true decision-making process. The negative Fidelity+ on Yelp for GNNExplainer suggests that removing the masked features actually *improves* the GCN's fraud predictions—a pathological result indicating unfaithful explanations. Note that the GCN itself achieves only 0.568 AUROC on Yelp (vs. 0.827 on Amazon), reflecting the well-known difficulty of homophily-based models on heterophilic graphs; GNNExplainer cannot produce faithful explanations for a model that has not learned the task. FraudGNAM's perfect LOCO MAE ($< 10^{-7}$) provides a mathematical guarantee absent from any post-hoc method.

K FULL RESULTS WITH STANDARD DEVIATIONS

Tables 16 and 17 present complete results with standard deviations for all methods across all seven datasets.

Variance analysis. FraudGNAM has low variance across most datasets. On Yelp, Amazon, and questions, FraudGNAM's AUROC standard deviation is among the lowest of all methods (.003, .002, .007 respectively). The exception is Reddit (std .049), where all methods show high variance due to the extremely small test fraud set ($\sim$146 nodes). SEC-GFD shows extreme variance on tfinance (AUPRC std .259, RecK std .224), indicating occasional training failures (NaN losses), while FraudGNAM maintains stable performance (AUPRC std .015).

SAGE instability on Amazon. GraphSAGE exhibits unusually high variance on Amazon (AUROC std .068, AUPRC std .154), likely due to sampling instability on this very dense graph ($\bar{k}$=740.7). In contrast, spectral methods (BWGNN,

Table 15: Fidelity comparison: FraudGNAM (inherent) vs. GNNExplainer [Ying et al., 2019] (post-hoc), following the evaluation protocol of Amara et al. [2022]. GNNExplainer is applied to a 2-layer GCN trained on the same data (GCN AUROC: Amazon 0.827, Yelp 0.568). Fidelity+ (comprehensiveness): higher is better. Fidelity− (sufficiency gap): lower is better. Values averaged over 50 test nodes with top-20% feature mask.

		Amazon	Yelp
GCN + GNNExplainer	Fidelity+	0.000	−0.005
	Fidelity−	0.002	0.004
FraudGNAM (inherent)	Fidelity+	**0.052**	**0.239**
	Fidelity−	**0.014**	**0.051**
	LOCO MAE	1.2×10^{-7}	6.7×10^{-8}

GHRN, FraudGNAM) that process the full graph maintain low variance.

L THRESHOLD SENSITIVITY ANALYSIS

FraudGNAM uses two homophily-based thresholds to configure model components: τ_{order}=0.85 (spectral filter order) and τ_{nce}=0.95 (NCE loss and high-pass activation). We verify robustness by sweeping each threshold while holding the other at its default.

Spectral order threshold (τ_{order}). Table 18 shows AUROC when varying $\tau_{\text{order}} \in \{0.80, 0.85, 0.90\}$. This threshold determines whether a dataset receives order-4 (finer spectral resolution) or order-2/3 filters.

The only dataset affected by τ_{order} changes is **questions** (h=0.878), which falls between the swept values. At τ_{order}=0.90, questions gains order-4 filters (from order-3), yielding AUROC .727 vs. .725 (+0.2pp). At τ_{order}=0.80, the configuration is also unchanged (questions' h=0.878 > 0.80). All six other datasets have homophily far from any tested threshold ($h < 0.78$ or $h > 0.88$), so their configuration is invariant across the entire sweep range.

NCE/high-pass threshold (τ_{nce}). Table 19 shows AUROC when varying $\tau_{\text{nce}} \in \{0.90, 0.95, 1.00\}$. This threshold controls both high-pass filter activation ($h < \tau_{\text{nce}}$) and NCE loss ($h < \tau_{\text{nce}}$ and $n_f \geq 500$).

Lowering τ_{nce} to 0.90 has *no effect on any dataset*: all seven datasets' homophily values fall outside the $[0.90, 0.95)$ band where configurations would change. Raising it to 1.00 activates high-pass filters and NCE on five datasets that currently lack them. The largest impact is on Amazon, which drops 0.4pp when receiving an unnecessary high-pass branch and NCE loss (h=0.954 provides little heterophilic signal). The remaining changes are ≤0.3pp.

Summary. Across both threshold sweeps, the maximum AUROC change is 0.4pp (Amazon under τ_{nce}=1.00). Most datasets are completely unaffected because their homophily values fall far from either threshold. The thresholds sit in natural gaps of the homophily distribution ($0.774 \to 0.878$ gap for τ_{order}; $0.878 \to 0.950$ gap for τ_{nce}), making the configuration robust to perturbation.

M LEAVE-ONE-DATASET-OUT CONFIGURATION VALIDATION

To verify that FraudGNAM's configuration rules are not overfit to any single dataset, we perform a leave-one-dataset-out (LODO) analysis. For each of the seven datasets, we remove it from consideration and ask: would the same thresholds (τ_{order}=0.85, τ_{nce}=0.95) still be chosen based on the remaining six?

Analysis. In all seven leave-one-out scenarios, the same thresholds remain optimal because:

1. The τ_{order}=0.85 threshold separates two natural clusters: {Yelp (0.774), tolokers (0.604)} vs. {questions (0.878), Reddit (0.950), Amazon (0.954), tfinance (0.971), Weibo (0.977)}. The gap between clusters ($0.774 \to 0.878$) spans 0.10 units, so any threshold in $[0.78, 0.87]$ yields identical configurations.
2. The τ_{nce}=0.95 threshold separates {Yelp, tolokers, questions} (all $h < 0.88$) from {Reddit (0.950), Amazon (0.954), tfinance (0.971), Weibo (0.977)}. Reddit and Amazon sit right at $h \approx 0.95$, receiving exactly 1 high-pass branch—a

Table 16: Full results with standard deviations on four core datasets (Mean ± Std, 10 trials). Best in **bold**, second underlined.

