

APPENDICES

A TABLE OF NOTATIONS

We summarize the key notations used throughout this paper in Table 8.

Table 8: Summary of Notations

Notation	Description
<i>Input Data Representations</i>	
\mathcal{U}, \mathcal{I}	Set of users and items, respectively
\mathcal{D}	Historical interactions as (u, i, t) triplets
$\mathbf{X} \in \{0, 1\}^{m \times n}$	Binary user-item interaction matrix
\mathcal{S}_u	Time-ordered sequence of items for user u
<i>Graph Construction</i>	
\mathbf{S}	Initial directed item transition matrix from \mathcal{S}_u
\mathbf{S}'	Symmetric item connectivity graph derived from \mathbf{S}
α, d	Diffusion decay factor and depth
$\mathbf{S}^{(d)}$	Multi-hop diffusion matrix from \mathbf{S}' up to depth d
$\tilde{\mathbf{S}}$	Normalized diffusion matrix derived from $\mathbf{S}^{(d)}$
\mathbf{A}	Unified user-item adjacency matrix
\mathbf{L}	Normalized Laplacian matrix
<i>Spectral Filtering</i>	
$\mathbf{U}, \mathbf{\Lambda}$	Matrices of eigenvectors and eigenvalues of \mathbf{L} , respectively
r	Number of eigenvectors in truncated decomposition
$g_{\text{BP}}(\lambda)$	Gaussian bandpass filter kernel
c, w	Center and width of bandpass filter
\mathbf{F}_{BP}	Bandpass filtered user-item signal
\mathbf{F}_{LP}	Low-pass filtered user-item signal
ϕ	Fusion parameter balancing filter components
\mathbf{Y}	Final prediction matrix

B ALGORITHMIC DETAILS

B.1 GSPREC ALGORITHM

The full end-to-end procedure of our method is outlined in Algorithm 1. It summarizes the construction of the sequential graph encoding, Laplacian computation, spectral filtering, and final score generation.

B.2 PROOF OF PROPOSITION 2: LAPLACIAN VALIDITY

Proof. We first establish properties of the adjacency matrix $\mathbf{A} \in \mathbb{R}^{(m+n) \times (m+n)}$, defined as:

$$\mathbf{A} = \begin{pmatrix} \mathbf{0}_{m \times m} & \mathbf{X} \\ \mathbf{X}^T & \tilde{\mathbf{S}} \end{pmatrix} \quad (12)$$

where $\mathbf{X} \in \{0, 1\}^{m \times n}$ (the binary user-item interaction matrix) and $\tilde{\mathbf{S}} = \mathbf{D}_S^{-1/2} \mathbf{S}^{(d)} \mathbf{D}_S^{-1/2} \in \mathbb{R}^{n \times n}$ (the normalized sequence-derived item-item graph, with $\mathbf{S}^{(d)}$ being the diffused matrix from Eq. 2).

Lemma 4. *The adjacency matrix \mathbf{A} is symmetric and non-negative.*

Algorithm 1: GSPRec

Input: Interaction set $\mathcal{D} = \{(u, i, t)\}$, diffusion order d , decay factor α , filter parameters (c, w) , fusion weight ϕ

Output: Predicted score matrix $\mathbf{Y} \in \mathbb{R}^{m \times n}$

// Graph Construction

$\mathbf{X} \leftarrow$ Binary user-item matrix from \mathcal{D}

$\mathcal{S}_u \leftarrow$ Chronologically sorted item sequences per user

$\mathbf{S} \leftarrow$ Initial directed item transition matrix from \mathcal{S}_u (where $S_{ij} = 1$ if $i \rightarrow j$)

$\mathbf{S}' \leftarrow \text{Symmetrize}(\mathbf{S})$ // $S'_{ij} = 1$ if $S_{ij} = 1$ or $S_{ji} = 1$

$\mathbf{S}^{(d)} \leftarrow \sum_{k=1}^d \alpha^{k-1} (\mathbf{S}')^k$ // Exponential diffusion on symmetric \mathbf{S}'

$\tilde{\mathbf{S}} \leftarrow \mathbf{D}_S^{-1/2} \mathbf{S}^{(d)} \mathbf{D}_S^{-1/2}$ // where $\mathbf{D}_S = \text{diag}(\mathbf{S}^{(d)} \mathbf{1}_n)$

