Spectral-based Graph Neutral Networks for Complementary Item Recommendation

Haitong Luo[1, 2], **Xuying Meng**[1, 5], **Suhang Wang**[3], **Hanyun Cao**[1, 2]
Weiyao Zhang[1], **Yequan Wang**[4], **Yujun Zhang**[1, 2, 6*]

[1]Institute of Computing Technology, Chinese Academy of Sciences, [2]University of Chinese Academy of Sciences,
[3]Pennsylvania State University, [4]BAAI, [5]Purple Mountain Laboratories, [6]Nanjing Institute of InforSuperBahn
{luohaitong21s, nrcyujun}@ict.ac.cn

Abstract

Modeling complementary relationships greatly helps recommender systems to accurately and promptly recommend the subsequent items when one item is purchased. Unlike traditional similar relationships, items with complementary relationships may be purchased successively (such as iPhone and Airpods Pro), and they not only share relevance but also exhibit dissimilarity. Since the two attributes are opposites, modeling complementary relationships is challenging. Previous attempts to exploit these relationships have either ignored or oversimplified the dissimilarity attribute, resulting in ineffective modeling and an inability to balance the two attributes. Since Graph Neural Networks (GNNs) can capture the relevance and dissimilarity between nodes in the spectral domain, we can leverage spectral-based GNNs to effectively understand and model complementary relationships. In this study, we present a novel approach called Spectral-based Complementary Graph Neural Networks (SComGNN) that utilizes the spectral properties of complementary item graphs. We make the first observation that complementary relationships consist of low-frequency and mid-frequency components, corresponding to the relevance and dissimilarity attributes, respectively. Based on this spectral observation, we design spectral graph convolutional networks with low-pass and mid-pass filters to capture the low-frequency and mid-frequency components. Additionally, we propose a two-stage attention mechanism to adaptively integrate and balance the two attributes. Experimental results on four e-commerce datasets demonstrate the effectiveness of our model, with SComGNN significantly outperforming existing baseline models.

Introduction

Complementary item recommendation (Liu et al. 2020; Hao et al. 2020; Bibas, Shalom, and Jannach 2023) aims to suggest related items to users after they make a purchase in order to stimulate further purchases. To ensure the success of an e-commerce platform, it is crucial to model the complementary relationships between items. Complementary relationships involve items that are relevant yet dissimilar, as they are purchased together but serve different functions (e.g., iPhone and AirPods Pro). These properties make complementary relationships more challenging to model than traditional similarity relationships (also known as substitutable relationships). In this study, we focus on complementary item recommendation, i.e., given a query item, the goal is to recommend relevant yet dissimilar items to satisfy users' needs and encourage joint purchases.

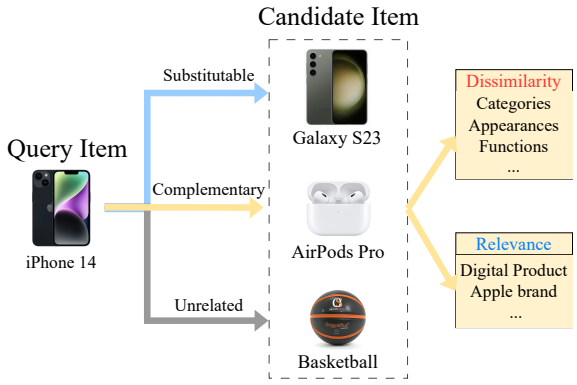

Figure 1: Item relationships in recommender systems.

The core attributes of complementary relationships are relevance and dissimilarity. As shown in Figure 1, for iPhone and AirPods Pro, they are relevant as digital products under the Apple brand. Their dissimilarity lies in that they are different products with different functions and appearances. When recommending complementary products for users, it is crucial to understand and balance these two characteristics. On one hand, if we emphasize their relevance too much, it may lead to substitutable item recommendations. On the other hand, if we emphasize their dissimilarity too much, it may lead to recommendations for unrelated items.

Hence, researchers have made many efforts on complementary item recommendations. Some works (McAuley, Pandey, and Leskovec 2015; Wang et al. 2018; Cen et al. 2019; Liu et al. 2020; Chen et al. 2023) tentatively decouple and focus on complementary relations from item relationships. However, they ignore the dissimilarity attribute and only consider the relevance. To further model the dissimilarity attribute, recent works (Hao et al. 2020; Bibas, Shalom, and Jannach 2023) model the dissimilarity with category mapping networks that consider category diversity. However, these works still simplify the complementary relationships since the dissimilarity is not limited to categories. Without a deep understanding of these two attributes, exist-

*Corresponding Author

ing works fail to model the essence of complementary relationships, which also leads to an inability to explore the trade-off between the two properties.

Recent advances show that GNNs can capture the relevance and dissimilarities of nodes from the spectral domain (Wu et al. 2022; Tang et al. 2022), which provides a promising direction to model the complementary relationships for simultaneously capturing the relevance and dissimilarity. Thus, in this work, we model complementary relationships with spectral-based GNNs. However, we are faced with two challenges: (1) **the lack of a deep understanding of complementary relationships from a spectral perspective**. Existing spectral-based GNNs do not explore and adapt to the spectral properties of complementary relationships, resulting in a gap between their spectral properties and the two attributes; (2) **the trade-off between the relevance and dissimilarity attributes**. Since the two attributes are opposites, over-emphasizing either one can lead to inaccurate complementary item recommendations. Therefore, it is crucial to strike a balance between these two attributes.