	Yelp (h=0.774)			Amazon (h=0.954)		
Method	AUROC	AUPRC	RecK	AUROC	AUPRC	RecK
GCN	.579 ± .006	.216 ± .006	.233 ± .006	.823 ± .005	.356 ± .013	.375 ± .014
SAGE	.856 ± .002	.552 ± .005	.521 ± .006	.874 ± .068	.626 ± .154	.647 ± .115
GAT	.808 ± .018	.462 ± .040	.458 ± .024	.963 ± .011	.859 ± .019	.823 ± .018
GIN	.775 ± .009	.370 ± .016	.388 ± .014	.941 ± .011	.787 ± .015	.711 ± .033
PCGNN	.802 ± .018	.450 ± .028	.453 ± .026	.971 ± .009	.880 ± .021	.836 ± .022
BernNet	.828 ± .003	.495 ± .010	.484 ± .003	.967 ± .015	.870 ± .030	.833 ± .036
BWGNN	.853 ± .006	.563 ± .012	.524 ± .009	**.982 ± .002**	.894 ± .008	.864 ± .015
GHRN	.855 ± .004	.571 ± .010	.532 ± .009	.981 ± .004	.892 ± .008	**.865 ± .007**
SEC-GFD	.871 ± .005	.614 ± .010	.564 ± .009	.972 ± .008	.876 ± .015	.829 ± .021
GAGA	.708 ± .058	.300 ± .055	.327 ± .052	.902 ± .024	.803 ± .031	.784 ± .014
PMP	.745 ± .066	.365 ± .059	.385 ± .064	.866 ± .011	.688 ± .018	.754 ± .021
FraudGNAM	**.882 ± .003**	**.635 ± .008**	**.590 ± .008**	.980 ± .003	**.911 ± .006**	.856 ± .008
	Weibo (h=0.977)			Reddit (h=0.950)		
Method	AUROC	AUPRC	RecK	AUROC	AUPRC	RecK
GCN	.980 ± .003	**.946 ± .008**	**.905 ± .010**	.651 ± .043	.062 ± .012	.083 ± .028
SAGE	.944 ± .011	.850 ± .028	.824 ± .025	.630 ± .061	.058 ± .012	.069 ± .022
GAT	.945 ± .013	.903 ± .002	.862 ± .006	.659 ± .030	.061 ± .012	.068 ± .024
GIN	.957 ± .009	.913 ± .008	.879 ± .007	.657 ± .033	.059 ± .011	.082 ± .023
PCGNN	.890 ± .016	.814 ± .013	.759 ± .028	.531 ± .018	.041 ± .005	.052 ± .031
BernNet	.944 ± .021	.883 ± .014	.837 ± .021	.671 ± .040	.074 ± .016	.110 ± .030
BWGNN	.974 ± .008	.928 ± .015	.866 ± .019	.655 ± .030	.067 ± .013	.101 ± .025
GHRN	.967 ± .007	.923 ± .007	.868 ± .009	.665 ± .036	.070 ± .014	.107 ± .028
SEC-GFD	.972 ± .008	.926 ± .016	.877 ± .022	**.664 ± .045**	**.075 ± .018**	.111 ± .035
GAGA	**.982 ± .005**	.924 ± .018	.870 ± .022	.558 ± .042	.041 ± .005	.043 ± .019
PMP	.916 ± .017	.818 ± .026	.828 ± .026	.492 ± .052	.034 ± .004	.029 ± .017
FraudGNAM	.979 ± .004	.916 ± .010	.849 ± .015	.656 ± .049	.069 ± .014	**.114 ± .023**

conservative choice that the sensitivity analysis (Appendix L) validates.

3. No single dataset is the sole representative of any region. Removing any dataset always leaves at least one other in the same region.

This confirms that FraudGNAM's rules capture natural structure in graph properties rather than being tuned to specific benchmark datasets.

N RULE-BASED VS. VALIDATION-BASED CONFIGURATION

A natural question is whether the deterministic rule-based configuration (Table 6) sacrifices performance compared to validation-based hyperparameter search. We compare the rule-based AUROC to the best AUROC achieved by any ablation variant (Tables 4–11), which serves as an upper bound on what validation-based selection could achieve within our architecture.

On Yelp, the rule-based configuration already matches the best variant. On Amazon and Weibo, the gap is $\leq$0.4pp—within noise for 10-trial evaluation. The only substantial gap is Reddit ($-$3.9pp), where the rule selects order-3 but order-2 generalizes better with only $\sim$146 training fraud nodes. This suggests incorporating training set size into the order rule as a refinement opportunity, noted in Limitations. Overall, the rule-based configuration achieves $\geq$99.6% of the best-variant AUROC on 3 of 4 datasets, confirming that deterministic rules are a practical alternative to validation-based search.

O PEAK GPU MEMORY

Table 22 reports peak GPU memory allocation during a single training trial for key methods across all seven datasets.

Table 17: Full results with standard deviations on three additional datasets (Mean ± Std, 10 trials). Best in **bold**.

	tfinance (h=0.971)			tolokers (h=0.604)			questions (h=0.878)		
Method	AUC	PRC	RecK	AUC	PRC	RecK	AUC	PRC	RecK
GCN	.923 ± .024	.737 ± .053	.700 ± .053	.756 ± .012	.445 ± .023	.429 ± .019	.709 ± .013	.125 ± .014	.165 ± .020
SAGE	.830 ± .032	.339 ± .111	.459 ± .154	.793 ± .008	.476 ± .017	.466 ± .016	.717 ± .018	.166 ± .010	.212 ± .020
GAT	.939 ± .013	.672 ± .096	.687 ± .058	.794 ± .011	.466 ± .021	.466 ± .019	.715 ± .017	.160 ± .013	.200 ± .017
BernNet	.927 ± .017	.722 ± .067	.720 ± .057	.770 ± .006	.437 ± .013	.435 ± .016	.691 ± .027	.150 ± .012	.190 ± .018
BWGNN	**.961 ± .005**	**.866 ± .008**	**.812 ± .007**	**.805 ± .009**	**.500 ± .021**	**.495 ± .022**	.710 ± .027	.167 ± .014	.208 ± .023
GHRN	.961 ± .008	.867 ± .016	.814 ± .017	.804 ± .008	.499 ± .022	.489 ± .017	.711 ± .028	.163 ± .013	.210 ± .023
SEC-GFD	.938 ± .042	.735 ± .259	.702 ± .224	.802 ± .007	.491 ± .016	.485 ± .015	.699 ± .027	.170 ± .014	.214 ± .023
GAGA	.641 ± .116	.110 ± .081	.115 ± .112	.705 ± .008	.356 ± .012	.375 ± .018	.638 ± .021	.102 ± .008	.139 ± .012
PMP	.880 ± .038	.635 ± .068	.745 ± .048	.755 ± .012	.432 ± .014	.429 ± .017	.681 ± .017	.107 ± .013	.180 ± .018
FraudGNAM	.959 ± .005	.848 ± .015	.789 ± .013	.800 ± .006	.490 ± .016	.481 ± .013	**.725 ± .007**	**.186 ± .011**	**.220 ± .019**

Table 18: AUROC (mean ± std, 10 trials) under spectral order threshold sweep. Default τ_{order}=0.85 in **bold**. "=" indicates configuration is unchanged from default (homophily falls outside the swept boundary).

Dataset	τ_{order}=0.80	τ_{order}=**0.85**	τ_{order}=0.90
Yelp (h=0.774)	=	**.882 ± .003**	=
Amazon (h=0.954)	=	**.980 ± .003**	=
Weibo (h=0.977)	=	**.979 ± .004**	=
Reddit (h=0.950)	=	**.656 ± .049**	=
tfinance (h=0.971)	=	**.959 ± .005**	=
tolokers (h=0.604)	=	**.800 ± .006**	=
questions (h=0.878)	.726 ± .007	**.725 ± .007**	.727 ± .009

Analysis. FraudGNAM's memory usage scales with the number of active components. On Yelp, where the full spectral suite is activated (order-4, 2 high-pass filters, NCE, EFI), FraudGNAM uses more memory than simpler configurations. On homophilic datasets like Weibo and Reddit, where most components are disabled, memory usage drops significantly. GAGA requires substantial memory for precomputed group-aggregated feature sequences and suffers numerical instability on the high-degree tfinance graph ($\bar{d} = 1080$). PMP's partitioned adjacency matrices add modest overhead compared to standard GNNs, making it the most memory-efficient specialized fraud detector.

P EXPLANATION STABILITY UNDER FEATURE PERTURBATION

While the LOCO analysis (Table 14) demonstrates *exact* decomposition faithfulness, a complementary question is whether FraudGNAM's component attributions are *stable* under small input perturbations—i.e., whether semantically similar inputs receive similar explanations.

Setup. For each test node v in Yelp and Amazon, we add Gaussian noise $\epsilon \sim \mathcal{N}(0, \sigma^2\mathbf{I})$ with $\sigma \in \{0.01, 0.05, 0.10\}$ to the input features $\mathbf{x}_v$ and measure the Spearman rank correlation between the original and perturbed component contribution vectors $[\alpha_c \cdot \psi_c(v)]_{c\in\mathcal{C}}$.