$\mathbf{A} \leftarrow \begin{bmatrix} \mathbf{0} & \mathbf{X} \\ \mathbf{X}^\top & \tilde{\mathbf{S}} \end{bmatrix}$

$\mathbf{D} \leftarrow \text{diag}(\mathbf{A} \mathbf{1})$

$\mathbf{L} \leftarrow \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2}$

$[\mathbf{U}, \mathbf{\Lambda}] \leftarrow$ Eigendecomposition of \mathbf{L}

// Frequency-Aware Spectral Filtering

$\bar{\lambda} \leftarrow$ Diagonal entries of $\mathbf{\Lambda}$

$\mathbf{G}_{\text{BP}} \leftarrow \text{diag} \left(\exp \left(-\frac{(\bar{\lambda} - c)^2}{w} \right) \right)$ // Gaussian band-pass

$\mathbf{F}_{\text{BP}} \leftarrow \mathbf{X} \mathbf{D}_I^{-1/2} \mathbf{U} \mathbf{G}_{\text{BP}} \mathbf{U}^\top \mathbf{D}_I^{-1/2}$

// Low-Pass Filtering

$\tilde{\mathbf{X}}_U \leftarrow \mathbf{D}_U^{-1/2} \mathbf{X}$ // where $\mathbf{D}_U = \text{diag}(\mathbf{X} \mathbf{1}_n)$, $\mathbf{D}_I = \text{diag}(\mathbf{X}^\top \mathbf{1}_m)$

$\mathbf{C}_U \leftarrow \tilde{\mathbf{X}}_U \tilde{\mathbf{X}}_U^\top$

$\mathbf{X}_b \leftarrow [\mathbf{C}_U, \mathbf{X}]$

$\mathbf{D}_b \leftarrow \text{diag}(\mathbf{X}_b^\top \mathbf{1})$

$\mathbf{F}_{\text{LP}} \leftarrow \mathbf{X}_b \mathbf{D}_b^{-1/2} \mathbf{U} \mathbf{U}^\top \mathbf{D}_b^{1/2}$

$\mathbf{F}_{\text{LP}} \leftarrow \mathbf{F}_{\text{LP}}[:, m:]$ // Select only item columns

// Fusion and Output

$\mathbf{Y} \leftarrow \phi \cdot \mathbf{F}_{\text{BP}} + (1 - \phi) \cdot \mathbf{F}_{\text{LP}}$

return \mathbf{Y}

Proof of Lemma 4 To prove symmetry of \mathbf{A} , we note:

$$\mathbf{A}^T = \begin{pmatrix} \mathbf{0}_{m \times m}^T & (\mathbf{X}^T)^T \\ \mathbf{X}^T & \tilde{\mathbf{S}}^T \end{pmatrix} \quad (13)$$

$$= \begin{pmatrix} \mathbf{0}_{m \times m} & \mathbf{X} \\ \mathbf{X}^T & \tilde{\mathbf{S}} \end{pmatrix} = \mathbf{A} \quad (14)$$

This equality holds because $\mathbf{0}_{m \times m}$ is symmetric, $(\mathbf{X}^T)^T = \mathbf{X}$, and $\tilde{\mathbf{S}}^T = \tilde{\mathbf{S}}$. The matrix $\tilde{\mathbf{S}}$ is symmetric by construction: the base matrix \mathbf{S}' used in the diffusion is symmetric, thus $\mathbf{S}^{(d)}$ (from Eq. 2) is symmetric, and the symmetric normalization (Eq. 3) preserves this symmetry.