In an attempt to address these challenges, we first analyze complementary relationships from a spectral perspective on graphs and observe that the spectrum of the complementary item graph is mainly composed of low-frequency and mid-frequency components, which correspond to the relevance and dissimilarity characteristics respectively. Based on the observation, we design low-pass and mid-pass graph convolutional networks to decouple and extract the corresponding low-frequency relevance and mid-frequency dissimilarity components. To balance the two attributes, we propose a two-step attention mechanism to adaptively integrate and balance them. Our contributions are summarized as follows:

- We conduct the first study to gain an understanding of the spectral properties of complementarity relationships based on GNNs, which associate the low-frequency and mid-frequency components with relevance and dissimilarity, respectively.
- We design a novel model with spectral-based GNNs and a two-stage attention mechanism, to decouple, extract and adaptively balance the low-frequency relevance and mid-frequency dissimilarity.
- We demonstrate the effectiveness of our proposed framework on four publicly available datasets, which outperforms the state-of-art approaches by a margin.

Related Work

In this section, we introduce related work on graph neural networks and complementary item recommendations.

Graph Neural Networks

GNNs (Wu et al. 2020) have shown great ability in modeling graph-structured data. Generally, GNNs can be classified into two main forms, i.e., spatial-based and spectral-based ones. *Spatial-based GNNs* (Hamilton, Ying, and Leskovec 2017; Veličković et al. 2017; Gao, Wang, and Ji 2018; Zhu et al. 2023) operate in the spatial domain, where the graph convolution is defined in terms of the neighborhood structure of each node. *Spectral-based GNNs* (Bruna et al. 2013; Defferrard, Bresson, and Vandergheynst 2016; Kipf and Welling 2016; Balcilar et al. 2020; Wu et al. 2022) operate in the spectral domain, where the graph convolution filter is defined in terms of the eigenvectors of the graph Laplacian matrix. Since GCN only utilizes low-frequency information (Balcilar et al. 2020), to broaden the available frequency bandwidth, recent studies (Balcilar et al. 2020; Wu et al. 2022) attempt to designs filter functions to incorporate all the bands of graph signals.

Complementary Item Recommendations

To maximize profit and provide convenience to users, modeling item relationships is a crucial task in recommendation systems. However, existing works (Wang, Sarwar, and Sundaresan 2011; Yao and Harper 2018; Meng et al. 2018) often oversimplify item relationships as merely being "related", disregarding the fact that these relationships can be further categorized as substitutable or complementary ones. *Complementary* relationships involve items that are relevant yet dissimilar, making the modeling of such relationships more challenging while *substitutable* items are almost similar. To tackle this, one straightforward method is frequent pattern mining and association rules (Han et al. 2007).

Recently, deep learning methods have been applied to recommend complementary items. Some studies (McAuley, Pandey, and Leskovec 2015; Wang et al. 2018; Cen et al. 2019; Liu et al. 2020; Wu, Zhou, and Zhou 2022; Chen et al. 2023) decouple and focus specifically on complementary relationships from item relationships to provide more precise recommendations. However, they tend to ignore the dissimilarity attribute. In response, some works try to consider the dissimilarity attribute. For example, P-companion (Hao et al. 2020) and ALCIR (Bibas, Shalom, and Jannach 2023) propose category mapping networks to recommend complementary items and include category diversity.

Since the dissimilarity is related to not only categories but also other features such as appearances and prices, these works still oversimplify the dissimilarity and do not have a deep understanding of the complementary relationships. Furthermore, their inability to accurately model and decouple the two attributes prevents them from striking a balance between relevance and dissimilarity.

Problem Statement and Motivation

We first provide preliminaries and definitions of our graph-based complementary item recommendation problem.

Notations and Problem Definition

Let $\mathcal{G} = \{\mathcal{V}, \mathbf{X}, \mathcal{E}\}$ denotes the complementary item graph, where $\mathcal{V}$ is the set of nodes $\{v_1, ..., v_N\}$ and each node is an item. $\mathcal{E} = \{e_{ij}\}$ is the set of undirected edges. Feature matrix $\mathbf{X} \in \mathbb{R}^{N \times d}$ is made of $d-$dimensional features of N nodes. Let $\mathbf{A} \in \mathbb{R}^{N \times N}$ denotes the adjacency matrix. $\mathbf{A}_{ij} = 1$ if v_i and v_j are complementary; otherwise, $\mathbf{A}_{ij} = 0$. Let $\mathbf{D} \in \mathbb{R}^{N \times N}$ be the diagonal degree matrix with $\mathbf{D}_{ii} = \sum_j \mathbf{A}_{ij}$. The normalized graph Laplacian matrix $\mathbf{L} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$, where $\mathbf{I}$ is an identity matrix. With

Dataset	Appliances	Grocery	Toys	Home
# items	804	38548	24638	75514
# edges	8290	642884	614730	776766
S_{high}	0.3408	0.4034	0.3150	0.4169

Table 1: Statistics and S_{high} of four datasets.

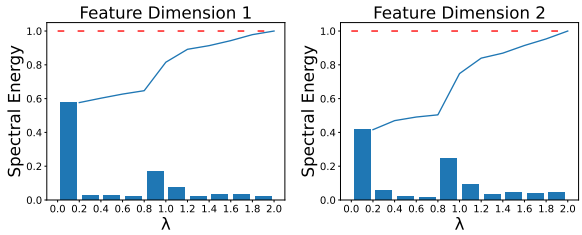

Figure 2: Spectral energy distribution of Appliances dataset.

these notations, we formally define the problem of graph-based complementary item recommendation as:

Problem 1. *For a complementary item graph* $\mathcal{G} = \{\mathcal{V}, \mathbf{X}, \mathcal{E}\}$*, where nodes denote items and edges denote complementary relationships, we aim to predict the probability of an edge* $e_{i,j}$ *when given two items* v_i *and* v_j*, and find items being complementary accordingly.*

Observations on Real-world Datasets

To observe the spectral properties of the complementary item graphs, we first introduce two metrics (Tang et al. 2022), i.e., **spectrum** and **high-frequency area**.