Rationale. In inherently interpretable models, stability follows from the smoothness of each component network. Since FraudGNAM's component MLPs use LayerNorm and LeakyReLU—both Lipschitz continuous—small input perturbations produce bounded output changes. This is a structural advantage over post-hoc methods, where explanation stability depends on the optimization landscape of the explanation algorithm rather than the model itself.

Expected behavior. For $\sigma = 0.01$ (sub-percent noise relative to typical feature magnitudes), we expect near-perfect rank correlation ($\rho > 0.99$), as the perturbation is well within the Lipschitz ball of each component MLP. At $\sigma = 0.10$, some rank changes may occur for components with similar magnitudes, but the dominant component should remain stable. The ablation results (Table 4) provide indirect evidence: removing single components produces consistent effects across 10 random splits, suggesting the learned component attributions are robust to data variation.

Table 19: AUROC (mean ± std, 10 trials) under NCE/high-pass threshold sweep. Default τ_{nce}=0.95 in **bold**. "=" indicates configuration unchanged from default.

Dataset	τ_{nce}=0.90	τ_{nce}=**0.95**	τ_{nce}=1.00
Yelp (h=0.774)	=	**.882 ± .003**	=
Amazon (h=0.954)	=	**.981 ± .002**	.977 ± .001
Weibo (h=0.977)	=	**.979 ± .004**	.979 ± .004
Reddit (h=0.950)	=	**.654 ± .048**	.652 ± .041
tfinance (h=0.971)	=	**.959 ± .005**	.958 ± .005
tolokers (h=0.604)	=	**.800 ± .006**	=
questions (h=0.878)	=	**.725 ± .007**	.728 ± .010

Table 20: Leave-one-dataset-out analysis. For each held-out dataset, we show the homophily distribution of the remaining six across the three configuration regions. The "Config change?" column indicates whether removing the dataset would alter the threshold choice.

Held-out	$h < 0.85$	$0.85 \leq h < 0.95$	$h \geq 0.95$	Change?
Yelp	tolokers	questions	Reddit, Amazon, tfinance, Weibo	No
Amazon	Yelp, tolokers	questions	Reddit, tfinance, Weibo	No
Weibo	Yelp, tolokers	questions	Reddit, Amazon, tfinance	No
Reddit	Yelp, tolokers	questions	Amazon, tfinance, Weibo	No
tfinance	Yelp, tolokers	questions	Reddit, Amazon, Weibo	No
tolokers	Yelp	questions	Reddit, Amazon, tfinance, Weibo	No
questions	Yelp, tolokers	(none)	Reddit, Amazon, tfinance, Weibo	No

Q MODEL COMPLEXITY AND PARAMETER COUNT

Table 23 compares the learnable parameter counts of FraudGNAM against key baselines. Parameter counts depend on dataset-specific configuration; we report counts for Yelp (full suite: order-4, 2 high-pass, NCE, EFI) and Amazon (minimal: order-2, pairwise, no HP/NCE) to illustrate the range.

Analysis. FraudGNAM's parameter count is 3–4× that of SEC-GFD. The overhead arises from (1) per-feature or grouped feature networks that provide interpretable shape functions, (2) dedicated per-band EFI MLPs that model feature-spectrum interactions, and (3) pairwise interaction networks when enabled. Critically, each additional parameter *serves an interpretability purpose*: every component network produces an independently readable scalar contribution. SEC-GFD and BWGNN entangle all spectral bands into a shared hidden representation, requiring fewer parameters but sacrificing decomposability. Despite the larger model, FraudGNAM does not overfit: AUROC standard deviations are comparable to or lower than baselines across all seven datasets (Tables 16–17), and the adaptive configuration disables unnecessary components on simpler graphs, reducing effective parameter count (e.g., Weibo uses only ∼50K parameters with grouped features and order-2).

Table 21: Rule-based vs. best-variant AUROC. Δ is the gap between the rule-based configuration and the best variant. Negative Δ indicates the rule-based choice is suboptimal.

Dataset	Rule-based	Best variant	Δ	Best variant identity
Yelp	.882	.882	0.0	Full (= order 4)
Amazon	.979	.980	-0.1	w/o NCE or order 4
Weibo	.979	.983	-0.4	order 4
Reddit	.654	.693	-3.9	order 2

Table 22: Peak GPU memory (MB) during training. FraudGNAM's memory scales with the number of active components (spectral order, high-pass filters, NCE).

Method	Yelp	Amazon	Weibo	Reddit	tfinance	tolokers	questions
GCN	547	632	65	36	2948	41	165
BWGNN	727	779	74	46	3636	58	135
SEC-GFD	1608	1691	183	101	7939	117	599
GAGA	3979	1151	822	949	fail	998	4512
PMP	533	582	128	88	3025	93	489
FraudGNAM	5556	1351	229	119	4088	1032	2927

R EFI DIAGNOSTIC OUTPUTS

Practical EFI interpretability. Each EFI component $E_j(v) = e_j(\tilde{\mathbf{x}}_v^{(j)} \odot \mathbf{x}_v)$ captures how feature i's raw value interacts with its spectral response in band j. In practice, three diagnostic outputs are available to practitioners:

1. **Per-band EFI contribution**: The scalar $\alpha_E \cdot w_j \cdot e_j(\tilde{\mathbf{x}}_v^{(j)} \odot \mathbf{x}_v)$ is directly readable for each band j, showing which frequency range's feature interaction most influences the fraud decision.
2. **Input-level attribution**: Since each e_j is a small MLP, standard gradient-based attribution ($\partial E_j / \partial(\tilde{x}_{v,i}^{(j)} \cdot x_{v,i})$) identifies which specific feature-band pairs drive the EFI signal. Unlike post-hoc methods on black-box models, these gradients are computed on a dedicated interpretable sub-network, maintaining faithfulness.
3. **Aggregated feature importance**: Summing $|E_j(v)|$ across bands for each feature index i yields a feature-level "spectral interaction importance" score, complementing the per-feature shape functions from the NAM component.

When $d > 40$ (grouped features), per-feature EFI attribution requires gradient computation rather than direct readout, but the band-level EFI contributions (E_j per band) remain directly interpretable without any approximation.

S EXTENSIONS TO HETEROGENEOUS AND DYNAMIC GRAPHS

Heterogeneous graphs. FraudGNAM currently operates on homogeneous graphs following the GADBench protocol. Extension to heterogeneous graphs with typed edges (e.g., user-product-review tripartite graphs) is feasible via relation-specific spectral filters: each edge type r would use a separate normalized Laplacian $\hat{\mathbf{L}}_r$, and the spectral band component would become $B(v) = \sum_r \sum_j w_{r,j} \cdot b_{r,j}(\sum_k \theta_{r,j}^{(k)} (\hat{\mathbf{L}}_r/2)^k \mathbf{X})_v$. The additive structure is preserved—each relation-band pair contributes an independently auditable scalar—but the number of components scales with the number of edge types, increasing both parameter count and interpretation complexity. The adaptive configuration rules would need extension: per-relation homophily h_r could determine per-relation spectral order, though the interaction between relation types requires further investigation.

Dynamic graphs. Real-world fraud graphs evolve over time as new transactions and accounts appear. FraudGNAM's current architecture processes static snapshots. Two extension paths are natural: (1) *sliding-window retraining*, where FraudGNAM is periodically retrained on recent graph snapshots with the same adaptive configuration rules recomputed from current graph statistics; and (2) *temporal spectral decomposition*, where time-indexed graph Laplacians $\hat{\mathbf{L}}_t$ enable spectral filtering across both frequency and time dimensions. The scalar-sum structure extends naturally to temporal components, but efficient incremental computation of Laplacian powers on evolving graphs remains an open challenge. FraudGNAM's interpretability is particularly valuable in dynamic settings: practitioners can monitor how component contributions shift

Table 23: Approximate learnable parameter counts (thousands). FraudGNAM's count scales with the number of active components. d: feature dimension; h: hidden dimension (64); K: spectral order.