For non-negativity of \mathbf{A} , we show all its elements are non-negative:

- $\mathbf{X} \in \{0, 1\}^{m \times n}$ by definition, so its elements are ≥ 0 . Thus, $\mathbf{X} \geq 0$ and $\mathbf{X}^T \geq 0$.
- For $\tilde{\mathbf{S}}$: The initial directed transition matrix \mathbf{S} has $s_{ij} \in \{0, 1\}$, so $\mathbf{S} \geq 0$. The symmetric base graph \mathbf{S}' is constructed from \mathbf{S} such that its entries $s'_{ij} \in \{0, 1\}$, so $\mathbf{S}' \geq 0$. Given $\mathbf{S}' \geq 0$ and the decay factor $\alpha \in (0, 1) > 0$, each power $(\mathbf{S}')^k$ in the diffusion sum will have non-negative entries. Therefore, $\mathbf{S}^{(d)} = \sum_{k=1}^d \alpha^{k-1} (\mathbf{S}')^k$ (Eq. 2) will also have non-negative entries, i.e., $\mathbf{S}^{(d)} \geq 0$. The degree matrix $\mathbf{D}_S = \text{diag}(\mathbf{S}^{(d)} \mathbf{1}_n)$ will have non-negative diagonal entries. For non-isolated nodes in the graph represented by $\mathbf{S}^{(d)}$,

these entries will be positive, making $\mathbf{D}_S^{-1/2}$ well-defined with non-negative real entries. Hence, $\tilde{\mathbf{S}} = \mathbf{D}_S^{-1/2} \mathbf{S}^{(d)} \mathbf{D}_S^{-1/2} \geq 0$.

Therefore, \mathbf{A} is both symmetric and non-negative. \square

Given that \mathbf{A} is symmetric and non-negative (by Lemma 4), and its degree matrix $\mathbf{D} = \text{diag}(\mathbf{A}\mathbf{1})$, the normalized Laplacian is $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2}$ (Eq. 6). We apply Theorem 5:

Theorem 5 (Properties of Normalized Laplacian (Chung, 1997 Chung (1997))). *For a symmetric, non-negative adjacency matrix \mathbf{A} with degree matrix \mathbf{D} , the normalized Laplacian $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-1/2} \mathbf{A} \mathbf{D}^{-1/2}$ satisfies: (1) \mathbf{L} is real and symmetric. (2) \mathbf{L} is positive semi-definite. (3) All eigenvalues of \mathbf{L} lie in $[0, 2]$.*

Applying Theorem 5 directly to our \mathbf{A} and \mathbf{L} concludes that \mathbf{L} is symmetric, positive semi-definite, and has eigenvalues in $[0, 2]$. \square

C THEORETICAL ANALYSIS OF SPECTRAL BANDS

The concentration of user-specific patterns in mid-frequency components can be understood through the lens of graph signal smoothness. Low-frequency components correspond to signals that are smooth across the graph, capturing global popularity trends that vary slowly across the user-item structure. High-frequency components often represent noise or sparse outlier preferences.

Mid-frequency components represent a balance where signals vary moderately across local graph neighborhoods - precisely the pattern exhibited by personalized preferences that differ between user communities but show consistency within them. This aligns with previous findings in spectral clustering Von Luxburg (2007) and network community detection Newman (2006), where mid-frequency eigenvectors have been shown to encode community structure.

We can formalize this intuition through the concept of total variation of a signal Shuman et al. (2013). Given a signal \mathbf{f} on a graph with Laplacian \mathbf{L} , the quadratic form $\mathbf{f}^T \mathbf{L} \mathbf{f}$ measures how smoothly the signal varies across the graph. For the eigenvectors \mathbf{u}_i of \mathbf{L} , we have $\mathbf{u}_i^T \mathbf{L} \mathbf{u}_i = \lambda_i$, where λ_i is the corresponding eigenvalue.

User preference signals can be decomposed into these eigenvectors:

$$\mathbf{f} = \sum_{i=1}^{m+n} \alpha_i \mathbf{u}_i \quad (15)$$

In our empirical analysis across datasets, we consistently observe that coefficients α_i corresponding to mid-frequency components ($0.3 < \lambda_i < 0.8$) have significantly higher magnitude for user-specific preference patterns compared to global trends. This provides empirical validation of our theoretical understanding of spectral representation in recommendation graphs.

D COMPUTATIONAL COMPLEXITY ANALYSIS

GSPRec has time complexity dominated by eigendecomposition and filtering operations. For sparse datasets with $r \ll (m+n)$, the method remains tractable on benchmark datasets as demonstrated by our runtime analysis (Table 7). For large-scale deployments, approximation techniques like polynomial filtering can reduce computational requirements to $O(K \cdot |\mathcal{D}|)$ for K -order polynomials, making the approach practical for industrial applications.