(1) The spectrum visualizes frequency distribution in spectral domains. It is plotted using eigenvalues λ as the x-axis and the spectral energy as the y-axis. The eigenvalues $\lambda = \{\lambda_1, \lambda_2, ..., \lambda_N\}$ and the corresponding eigenvectors $\mathbf{U} = (\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_N)$ can be obtained by the decomposition of the normalized Laplacian matrix $\mathbf{L}$, where eigenvalues λ also denote the frequencies of the graph. The spectral energy $\hat{x}_k^2 / \sum_{i=1}^N \hat{x}_i^2$ is based on the graph Fourier transform $\hat{\mathbf{x}} = (\hat{x}_1, \hat{x}_2, ..., \hat{x}_N)^T = \mathbf{U}^T\mathbf{x}$, where $\mathbf{x} = (x_1, x_2, ..., x_N)^T \in \mathbb{R}^N$ denotes one dimension of features from N nodes. Since λ ranges from 0 to 2, we define λ close to 2 as high frequencies, λ close to 0 as low frequencies, and λ close to 1 as medium frequencies. Due to the huge computational effort of eigenvalue decomposition, only small-scale graph datasets can be drawn for spectrum plots.

(2) The high-frequency area S_{high} denotes the area of the high-frequency region in the spectrum. It measures the area between the accumulated spectral energy curve (the solid lines in Figure 2) and the curve with a y-value of 1 (the dashed lines in Figure 2). Thus, S_{high} is within $[0, 2]$. Previous work (Tang et al. 2022) shows that S_{high} can be obtained by $S_{high} = \frac{\sum_{k=1}^N \lambda_k \hat{x}_k^2}{\sum_{k=1}^N \hat{x}_k^2} = \frac{\mathbf{x}^T\mathbf{L}\mathbf{x}}{\mathbf{x}^T\mathbf{x}}$. There is no need for eigenvalue decomposition, making it feasible for large-scale datasets. The larger the S_{high} is, the more the mid- and high-frequency components are.

Based on the two metrics, we conduct our analysis on four real-world datasets obtained from Amazon (He and McAuley 2016), i.e., "Appliances", "Grocery and Gourmet Food" (abbreviated as Grocery), "Toys and Games" (Toys), and "Home and Kitchen" (Home). The details of the datasets can be found in the experiment section. "Appliances" is a small-scale dataset, while the others are large-scale datasets. In each dataset, nodes represent items and edges represent complementary relationships.

We draw the spectrums for the small-scale dataset and compute high-frequency area S_{high} for all datasets. Since both the spectrum and S_{high} are obtained based on a single feature dimension, we randomly select two dimensions to plot the spectrum in Figure 2 and calculate the average value of all feature dimensions to obtain S_{high} in Table 1. In Figure 2, the histogram denotes the spectral energy distribution and the solid curve denotes the accumulated spectral energy distribution.

From Figure 2 and Table 1, we can conclude that the complementary item graph is mainly composed of low-frequency and mid-frequency components in the spectral domain. In detail, (1) from Figure 2, the spectrum shows that λ with low and medium values have larger spectral energies, which means the complementary relationship is composed of low-frequency and mid-frequency components; and (2) from Tabel 1, the high-frequency area S_{high} of all the datasets fall between 0 and 1, indicating that similar to the Appliances dataset, the spectrum of the other three datasets are mainly composed of low-frequency and mid-frequency parts. Additionally, as S_{high} is below 0.5, the low-frequency component is greater than the mid-frequency component.

Since the more similar a node is to its neighbors in the spatial domain, the lower the corresponding frequency component is in the spectral domain (Wu et al. 2022; Bo et al. 2021), we can regard the low-frequency component as the relevance attribute and the mid-frequency component as the dissimilarity attribute. To further verify it, we conduct a case study experiment in the experiment section. In this way, we bridge the gap between the properties of the complementary relationship and the spectral characteristics.

Methodology

Based on our observations, we propose a novel framework for complementary item prediction by modeling the two attributes in the spectral domain. As illustrated in Figure 3, the framework consists of three key modules: spectral-based GCN filters, a two-stage attention mechanism, and contrastive learning optimization. (1) To model the low-frequency relevance and mid-frequency dissimilarity, we decouple and extract them using specialized GCN filters. (2) Integrating these attributes poses a challenge, as manually determining their importance is difficult. Thus we introduce a two-stage attention mechanism that adaptively integrates the attributes. It utilizes a pairwise attention mechanism to determine the significance of relevance and dissimilarity between by item pairs, followed by a self-attention mechanism that integrates the attributes independently. (3) Finally, we optimize our model using contrastive learning. In the following sections, we provide details of each module.

Spectral-based GCN Filters

To model the low- and mid- components of the complementary item graph, we decouple the low-frequency relevance