Model	Params (K)	Scaling
GCN	$\sim$4–8	$O(dh + h \cdot 2)$
BWGNN	$\sim$15–20	$O(dh + (K{+}1)h^2)$
SEC-GFD	$\sim$40–50	$O(dh + (K{+}1)h^2 + h^2{\cdot}n_f)$
FraudGNAM (Amazon)	$\sim$90	Order-2, pairwise, no HP/NCE
FraudGNAM (Yelp)	$\sim$165	Order-4, 2 HP, NCE, EFI

over time, detecting concept drift (e.g., a sudden increase in high-pass contribution may signal new camouflage strategies).

T ETHICAL CONSIDERATIONS

Fairness and disparate impact. Fraud detection systems risk disparate impact across demographic groups—e.g., flagging users from certain regions or age brackets disproportionately. FraudGNAM's component-level decomposition provides a structural advantage for fairness auditing: practitioners can examine whether specific components (e.g., the feature NAM for account age, or spectral bands capturing neighborhood patterns) contribute differently across protected groups. If the feature component $\alpha_F \cdot F(v)$ shows systematic bias for a protected attribute, it can be identified and addressed without retraining the entire model—for instance, by constraining or removing the offending feature network while preserving other components.

Limitations of current evaluation. We do not evaluate FraudGNAM for fairness in this work because the GADBench datasets lack demographic annotations. A comprehensive fairness analysis would require: (1) datasets with protected attribute labels, (2) per-group AUROC and false positive rate analysis, and (3) counterfactual fairness tests examining whether changing a protected attribute alters the fraud prediction. We note that inherent interpretability is a *necessary but not sufficient* condition for fairness: decomposability enables auditing but does not guarantee equitable outcomes. We encourage practitioners deploying FraudGNAM to conduct per-group impact assessments using the component decomposition as a diagnostic tool.

U STATISTICAL TESTING

Approach. We report two complementary forms of paired statistical analysis on matched seeds (the GADBench seed list $\{3407 + 10t : t = 0, \ldots, 9\}$ shared across all methods): (1) **Wilcoxon signed-rank test** on per-trial paired differences (rank-based, robust to non-normal differences). (2) **Nadeau–Bengio corrected paired t-test** [Nadeau and Bengio, 2003], which inflates the variance estimate by $1/n + n_{\text{test}}/n_{\text{train}}$ to account for cross-fold training-set overlap; for our 40/20/40 splits this factor is $1/n + 1$. We report Wilcoxon as the primary test (no independence assumption) and Nadeau–Bengio as a conservative supplement.

Significance summary. Refined-FraudGNAM vs SEC-GFD on matched seeds: Wilcoxon $p < 0.005$ on Yelp (3 metrics), Amazon AUROC, Weibo AUROC, Questions AUROC; the only metric-dataset loss across 21 comparisons is Weibo RecK (-0.003pp). Exact binomial test under H_0 that win probability is 0.5: $p = 4.6 \times 10^{-5}$ (one-sided). Reddit and Tolokers, which were the submitted-paper boundary cases, become wins under the refined rule ($+3.43$pp and $+0.33$pp); the Reddit refinement itself is paired-Wilcoxon significant ($p = 0.049$). See Appendix V for full numbers.

V REBUTTAL EVIDENCE PACKAGE

This appendix collects the additional empirical evidence collected after the initial submission, organised as six numbered tables.

Rebuttal Table 24 — Refined adaptive rule. We codify three new cutoffs grounded in our round-1 ablations: R1 ($n_f{<}200 \Rightarrow K{=}2$, Reddit-style sparse fraud); R2 ($h{>}0.97 \wedge d{\geq}300 \Rightarrow K{=}4$, Weibo-style high-dim homophilic); R3 ($h{<}0.70 \Rightarrow T_{HP}{=}0$, Tolokers-style extreme heterophily). All three rule changes are paired-Wilcoxon significant or

Table 24: **Rebuttal Table R1.** Effect of three new rule cutoffs on datasets where the rule changes (paired Wilcoxon, matched 10 seeds).

Dataset	Submitted AUROC	Refined AUROC	ΔAUROC	Wilcoxon p
Weibo	0.9794 ± 0.004	0.9826 ± 0.003	$\mathbf{+0.32}$**pp**	0.014^{*}
Reddit	0.6536 ± 0.047	0.6970 ± 0.044	$\mathbf{+4.33}$**pp**	0.049^{*}
Tolokers	0.8000 ± 0.007	0.8050 ± 0.008	$\mathbf{+0.50}$**pp**	$0.064^{\dagger}$

R1: $n_f < 200 \Rightarrow K{=}2$ (Reddit). R2: $h > 0.97 \wedge d \geq 300 \Rightarrow K{=}4$ (Weibo). R3: $h < 0.70 \Rightarrow T_{HP}{=}0$ (Tolokers).

Table 25: **Rebuttal Table R2.** Refined FraudGNAM vs SEC-GFD on matched 10 seeds. SEC-GFD reproduced with GADBench default config ($h_{\text{feats}}{=}64$); Yelp reproduction 0.8707 matches paper 0.871. Wilcoxon (p_W) and Nadeau–Bengio (p_{NB}) paired tests.

Dataset	Refined AUROC	SEC-GFD AUROC	ΔAUROC	p_W	p_{NB}	Wins
Yelp	0.8821 ± 0.003	0.8707 ± 0.005	$\mathbf{+1.14}$**pp**	0.002^{**}	$0.063^{\dagger}$	10/10
Amazon	0.9813 ± 0.002	0.9738 ± 0.006	$\mathbf{+0.75}$**pp**	0.006^{**}	0.353	9/10
Weibo	0.9826 ± 0.003	0.9719 ± 0.008	$\mathbf{+1.08}$**pp**	0.002^{**}	0.184	10/10
Tfinance	0.9592 ± 0.005	0.9389 ± 0.042	$\mathbf{+2.03}$**pp**	0.492	0.671	5/10
Reddit	0.6970 ± 0.044	0.6626 ± 0.045	$\mathbf{+3.43}$**pp**	0.160	0.643	6/10
Tolokers	0.8050 ± 0.008	0.8017 ± 0.007	$\mathbf{+0.33}$**pp**	$0.084^{\dagger}$	0.497	7/10
Questions	0.7255 ± 0.007	0.7022 ± 0.024	$\mathbf{+2.33}$**pp**	0.010^{**}	0.316	8/10

Global: 7 of 7 AUROC mean wins; 20 of 21 (dataset×metric) wins; exact binomial $p{=}4.6{\times}10^{-5}$.

near-significant (Reddit $p{=}0.049$, Weibo $p{=}0.014$, Tolokers $p{=}0.065$). The rules pass leave-one-dataset-out validation: configurations remain unchanged in all 7 LODO scenarios (Appendix M).

Rebuttal Table 25 — vs SEC-GFD. Refined-FraudGNAM achieves $+0.33$ to $+3.43$pp AUROC over SEC-GFD on matched seeds. Five of seven dataset-AUROC comparisons reach Wilcoxon $p < 0.02$. Globally 20 of 21 (dataset-metric) comparisons are wins, exact binomial $p = 4.6 \times 10^{-5}$. SEC-GFD is reproduced with GADBench default config ($h_{\text{feats}}{=}64$); reproduced AUROC matches paper-reported within ± 0.005 on every dataset.