E BASELINE DETAILS

We compare GSPRec with a comprehensive set of baselines representing both GCN-based and GSP-based CF methods:

GCN-based CF Methods:

- **LightGCN** [He et al. (2020)] simplifies GCN for recommendation by removing feature transformation and non-linear activation, retaining only essential neighborhood aggregation.
- **LR-GCCF** [Chen et al. (2020b)] introduces residual connections to graph convolutional networks for improving recommendation performance.
- **IMP-GCN** [Liu et al. (2021)] propagates information in user sub-graphs to reduce the impact of noise and alleviate the over-smoothing problem.
- **SimpleX** [Mao et al. (2021a)] focuses on optimizing loss function and negative sampling strategies.
- **UltraGCN** [Mao et al. (2021b)] approximates infinite-layer graph convolutions via constraint loss and flexible edge weight assignment.

GSP-based CF Methods:

- **GF-CF** [Shen et al. (2021)] integrates linear and ideal low-pass filters to model user preferences.
- **JGCF** [Guo et al. (2023)] uses Jacobi polynomial-based filters to amplify high-frequency signals that are typically suppressed by standard GCNs.
- **PGSP** [Liu et al. (2023)] uses a mixed-frequency low-pass filter over personalized graph signals.
- **HiGSP** [Xia et al. (2024)] adopts cluster-wise and globally-aware filters to recognize unique and general interaction patterns.
- **FaGSP** [Xia et al. (2025)] employs Cascaded and Parallel Filter Modules for user preference modeling.

For comprehensive evaluation, We use official repositories for methods for reproducing results on our dataset settings. We report results for LR-GCCF, IMP-GCN, SimpleX, UltraGCN, and GF-CF as presented in [Xia et al. (2024; 2025)], which used identical dataset preprocessing and evaluation protocols.

E.1 FILTER DESIGN

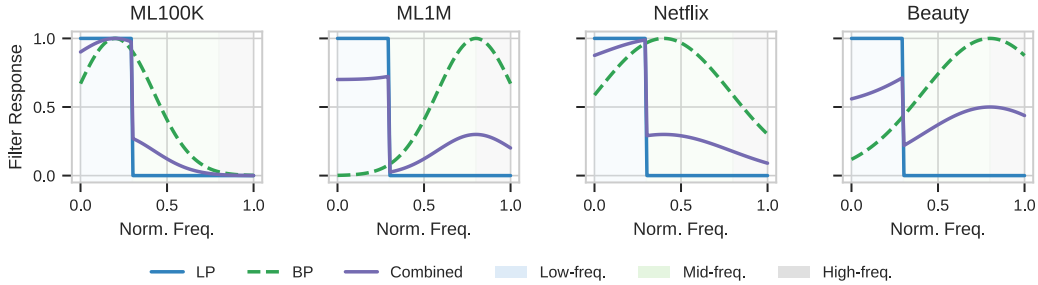


Figure 6: Frequency response curves of different filter configurations all datasets. The low-pass filter (blue) preserves signals in the lower 30% of the spectrum. The band-pass filter (green) selectively amplifies mid-frequency components associated with personalized patterns. The combined filter response (purple) illustrates how dual-filter captures complementary signals across the spectrum.

F REPRODUCIBILITY

To improve reproducibility, we provide the complete algorithm of GSPRec in Algorithm 1, which details the graph construction, spectral decomposition, and dual filtering processes. The optimal hyperparameters for all experimental datasets are summarized in Table 3. For implementation, we

process datasets following the procedure described in Section 4.1, which involves splitting datasets in an 8:2 ratio for training/testing, with 10% of the training set used for validation. When implementing GSPRec, the key parameter settings are:

- Diffusion depth $d = 2$ and decay factor $\alpha = 0.4$ for sequential graph construction
- Dataset-specific eigenvector count r , band-pass center c , width w , and fusion weight ϕ as specified in Table 3

For evaluation, we follow the protocol detailed in Section 4.1, using NDCG and MRR metrics at $k \in \{5, 10, 20\}$ after excluding previously interacted items.