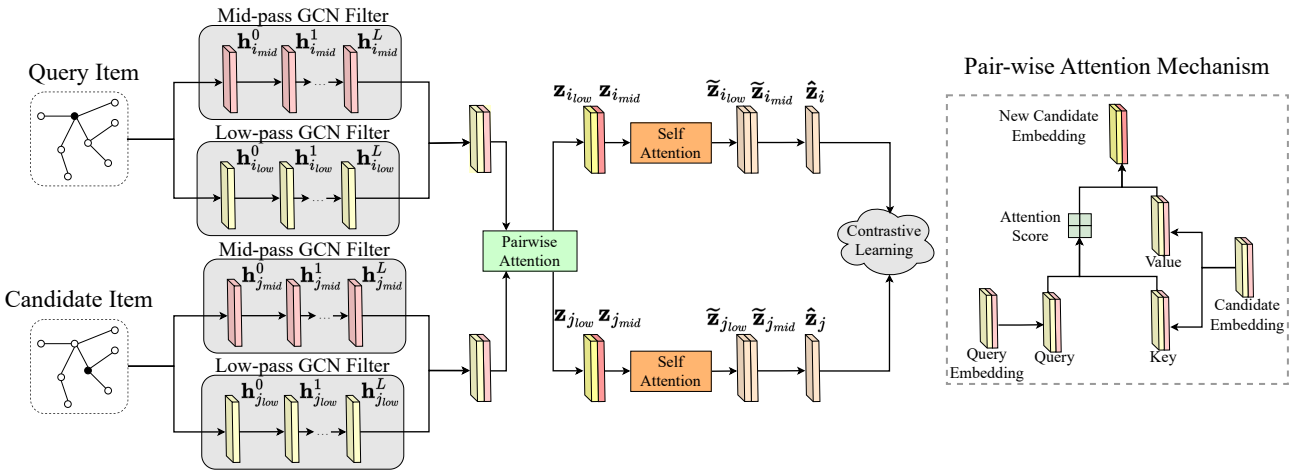

Figure 3: The overall framework of our proposed model SComGNN.

and mid-frequency dissimilarity using specialized GCN filters. We will first introduce the unified form of spatial-based and spectral-based GCNs. Based on it, we then design spectral-based low-pass and mid-pass filters and turn them into spatial forms for implementation.

Unified form of Spatial-based and Spectral-based GCNs GCNs can be explored from both spatial and spectral domain perspectives. Though the two approaches start from different domains, they can be interchanged (Balcilar et al. 2020). The GCN propagation can be formulated as:

$$\mathbf{H}^{l+1} = \sigma(\sum_{k=1}^{K} \mathbf{C}_k \mathbf{H}^l \mathbf{W}_k^l), \tag{1}$$

where σ is the activation function, K is the number of filters, $\mathbf{H}^l$ denotes the node representation at layer l, and $\mathbf{W}_k^l$ is a learnable weight matrix of the filter k at layer l. Here $\mathbf{C}_k$ is the graph convolution kernel in the spatial domain, which can be formulated in the spectral domain as:

$$\mathbf{C}_k = \mathbf{U} diag(F_k(\lambda)) \mathbf{U}^T, \tag{2}$$

where $\mathbf{U}$ and λ are the eigenvectors and the eigenvalues of the normalized graph Laplacian matrix $\mathbf{L}$. Here *diag* represents the diagonal matrix with specified elements. $F_k(\lambda)$ is the graph convolutional filter in the spectral domain and is a function with λ as the independent variable. Eq. (2) can also be formulated as:

$$F_k(\lambda) = diag^{-1}(\mathbf{U}^T \mathbf{C}_k \mathbf{U}). \tag{3}$$

The key to the spatial-based GNNs is the design of $\mathbf{C}_k$, while the key to the spectral-based GNNs is the design of $F_k(\lambda)$. With Eq. (2) and Eq. (3), these two convolutional kernels can be converted to each other. Since spatial-based GNNs are generally easier to understand and implement than spectral-based ones, Eq. (2) inspires us to design the spectral convolutional kernel $F_k(\lambda)$ first, and then convert it to a spatial form to implement it, like what existing works do (Kipf and Welling 2016; Wu et al. 2022).

Spectral-based Filters Existing spectral-based GNNs (Bruna et al. 2013; Defferrard, Bresson, and Vandergheynst 2016; Kipf and Welling 2016) design different $F_k(\lambda)$ to obtain different GNN models. Since complementary item graphs are mainly composed of low and mid-frequency components in the spectral domain, our goal is to design a low-pass and a mid-pass GCN filter to extract the low and mid-frequency components, respectively, and filter out other components. As the convolution of the spatial domain is equal to the product of the spectral domain, the larger the amplitude value, the more of the corresponding frequency component is retained. To avoid complex calculations, we design the spectral convolution kernel of the low-pass filter as a linear decreasing function of λ:

$$F_{low}(\lambda) = 1 - \lambda/2. \tag{4}$$

With Eq. (2), we can turn it into a spatial form:

$$\mathbf{C}_{low} = (\widetilde{\mathbf{A}} + \mathbf{I})/2, \tag{5}$$

where $\widetilde{\mathbf{A}} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$, denoting the normalized adjacency matrix. Linear functions can be implemented as high-pass or low-pass in increasing or decreasing form, however, it is not possible to implement a mid-pass filter, where the mid-frequency component is retained while others are filtered out. Therefore, we use the quadratic function of λ to realize the spectral convolution kernel of the mid-pass filter:

$$F_{mid}(\lambda) = -(\lambda - 1)^2 + 1. \tag{6}$$

With Eq. (2), it can also be formulated in the spatial domain:

$$C_{mid} = \mathbf{I} - \widetilde{\mathbf{A}}^2. \tag{7}$$

To better understand the low-pass and mid-pass filters, we look at both the spectral and spatial domains. For the spectral domain, we plot the spectral convolutional kernels $F_{low}(\lambda)$ and $F_{mid}(\lambda)$ (i.e., the spectrums) in Figure 4. As shown in the spectrums, the low-pass filter retains the low-frequency part and filters out other parts while the mid-pass filter retains the middle-frequency part and filters out other parts. For the spatial domain, with Eq. (5) and Eq. (7), our low-pass

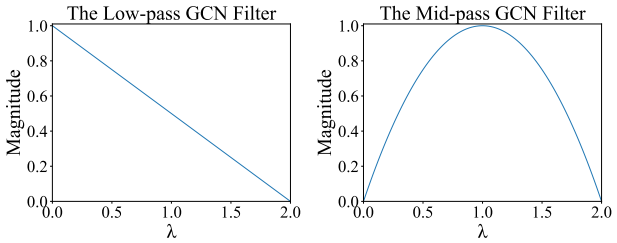

Figure 4: Spectrums of low-pass and mid-pass GCNs.

filter aggregates a node's self-information with its neighborhood information while the mid-pass filter obtains the difference between a node's self-information and its two-hop neighborhood information. Therefore, we can conclude that the low-pass filter extracts the relevance between nodes and neighbors, while the mid-pass filter extracts the dissimilarity between nodes and neighbors.