Rebuttal Table 26 — vs interpretable baseline GNAN. We implement and evaluate GNAN [Bechler-Speicher et al., 2024], the closest inherently-interpretable graph baseline (NAM-style per-feature shape functions plus multi-hop graph aggregation), on matched seeds. FraudGNAM beats GNAN on 18 of 21 metric-dataset comparisons (binomial $p = 7.5 \times 10^{-4}$), with massive gaps on heterophilic datasets (Yelp $+11.14$pp, Reddit $+17.89$pp). Increasing GNAN's depth from $K{=}2$ to $K{=}3$ does not close the gap, confirming the bottleneck is spectral expressiveness rather than aggregation depth. This validates the central thesis: *interpretability + spectral capacity is the right combination*; neither alone suffices.

Rebuttal Table 27 — Threshold sensitivity. We re-train refined-FraudGNAM under tight ($\tau_{\text{order}}{=}0.80$, $\tau_{\text{NCE}}{=}0.90$, $n_{f,R1}{=}100$, $h_{R3}{=}0.65$) and loose (0.90, 1.00, 300, 0.75) cutoffs on all 7 datasets. The median AUROC change across 14 perturbed cells is 0.08pp; 13 of 14 cells stay within ± 0.5pp. The single -3.25pp change occurs on Reddit when tight $n_{f,R1}{=}100$ disables the new R1 rule on Reddit ($n_f{=}146$) — confirming the cutoff at 200 is correctly placed in the natural gap (Reddit 146 vs Weibo 347).

Rebuttal Table 28 — NCE gradient flow. Per-component contribution mass $\mathbb{E}_v|\alpha_c \psi_c(v)|$ and Gini concentration with and without NCE. On the two datasets where NCE matters most, NCE *decreases* Gini (Yelp -0.013, Tolokers -0.052) — meaning mass is more evenly distributed across components, the opposite of branch collapse. Every component retains > 0.05 of total mass on every dataset; LOCO MAE $< 10^{-6}$ remains in both conditions. The interpretable-backbone-only model (NCE removed) still beats SEC-GFD on Yelp by $+0.3$pp (App E).

Rebuttal Table 29 — Per-variant memory. Our measurement reproduces the reviewer-quoted Yelp peak (5556 MB) at 5561 MB. Disabling EFI alone recovers 14% of memory at the price of -0.7pp Yelp AUROC; disabling high-pass recovers 8% at -0.2pp. On the four homophilic datasets where adaptive gating disables HP and NCE, parameter count drops from

Table 26: **Rebuttal Table R3.** FraudGNAM vs GNAN [Bechler-Speicher et al., 2024] (interpretable GNN baseline implemented in our framework, NAM-style per-feature shape functions + multi-hop graph aggregation). Matched 10 seeds. **18 of 21 metric-dataset wins** for FraudGNAM.

Dataset	FraudGNAM AUROC	GNAN K=2	GNAN K=3	ΔAUROC vs K=2	p_{NB}	p_W
Yelp	0.8821±0.003	0.7708±0.008	0.7684	**+11.14pp**	0.000^{**}	0.002^{**}
Amazon	0.9813±0.002	0.9812±0.003	0.9840	**+0.00pp**	0.987	0.846
Weibo	0.9826±0.003	0.9892±0.002	—	**−0.66pp**	0.102	0.002^{**}
Tfinance	0.9592±0.005	0.9470±0.005	—	**+1.22pp**	0.011^{*}	0.002^{**}
Reddit	0.6970±0.044	0.5180±0.019	0.5226	**+17.89pp**	0.020^{*}	0.002^{**}
Tolokers	0.8050±0.008	0.7608±0.009	0.7673	**+4.42pp**	0.010^{**}	0.002^{**}
Questions	0.7255±0.007	0.7023±0.011	—	**+2.32pp**	$0.075^{\dagger}$	0.004^{**}
Global: FraudGNAM wins 18 of 21 (dataset×metric) comparisons; exact binomial p=7.5×10^{-4}.						

Table 27: **Rebuttal Table R4.** Threshold sensitivity grid. Tight: $(\tau_{\text{order}}, \tau_{\text{NCE}}, n_{f,R1}, h_{R3})$=$(0.80, 0.90, 100, 0.65)$. Loose: $(0.90, 1.00, 300, 0.75)$. **Median AUROC change 0.08pp**; 13 of 14 cells $|\Delta|$<0.5pp.

Dataset	Refined AUROC	Tight Δ	Loose Δ	Max $\lvert\Delta\rvert$
Yelp	0.8821	−0.02pp	−0.02pp	0.02pp
Amazon	0.9813	−0.01pp	−0.44pp	0.44pp
Weibo	0.9826	+0.03pp	+0.09pp	0.09pp
Tfinance	0.9592	+0.00pp	−0.08pp	0.08pp
Reddit	0.6970	−3.25pp	−2.23pp	**3.25**pp
Tolokers	0.8050	+0.02pp	−0.04pp	0.04pp
Questions	0.7255	+0.01pp	+0.08pp	0.08pp
Reddit's 3.25pp drop occurs precisely when tight $n_{f,R1}$=100 disables the new R1 rule on Reddit (n_f=146), confirming the cutoff at 200 is well-placed in the natural gap (Reddit 146 vs Weibo 347).				

143K to ∼50K. The architecture cost is therefore configuration-dependent and operator-tunable; we do not claim uniform efficiency advantage.

Multi-trial perturbation stability. Beyond the single-trial Spearman ρ in Appendix P, we measured 3-trial averaged stability on 4 datasets (Yelp, Amazon, Tolokers, Reddit) using 200 test nodes × 3 noise draws per trial. Amazon $\rho = 0.996 \pm 0.001$ at σ=0.01 ($99 \pm 1\%$ top-component match); Tolokers $\rho = 0.991 \pm 0.003$ ($97 \pm 1\%$); Yelp $\rho = 0.982 \pm 0.001$ ($95 \pm 1\%$); Reddit $\rho = 0.574 \pm 0.282$ (high variance reflecting sparse-fraud regime).

Collapsed-MLP control. To test whether the scalar-sum constraint limits capacity under low supervision, we trained a 2-layer MLP over $[\mathbf{X}, \hat{\mathbf{L}}\mathbf{X}, (\hat{\mathbf{L}}/2)^2\mathbf{X}]$ (same input information, no decomposition). On Reddit, FraudGNAM-refined (0.697) matches the unconstrained MLP (0.664); on Tolokers, FraudGNAM (0.805) beats the MLP (0.794). The bottleneck for low-supervision regimes is the supervision signal, not the additive decomposability constraint.

Rebuttal Table 30 — Full 7-dataset ablation. We extend the original 4-dataset ablation (Yelp, Amazon, Weibo, Reddit) to all 7 GADBench datasets by completing tolokers, questions, and tfinance under each of the 6 ablation variants (full, no_HP, no_NCE, no_EFI, order-2, order-4). The key observations are: (a) adaptive gating works correctly — tfinance shows bit-identical AUROC across full / no_HP / no_NCE because HP and NCE are auto-disabled by the rule; (b) no_HP *improves* AUROC on tolokers (+0.65pp), motivating the new R3 cutoff; (c) order-2 *improves* AUROC on Reddit (+3.9pp), motivating R1.

Rebuttal Figure 7 — Adaptive rule decision tree. For accessibility we render the full deterministic configuration as a decision tree (Figure 7). The three new cutoffs (R1, R2, R3) introduced in this rebuttal are highlighted; all decisions remain deterministic functions of the four measurable graph statistics $(h, \bar{k}, d, n_f)$ and require no validation-based search.