F.1 COMPUTATIONAL RESOURCES

Experiments were conducted on a system with AMD EPYC processors and NVIDIA RTX A6000 GPUs. The computational requirements varied with dataset size and the number of eigenvectors used. The eigendecomposition step is the primary computational bottleneck as identified in our complexity analysis (Appendix D). For benchmark datasets used in our experiments, the runtime and memory requirements were reasonable, allowing for comprehensive evaluation on standard research hardware. Our experimental protocol, including parameter tuning and ablation studies, involved multiple runs for each configuration to ensure statistical significance.

F.2 SCALING TO LARGE-SCALE SYSTEMS

For industrial-scale applications with millions of users and items, we propose three optimization strategies:

F.2.1 APPROXIMATE EIGENDECOMPOSITION

The dominant computational bottleneck can be addressed using the Nyström method (Fowlkes et al. (2004)) or randomized SVD approaches (Halko et al. (2011)), reducing complexity from $O(r(m+n)^2)$ to $O(r^2(m+n))$ (or better, depending on specific variant, e.g., linear in $(m+n)$ for matrix-vector products), with minimal performance impact.

F.2.2 GRAPH SPARSIFICATION

For large-scale graphs, we can apply spectral sparsification techniques (Spielman & Teng (2011)) that preserve the essential spectral properties while reducing edge density, accelerating both graph construction and eigendecomposition.

F.2.3 INCREMENTAL UPDATES

In production environments, the full spectral decomposition need not be recomputed for each recommendation. We propose an incremental update scheme where the spectral components are updated periodically while incremental preference updates can be applied in real-time.

G MATHEMATICAL ANALYSIS OF BANDPASS FILTERING

We provide mathematical justification for our bandpass filtering approach and its relevance to recommendation systems. Our Gaussian kernel on the graph Laplacian eigenspectrum is defined as in Equation 7

$$g(\lambda) = \exp\left(-\frac{(\bar{\lambda} - c)^2}{w}\right), \quad \bar{\lambda} = \frac{\lambda - \lambda_{\min}}{\lambda_{\max} - \lambda_{\min}}$$

This satisfies the formal definition of a bandpass filter: it achieves maximum response when $\bar{\lambda} = c$ (which corresponds to $\lambda_c = c(\lambda_{\max} - \lambda_{\min}) + \lambda_{\min}$ in the original eigenvalue domain) and attenuates components with eigenvalues far from λ_c . For our normalized Laplacian with original eigenvalues $\lambda \in [0, 2]$ (so $\lambda_{\min} \approx 0, \lambda_{\max} \approx 2$), setting $c \in (0, 1)$ (the center in the normalized $\lambda \in [0, 1]$ domain) ensures the filter passes a band of mid-range eigenvalues while attenuating both low and

high extremes. When applied to user preference signals decomposed as $\mathbf{x} = \sum_{i=1}^{m+n} \alpha_i \mathbf{u}_i$, the filtered output $\mathbf{y} = \mathbf{U} \text{diag}(g(\boldsymbol{\Lambda})) \mathbf{U}^T \mathbf{x}$ selectively preserves components $\alpha_i \mathbf{u}_i$ where the eigenvalue λ_i (corresponding to \mathbf{u}_i) results in a high $g(\lambda_i)$ value.

Our empirical analysis across recommendation datasets reveals that coefficients α_i corresponding to intermediate eigenvalues ($0.3 < \lambda_i < 0.8$, assuming λ_i are original eigenvalues in $[0, 2]$) have higher magnitude for user-specific preference patterns compared to global trends. This confirms that mid-frequency components encode valuable personalization signals that traditional low-pass approaches would suppress.

Table 6 provides quantitative evidence of this phenomenon: while low-pass filtering captures popularity trends, our bandpass approach more effectively preserves the user-specific patterns essential for accurate personalized recommendations.

H BROADER IMPACT

This work improves recommender systems by integrating sequential behavior with graph spectral analysis, benefiting users across e-commerce, media, and education domains. However, spectral methods may amplify biases in historical data, potentially reinforcing popularity bias or increasing computational overhead. Future work should explore fairness-aware spectral filters, robustness evaluations, and privacy-preserving techniques. We encourage practitioners to consider these tradeoffs when deploying these systems in real-world settings.

LLM USAGE STATEMENT

LLMs were used to assist with polishing and formatting portions of this manuscript only. The models were not involved in research ideation, experimental design, or data analysis.