With the low-pass and mid-pass GCN filters, we can extract the low and mid-frequency components from the complementary item graph, which correspond to the relevance and dissimilarity attributes, respectively. We formulate it as:

$$\mathbf{H}_{mid}^{l} = Relu(\mathbf{C}_{mid}\mathbf{H}_{mid}^{l-1}\mathbf{W}^{l-1}), \tag{8}$$

$$\mathbf{H}_{low}^{l} = Relu(\mathbf{C}_{low}\mathbf{H}_{low}^{l-1}\mathbf{W}^{l-1}), \tag{9}$$

where $\mathbf{H}_{low}^{l}$ and $\mathbf{H}_{mid}^{l}$ are the low-frequency and mid-frequency node representation matrix at layer l, $\mathbf{W}^{l-1}$ is the weight matrix. Note that, $\mathbf{H}_{low}^{0} = \mathbf{H}_{mid}^{0} = \mathbf{X}$.

Tow-stage Attention Mechanism

Since items with complementary relationships are relevant yet dissimilar, it is challenging to determine manually which attribute is more crucial when making complementary relationship predictions. Even for the same item, the significance of these two attributes may differ in different item pairs. To tackle this problem, we propose a two-stage attention mechanism to merge the two attributes adaptively, which is composed of pairwise attention in item pairs and self-attention independently.

Pairwise Attention Mechanism. We first adopt a pairwise attention mechanism to integrate the low- and mid-frequency components of items in pairs. For example, given an item pair (v_i, v_j), item v_j determines the proportion of low-frequency relevance and mid-frequency dissimilarity of item v_i, and vice versa. With the low-pass and mid-pass GCN filters, we obtain the low and mid-frequency item representation matrix $\mathbf{H}_{low}^{L}$ and $\mathbf{H}_{mid}^{L}$, where L is the depth of propagation layers. For item v_i, we denote its embeddings as $\mathbf{h}_{i_f}^{L}$, where $f \in \{low, mid\}$. With these notations, we first introduce how the low-frequency component of item v_j selects and integrates the two embeddings of item v_i:

$$\mathbf{z}_{i_{low}} = \sum_{f} \alpha_{j_{low},i_f}\mathbf{h}_{i_f}^{L}, \quad \alpha_{j_{low},i_f} = \frac{\exp(\mathbf{h}_{j_{low}}^{L\,T}\mathbf{h}_{i_f}^{L})}{\sum_f \exp(\mathbf{h}_{j_{low}}^{L\,T}\mathbf{h}_{i_f}^{L})}, \tag{10}$$

where $\mathbf{h}_{j_{low}}^{L}$ denotes the low-frequency embedding of item v_j. Here α_{j_{low},i_f} denotes the proportion of low-frequency and mid-frequency embeddings of item v_i. $\mathbf{z}_{i_{low}}$ denotes the integrated representation of item v_i decided by low-frequency component of item v_j.

Also, the two embeddings of item v_i can be selected and integrated by the mid-frequency component of item v_j:

$$\mathbf{z}_{i_{mid}} = \sum_{f} \alpha_{j_{mid},i_f}\mathbf{h}_{i_f}^{L}, \quad \alpha_{j_{mid},i_f} = \frac{\exp(\mathbf{h}_{j_{mid}}^{L\,T}\mathbf{h}_{i_f}^{L})}{\sum_f \exp(\mathbf{h}_{j_{mid}}^{L\,T}\mathbf{h}_{i_f}^{L})}, \tag{11}$$

where $\mathbf{h}_{j_{mid}}^{L}$ denotes the mid-frequency embedding of item v_j. Here α_{j_{low},i_f} denotes the proportion of two embeddings of item v_i. $\mathbf{z}_{i_{mid}}$ denotes the integrated representation of item v_i decided by mid-frequency component of item v_j.

By the pairwise attention mechanism with item v_j, we obtain the integrated embedding $\mathbf{z}_{i_{low}}$ and $\mathbf{z}_{i_{mid}}$ of item v_i. Here the low and mid in the subscripts no longer denote the frequency component in item v_i, but rather which frequency component of item v_j it is integrated from. For item v_j in pair (v_i, v_j), we can do the same pairwise attention step to obtain $\mathbf{z}_{j_{low}}$ and $\mathbf{z}_{j_{mid}}$.

Self-attention Mechanism. After the pairwise attention step, the low and mid-frequency components of items v_i and v_j have been selected and integrated by each other. Next, we use the self-attention mechanism to further adaptively integrate the low and mid-frequency components by themselves. Similar to the above pairwise attention step, two embeddings of item v_i can be selected and integrated by $\mathbf{z}_{i_{low}}$:

$$\widetilde{\mathbf{z}}_{i_{low}} = \sum_{f} \beta_{i_{low},i_f}\mathbf{z}_{i_f}, \quad \beta_{i_{low},i_f} = \frac{\exp(\mathbf{z}_{i_{low}}^{T}\mathbf{z}_{i_f})}{\sum_f \exp(\mathbf{z}_{i_{low}}^{T}\mathbf{z}_{i_f})}, \tag{12}$$

where β_{i_{low},i_f} denotes the proportion of two embeddings of item v_i, $\widetilde{\mathbf{z}}_{i_{low}}$ denotes the further integrated representation of item v_i by $\mathbf{z}_{i_{low}}$ after the self-attention step. Also, the two embeddings of v_i can be selected and integrated by $\mathbf{z}_{i_{mid}}$:

$$\widetilde{\mathbf{z}}_{i_{mid}} = \sum_{f} \beta_{i_{mid},i_f}\mathbf{z}_{i_f}, \quad \beta_{i_{mid},i_f} = \frac{\exp(\mathbf{z}_{i_{mid}}^{T}\mathbf{z}_{i_f})}{\sum_f \exp(\mathbf{z}_{i_{mid}}^{T}\mathbf{z}_{i_f})}, \tag{13}$$

where β_{i_{mid},i_f} denotes the proportion of two embeddings of v_i, $\widetilde{\mathbf{z}}_{i_{mid}}$ denotes the further integrated representation of item v_i decided by $\mathbf{z}_{i_{mid}}$.