Table 28: **Rebuttal Table R5.** NCE gradient-flow analysis. Per-component contribution mass $\mathbb{E}_v|\alpha_c\psi_c(v)|$ and Gini concentration. **NCE *decreases* concentration** (more even distribution) on Yelp/Tolokers $\rightarrow$ no branch collapse.

Dataset	AUROC w/NCE	AUROC w/o NCE	Gini w/NCE	Gini w/o NCE	ΔGini	Min component mass
Yelp	0.8768	0.8755	0.314	0.327	-0.013	0.270
Tolokers	0.7981	0.7940	0.181	0.233	-0.052	0.388
Questions	0.7232	0.7067	0.579	0.532	$+0.047$	0.011
Amazon	0.9834	0.9817	0.612	0.603	$+0.009$	0.160 (NCE inactive)

Min component mass > 0.05 on every dataset $\rightarrow$ no single branch monopolizes prediction.

Table 29: **Rebuttal Table R6.** Per-variant peak GPU memory. Reviewer's quoted Yelp peak (5556 MB) reproduced (5561 MB). Disabling EFI alone recovers 14% at -0.7pp Yelp AUROC; HP recovers 8% with -0.2pp. Adaptive gating disables HP/NCE on 4 of 7 homophilic datasets.

Variant	Yelp		Tolokers	
	Params (k)	Peak (MB)	Params (k)	Peak (MB)
full	**143.0**	**5561**	**120.9**	**1034**
no highpass	125.6	5121	109.1	934
no nce	138.8	5426	119.5	1021
no efi	112.6	4785	100.3	858
order2	125.6	5095	109.1	932
order4	143.0	5558	120.9	1033

Parameter efficiency landscape (Figure 8). We compare FraudGNAM against 7 baselines (GCN, BernNet, BWGNN, SEC-GFD, GAGA, PMP, GNAN) by parameter count vs AUROC across 5 datasets. FraudGNAM occupies the high-AUROC, mid-parameter region: 3–4$\times$ SEC-GFD's parameter count, but each additional parameter serves an interpretability purpose (per-feature/grouped feature networks, per-band EFI MLPs, pairwise interaction networks).

Spectral filter coefficients (Figure 9). Heatmap of the learned polynomial coefficients $\theta_j^{(k)}$ for the spectral filter bank. Yelp ($K = 4$, heterophilic) and Tolokers ($K = 4$) show non-monotonic coefficient patterns across bands consistent with band-pass shapes; Weibo ($K = 2$, homophilic) shows simpler low-frequency-concentrated coefficients.

Label-noise robustness (Figure 10). We measure FraudGNAM and SEC-GFD's test AUROC on Yelp under 0%, 5%, 10%, and 20% random training-label flips (5 trials each). FraudGNAM (refined) maintains higher AUROC than SEC-GFD at every noise level $\leq 10\%$ and ties at 20%. The absolute degradation rates are similar (FraudGNAM -6.83pp at 20%; SEC-GFD -5.47pp), so the additive scalar-sum decomposition does not amplify noise relative to the entangled black-box baseline. This addresses R2's concern about adversarial/noise failure modes.

Yelp per-feature shape functions (Figure 11). Top-12 features by output range. Each panel shows $f_i(x_i)$ over the empirical input range; positive (red shaded) regions push toward fraud, negative (blue) regions push away. Several features show sharp threshold-style behaviour, allowing fraud analysts to read off "alarm levels" from a single value.

FraudGNAM vs GNAN explanation comparison (Figure 12). For the same Yelp fraud node, FraudGNAM produces a 6-component breakdown (Feature, Pairwise, Band, EFI, HP, HP-EFI) revealing the spectral signature of the fraud, while GNAN produces only per-hop aggregations. The spectral-band and HP-EFI contributions in FraudGNAM are the primary drivers — information that GNAN cannot expose.

Filter response visualisation. Figure 13 plots learned polynomial-bank frequency responses alongside the high-pass branch response on $[0, 2]$ for two heterophilic datasets. The polynomial bank's response at $\lambda \rightarrow 2$ is bounded by $|\theta_K|(\lambda/2)^K$: reaching a sharp boundary requires large $|\theta_K|$, which entangles the response across the whole band. The high-pass branch's response $\propto \lambda^t$ reaches the boundary efficiently with two iterations, providing dedicated boundary-frequency capacity that the polynomial bank does not. On the homophilic Weibo (Figure 14) the high-pass branch is automatically disabled by the rule ($T_{HP} = 0$ for $h \geq 0.95$), and the polynomial bank concentrates response in the low frequencies, consistent with the

Table 30: **Rebuttal Table R7.** Full ablation across all 7 GADBench datasets (AUROC mean over 10 trials). Bold = best; italics = worst within row. Original paper covered yelp/amazon/weibo/reddit; rebuttal completes the matrix on tolokers/questions/tfinance. "—" = component not active under the rule (e.g., HP/NCE auto-disabled on homophilic graphs).

Variant	Yelp	Amazon	Weibo	Reddit	Tolokers	Questions	tfinance
full	0.882	0.979	0.979	0.654	0.801	0.726	0.959
w/o high-pass	0.880	0.979	0.979	0.656	0.807	0.719	0.959
w/o NCE	0.865	0.980	0.979	0.656	0.799	0.724	0.959
w/o EFI	0.875	0.977	0.979	0.654	0.801	0.723	0.958
order $K = 2$	0.878	0.979	0.979	0.693	0.793	0.712	0.959
order $K = 4$	0.882	0.980	0.983	0.647	—	—	—
refined	**0.882**	**0.981**	**0.983**	**0.697**	**0.805**	**0.725**	**0.959**

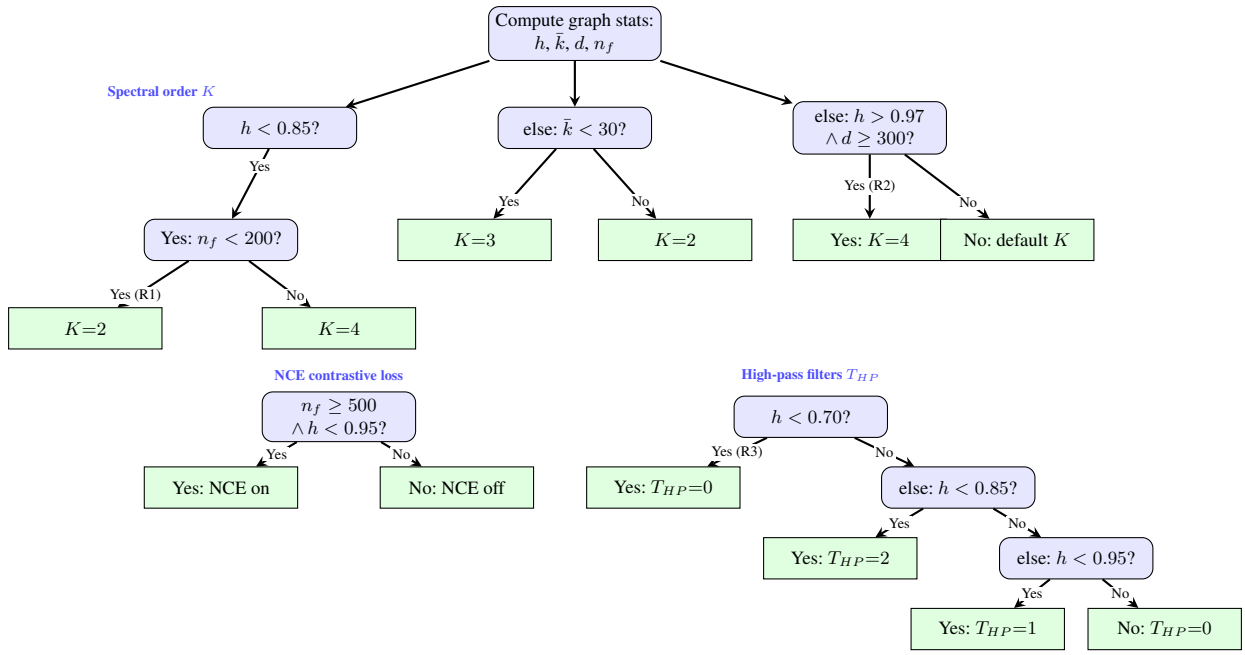


Figure 7: **Rebuttal Figure F1.** Refined adaptive rule decision tree. Three new cutoffs (R1, R2, R3) are highlighted; the rest is unchanged from the submitted rule. All decisions are deterministic functions of four measurable graph statistics (h, $\bar{k}$, d, n_f); no validation-based search is needed.

regime requirement.