By the self-attention mechanism, we obtain the further integrated embedding $\widetilde{\mathbf{z}}_{i_{low}}$ and $\widetilde{\mathbf{z}}_{i_{mid}}$ of item v_i. Finally, we concat the two embeddings and turn them into a low-dimensional representation:

$$\hat{\mathbf{z}}_i = ([\widetilde{\mathbf{z}}_{i_{low}} \oplus \widetilde{\mathbf{z}}_{i_{mid}}])\mathbf{W}, \tag{14}$$

where $\mathbf{W} \in \mathbb{R}^{2d' \times d'}$ and d' is the embedding size. For item v_j, we do the same step to obtain $\hat{\mathbf{z}}_j$.

Contrastive Learning Optimization

We treat the graph complementary item recommendation as a link prediction problem and follow the principle of contrastive learning (He et al. 2020) to construct positive and negative samples for each item, which encourages the model to pull together items that are complementary and push apart

Method	Datasets	Appliances			Toys			Grocery			Home		
	Metric	HR@5	HR@10	NDCG	HR@5	HR@10	NDCG	HR@5	HR@10	NDCG	HR@5	HR@10	NDCG
Baselines	GIN	0.4347	0.6226	0.4279	0.5242	0.7408	0.4866	0.4344	0.6107	0.4425	0.4843	0.6552	0.4681
	GraghSage	0.4402	0.6574	0.4215	0.5313	0.7514	0.4863	0.6255	0.8000	0.5359	0.7272	0.8417	0.6061
	Popularity	0.2040	0.3208	0.2906	0.1809	0.2816	0.2914	0.2556	0.3690	0.3392	0.2263	0.3505	0.3223
	DCF	0.3630	0.5366	0.3817	0.4876	0.6714	0.4661	0.5991	0.7574	0.5326	0.6846	0.7798	0.6015
	P-Companion	0.3545	0.5414	0.3759	0.4098	0.6017	0.3923	0.4943	0.6774	0.4152	0.5847	0.7220	0.5145
	ALCIR	0.3754	0.5394	0.3792	0.3930	0.5959	0.3994	0.5067	0.6892	0.4614	0.5411	0.6885	0.4826
Ours	SComGNN	**0.4919**	**0.7127**	**0.4377**	**0.6561**	**0.8589**	**0.5501**	**0.7207**	**0.8565**	**0.5959**	**0.7943**	**0.8789**	**0.6610**

Table 2: Performance comparison on four datasets.

those that are not. For each item, positive samples are its complementary items and negative samples are randomly sampled from nodes that do not have links to it. The loss function can be formally defined as :

$$\mathcal{L} = -\sum_{e_{i,+}\in\mathcal{E}} \log \frac{\exp(\hat{\mathbf{z}}_i^T \hat{\mathbf{z}}_+/\tau)}{\sum_{j=0}^{M} \exp(\hat{\mathbf{z}}_i^T \hat{\mathbf{z}}_j/\tau)}, \tag{15}$$

where $\hat{\mathbf{z}}_+$ is the positive sample, M is the number of the negative items, and τ is a temperature hyperparameter. Note that, the value of $\hat{\mathbf{z}}_i$ is not fixed, instead, it changes as the item pair changes. In the inference phase, we use the representations generated from the trained model to predict whether two items are complementary. The prediction score can be computed by:

$$s_{i,j} = Sigmoid(\hat{\mathbf{z}}_i^T \hat{\mathbf{z}}_j). \tag{16}$$

The algorithm and detailed time complexity analysis can be found in the supplementary files, where the time complexity of spectral-based GCN filters and the two-stage attention mechanism is $\mathcal{O}(3|\mathcal{E}|)$ and $\mathcal{O}(8|d'|)$, respectively.

Experiments

In this section, we carry out comprehensive experiments to demonstrate the effectiveness of our method.

Experimental Setup

Datasets Following (Liu et al. 2020; Hao et al. 2020; Bibas, Shalom, and Jannach 2023), we use publicly available benchmark datasets from Amazon. We consider "also-bought" as complementary relationships, and our task is to realize the link prediction on the complementary item graphs. We select four datasets: "Appliances", "Grocery", "Toys", and "Home", and use the categories and price of each item as features. For categories, we choose BERT (Vaswani et al. 2017) as the pre-trained model to obtain the category embedding, and for the price, we discretize the continuous price to bins using equal-depth binning. Similar to previous work (Liu et al. 2020), for each item, we randomly sample one edge for constructing the test data, one for the validation data, and use the remaining edges as the training data. The statistics of the datasets are shown in Table 1.

Baselines and Implementation. The baseline models can be divided into two groups: traditional GNNs and complementary item recommendation models. The first group includes GIN (Xu et al. 2018) and GraphSage (Hamilton, Ying, and Leskovec 2017). The second group includes Popularity (Bibas, Shalom, and Jannach 2023), DCF (Galron et al. 2018), P-Companion (Hao et al. 2020), and ALCIR (Bibas, Shalom, and Jannach 2023). We exclude DecGCN (Liu et al. 2020) and EMRIGCN (Chen et al. 2023) from our comparison since they incorporate substitutable relationships which are beyond the scope of this paper, as our focus is on complementary item recommendations. For our implementation, we set the embedding size and network layers of both two GCN filters to 16 and 1, respectively. We evaluated the performance using two metrics: Hit Rate (HR@K) and NDCG (Normalized Discounted Cumulative Gain), with K set to 5 and 10. Detailed descriptions of baselines and implementations can be found in the supplementary files and our code is available[1].