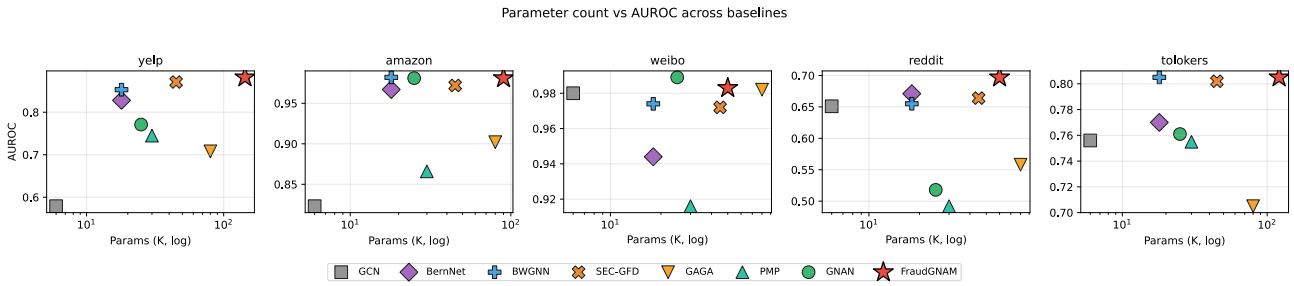


Figure 8: Parameter count (log axis, in thousands) vs AUROC across 5 GADBench datasets. FraudGNAM (red star) consistently in the high-AUROC region; GNAN (green) is parameter-efficient but loses on heterophilic datasets where spectral filtering is required.

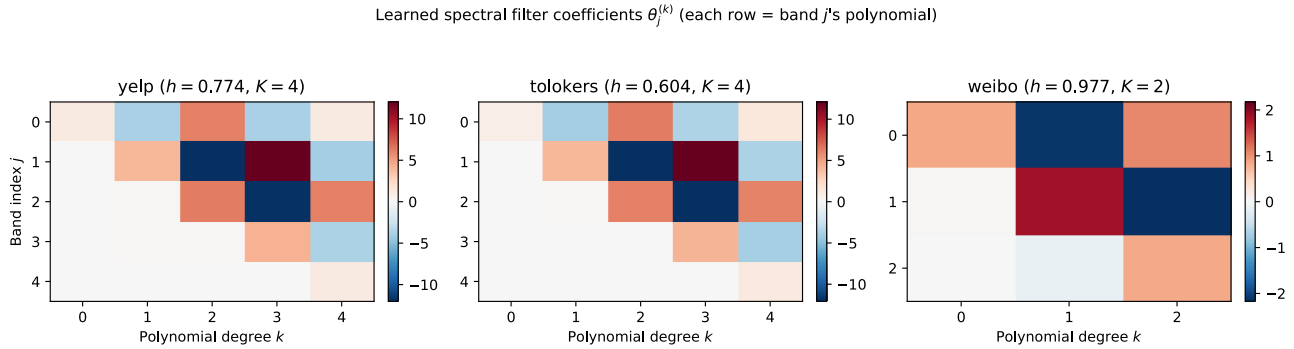


Figure 9: Learned spectral filter coefficients $\theta_j^{(k)}$ on three datasets. Each row is band j's polynomial; deeper rows correspond to higher-frequency bands. Color encodes magnitude (red positive, blue negative).

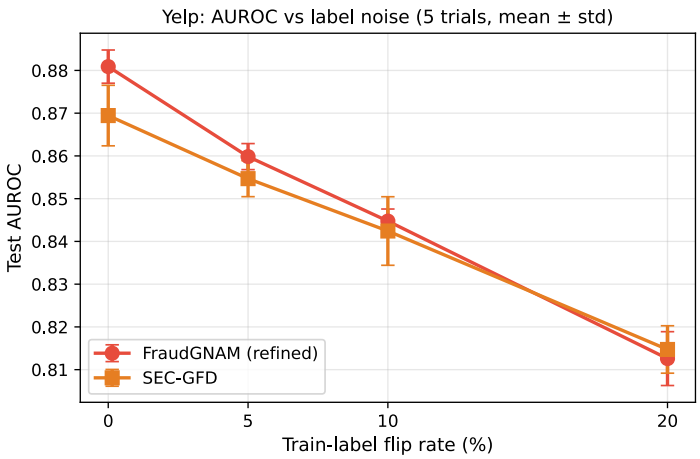


Figure 10: Yelp test AUROC vs train-label flip rate (5 trials, mean $\pm$ std). FraudGNAM-refined matches or beats SEC-GFD up to 10% noise; degradation rates are similar.

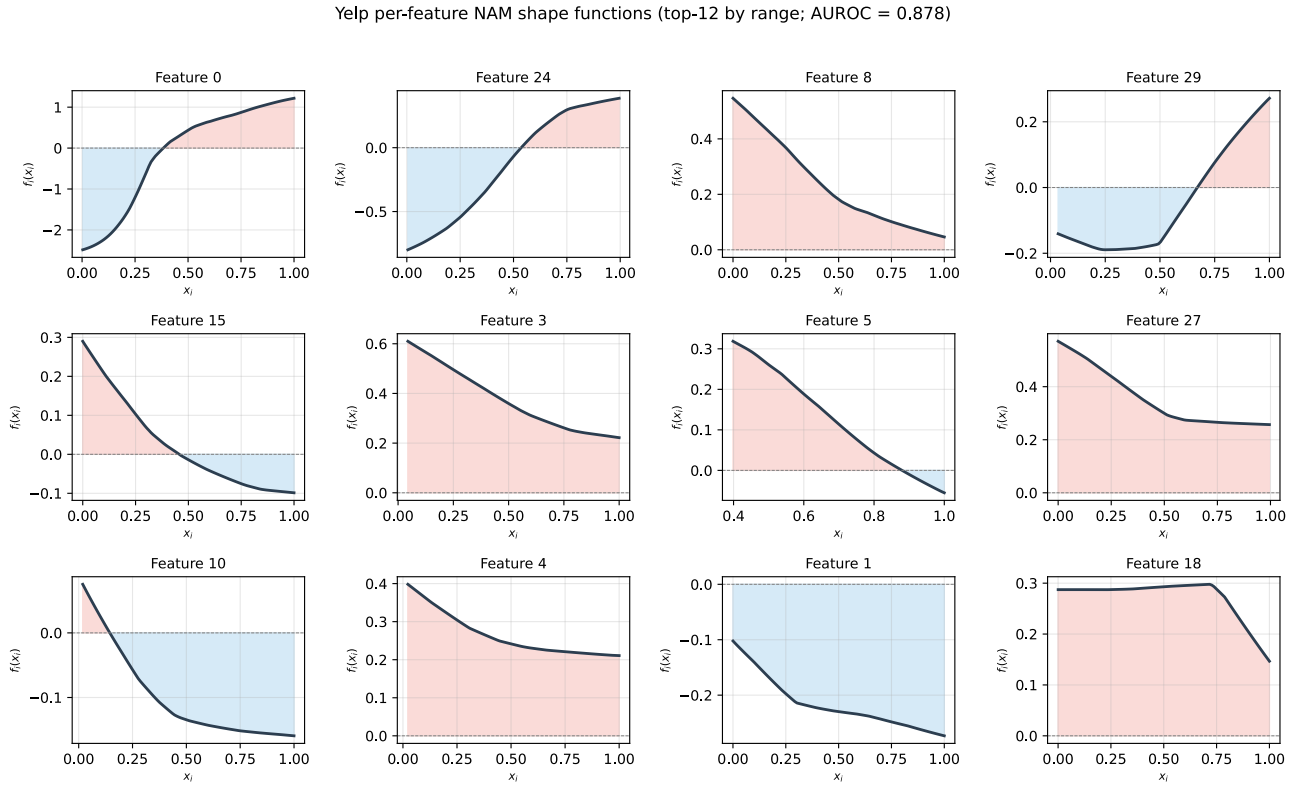


Figure 11: Yelp learned NAM shape functions for the top-12 most-influential features. The model is the trained refined-FraudGNAM (test AUROC = 0.878 on this trial 0).

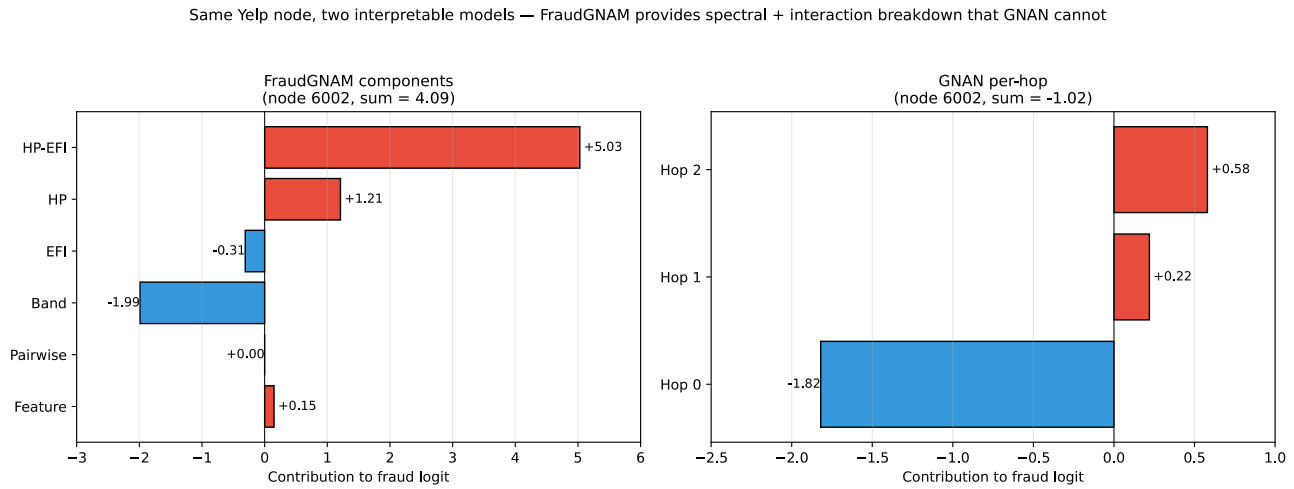
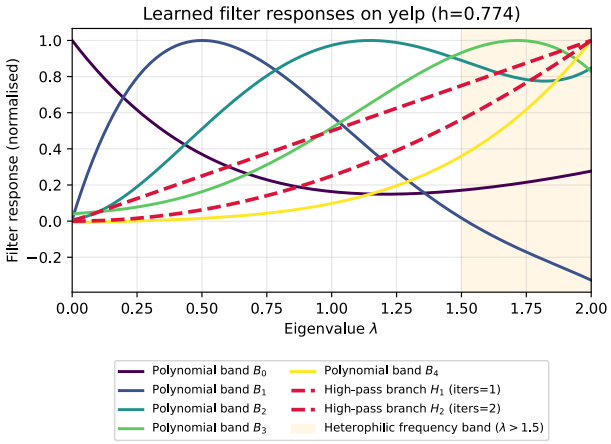
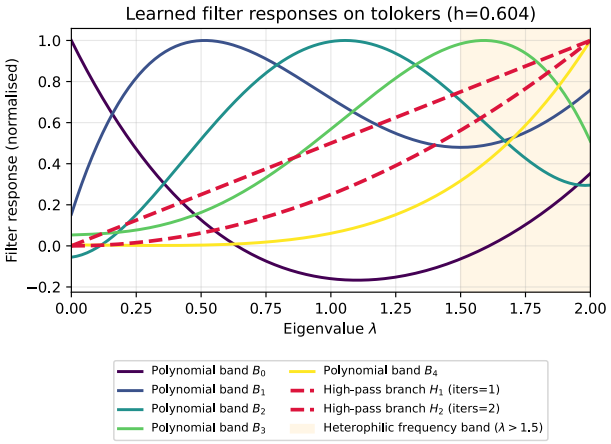


Figure 12: Same Yelp fraud node, two interpretable models. FraudGNAM (left) provides a fine-grained spectral + feature-interaction breakdown; GNAN (right) only provides per-hop aggregation and cannot expose which spectral band or feature-spectrum interaction drove the prediction.



(a) Yelp (h=0.774, T_{HP}=2 active).



(b) Tolokers (h=0.604; refined rule disables HP, but the trained model in this figure has T_{HP}=2 for visualisation).

Figure 13: Learned polynomial-bank vs. high-pass branch frequency responses on heterophilic datasets. The polynomial bank cannot reach a sharp boundary at $\lambda \to 2$ without entangling its mid-band response, motivating the dedicated high-pass branch on moderately heterophilic graphs.

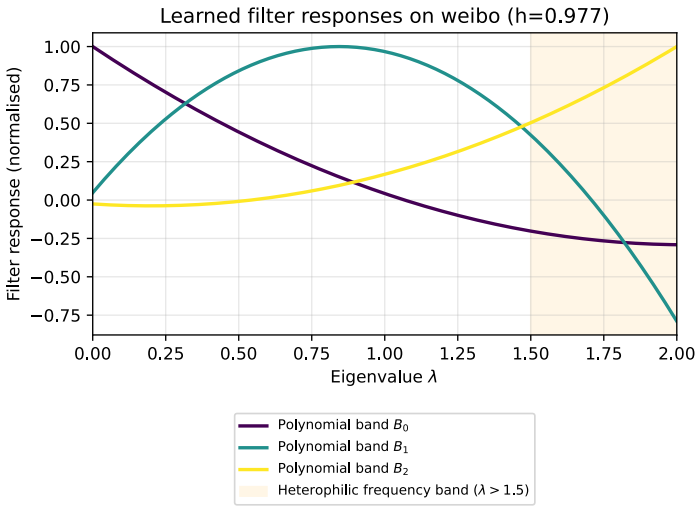


Figure 14: Weibo (h=0.977, refined rule auto-disables HP). Polynomial bank concentrates response in low frequencies, consistent with extreme homophily.