Overall Performance

We present our experimental results in Table 2, including the results of our model and baselines on four datasets, where the boldfaced and underlined values represent the best and the second-best performance, respectively. Based on the results, we can make the following observations:

First, SComGNN outperforms all other models on all datasets, making it a state-of-the-art model for complementary item recommendation. Specifically, for the HR@10 score, SComGNN outperforms the best baseline model by 7.8%, 14.3%, 13.3%, and 4.4% on the four datasets, respectively. On the other two metrics (i.e., HR@5 and NDCG), SComGNN achieves a similar improvement in performance. The results demonstrate the importance of leveraging both low-frequency relevance and mid-frequency dissimilarity to enhance performance.

Additionally, we observe that traditional graph-based models can also perform well, even without custom modifications for complementary relationships. In fact, some of these models outperform non-graph-based complementary item recommendation models. This highlights the powerful capabilities of GNNs in modeling relationships. Compared to GIN, Graphsage's strong performance comes from the operation on neighbor node sampling, which reduces its reliance on low-frequency components.

Ablation Study

We carry out ablation experiments to investigate the contributions of three key modules, i.e., the low-pass GCN fil-

[1]https://github.com/luohaitong/SComGNN

Dataset	Appliances		Toys		Grocery		Home	
Metric	HR@10	NDCG	HR@10	NDCG	HR@10	NDCG	HR@10	NDCG
Ours	**0.7127**	**0.4377**	**0.8589**	**0.5501**	**0.8565**	**0.5959**	**0.8789**	**0.6610**
w/o l	0.1855	0.2457	0.1365	0.2236	0.2194	0.2560	0.1745	0.2375
w/o m	0.6420	0.4235	0.7186	0.4907	0.7839	0.5389	0.8387	0.6239
w/o a	0.6766	0.4195	0.8228	0.5149	0.8034	0.5330	0.8626	0.6223

Table 3: The ablation study performance.

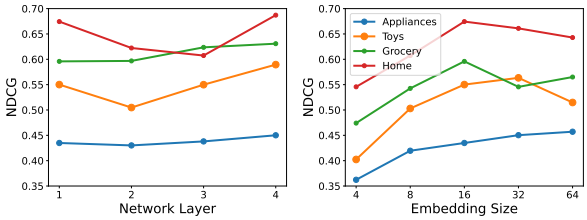

Figure 5: Hyperparameter sensitivity evaluation.

ter, the mid-pass GCN filter, and the two-stage attention mechanism. SComGNN *w/o* l is the variant without the low-frequency GCN filter and the two-stage attention mechanism, thus the model only obtains the mid-frequency representation. SComGNN *w/o* m is the variant without the mid-frequency GCN filter and the two-stage attention mechanism, thus the model only obtains the low-frequency representation. SComGNN *w/o* a is the variant without the two-stage attention mechanism thus the mid-frequency and low-frequency representations are simply concatenated. Due to similar trends and page limitations, we only show results on two metrics in Table 3, and complete results on three metrics can be found in supplementary files.

In Table 3, we can observe that (1) SComGNN achieves the best performance among the four models, indicating the collective importance of all three modules. (2) Compared to SComGNN *w/o* l, ScomGNN *w/o* m achieves better performance, which verifies our observations that the low-frequency component is greater than the mid-frequency component. (3) In some cases, the performance of SComGNN *w/o* a can even be inferior to that of SComGNN *w/o* m. This indicates although the mid-frequency representation is valuable, its integration needs to be more efficient.

Hyperparameter Analysis

We investigate the impact of two hyperparameters on the performance of our model, i.e., the depth of low-pass and mid-pass GCN layers and the embedding size. Due to similar trends and page limitations, we only present the NDCG results in Figure 5, and complete results can be found in supplementary files. We can make some conclusions. First, for the depth of GCN layers, the performance may decrease as the depth of network layers increases. This is because our model aggregates structural information from different perspectives so that a one-layer network can perform well. Next, for the embedding size, 16 is the most appropriate value. Increasing the embedding size to 32 and 64 not only does not necessarily improve the model performance but also increases the model complexity and training time. Also, reducing the embedding size to 8 or 4 results in a significant drop in performance, indicating the importance of learning rich and expressive feature representations.

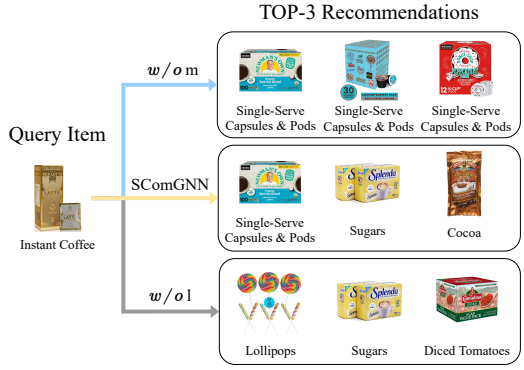

Figure 6: Examples of complementary recommendation results with different frequency components.

Case Study

To assess the impact of low and mid-frequency components in the production environment, we compare the performance of three models: SComGNN *w/o* m, SComGNN *w/o* l, and SComGNN. Figure 6 shows the TOP-3 complementary items recommended for "Instant Coffee". In the recommendations solely based on the low-frequency components, items are all coffee, which is strongly similar to the query item. Conversely, with only the mid-frequency components, items appear to have a low correlation to the query item. However, with both the low-frequency and mid-frequency components, a diverse range of items, including coffee, sugar, and cocoa, are recommended. The inclusion of sugar as a flavoring for coffee, and the presence of cocoa alongside coffee as distinct but related beverages, illustrate our ability to capture both relevance and dissimilarity. This outcome highlights the crucial roles played by the low-frequency component in representing relevance and the mid-frequency component in representing dissimilarity, both of which are essential for complementary relationships.

Conclusion

In this paper, we bridge the gap between spectral properties and attributes of complementary relationships. Our analysis reveals that complementary item graphs primarily consist of low-frequency and mid-frequency components in the spectral domain, representing relevance and dissimilarity attributes. We propose GCN filters to extract the two components and employ a two-stage attention mechanism for adaptive integration. Experiments on four publicly available datasets demonstrate the effectiveness of our theoretical analysis and the proposed method. In the future, more effective GCN filters and integration approaches of two frequency components deserve to be explored.

Acknowledgements

This work is supported in whole or in part, by National Science Foundation of China (61972381, 62106249, 62372429), Project on Cyber Security and Informatization of Chinese Academy of Sciences (CAS-WX2022SF-0401), and the Pilot for Major Scientific Research Facility of Jiangsu Province of China (NO.BM2021800).

References

Balcilar, M.; Renton, G.; Héroux, P.; Gauzere, B.; Adam, S.; and Honeine, P. 2020. Bridging the gap between spectral and spatial domains in graph neural networks. *arXiv preprint arXiv:2003.11702*.

Bibas, K.; Shalom, O. S.; and Jannach, D. 2023. Semi-supervised Adversarial Learning for Complementary Item Recommendation. *arXiv preprint arXiv:2303.05812*.

Bo, D.; et al. 2021. Beyond low-frequency information in graph convolutional networks. In *AAAI*, 3950–3957.

Bruna, J.; Zaremba, W.; Szlam, A.; and LeCun, Y. 2013. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*.

Cen, Y.; Zou, X.; Zhang, J.; Yang, H.; Zhou, J.; and Tang, J. 2019. Representation learning for attributed multiplex heterogeneous network. In *KDD*, 1358–1368.

Chen, H.; He, J.; Xu, W.; Feng, T.; Liu, M.; Song, T.; Yao, R.; and Qiao, Y. 2023. Enhanced Multi-Relationships Integration Graph Convolutional Network for Inferring Substitutable and Complementary Items. In *AAAI*, volume 37, 4157–4165.

Defferrard, M.; Bresson, X.; and Vandergheynst, P. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. *Neurips*, 29.

Galron, D. A.; Brovman, Y. M.; Chung, J.; Wieja, M.; and Wang, P. 2018. Deep item-based collaborative filtering for sparse implicit feedback. *arXiv preprint arXiv:1812.10546*.

Gao, H.; Wang, Z.; and Ji, S. 2018. Large-scale learnable graph convolutional networks. In *KDD*, 1416–1424.

Hamilton, W.; Ying, Z.; and Leskovec, J. 2017. Inductive representation learning on large graphs. *Neurips*, 30.

Han, J.; Cheng, H.; Xin, D.; and Yan, X. 2007. Frequent pattern mining: current status and future directions. *Data mining and knowledge discovery*, 15(1): 55–86.

Hao, J.; Zhao, T.; Li, J.; Dong, X. L.; Faloutsos, C.; Sun, Y.; and Wang, W. 2020. P-companion: A principled framework for diversified complementary product recommendation. In *CIKM*, 2517–2524.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 9729–9738.

He, R.; and McAuley, J. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *WWW*, 507–517.

Kipf, T. N.; and Welling, M. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Liu, Y.; Gu, Y.; Ding, Z.; Gao, J.; Guo, Z.; Bao, Y.; and Yan, W. 2020. Decoupled graph convolution network for inferring substitutable and complementary items. In *CIKM*, 2621–2628.

McAuley, J.; Pandey, R.; and Leskovec, J. 2015. Inferring networks of substitutable and complementary products. In *KDD*, 785–794.

Meng, X.; Wang, S.; Shu, K.; Li, J.; Chen, B.; Liu, H.; and Zhang, Y. 2018. Personalized Privacy-Preserving Social Recommendation. In *AAAI*, 3796–3803.

Tang, J.; Li, J.; Gao, Z.; and Li, J. 2022. Rethinking graph neural networks for anomaly detection. In *ICML*, 21076–21089.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Neurips*, 30.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Wang, J.; Sarwar, B.; and Sundaresan, N. 2011. Utilizing related products for post-purchase recommendation in e-commerce. In *Proceedings of the fifth ACM conference on Recommender systems*, 329–332.

Wang, Z.; Jiang, Z.; Ren, Z.; Tang, J.; and Yin, D. 2018. A path-constrained framework for discriminating substitutable and complementary products in e-commerce. In *WSDM*, 619–627.

Wu, L.; Zhou, Y.; and Zhou, D. 2022. Towards high-order complementary recommendation via logical reasoning network. In *ICDM*, 1227–1232.

Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; and Philip, S. Y. 2020. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1): 4–24.

Wu, Z.; Pan, S.; Long, G.; Jiang, J.; and Zhang, C. 2022. Beyond low-pass filtering: Graph convolutional networks with automatic filtering. *IEEE Transactions on Knowledge and Data Engineering*.

Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2018. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*.

Yao, Y.; and Harper, F. M. 2018. Judging similarity: a user-centric study of related item recommendations. In *RecSys*, 288–296.

Zhu, H.; Tang, X.; Zhao, T.; and Wang, S. 2023. You Need to Look Globally: Discovering Representative Topology Structures to Enhance Graph Neural Network. In *PAKDD*, 40–52.