Don't Lose Yourself: Boosting Multimodal Recommendation via Reducing Node-neighbor Discrepancy in Graph Convolutional Network

Zheyu Chen[1†], Jinfeng Xu[2†], Haibo Hu[1*]
The Hong Kong Polytechnic University[1], The University of Hong Kong[2]
zheyu.chen@connect.polyu.hk, jinfeng@connect.hku.hk, haibo.hu@polyu.edu.hk

***Abstract*—The rapid expansion of multimedia contents has led to the emergence of multimodal recommendation systems. It has attracted increasing attention in recommendation systems because its full utilization of data from different modalities alleviates the persistent data sparsity problem. As such, multimodal recommendation models can learn personalized information about nodes in terms of visual and textual. To further alleviate the data sparsity problem, some previous works have introduced graph convolutional networks (GCNs) for multimodal recommendation systems, to enhance the semantic representation of users and items by capturing the potential relationships between them. However, adopting GCNs inevitably introduces the over-smoothing problem, which make nodes to be too similar. Unfortunately, incorporating multimodal information will exacerbate this challenge because nodes that are too similar will lose the personalized information learned through multimodal information. To address this problem, we propose a novel model that retains the personalized information of ego nodes during feature aggregation by Reducing Node-neighbor Discrepancy (RedN$^{\mathbf{n}}$D). Extensive experiments on three public datasets show that RedN$^{\mathbf{n}}$D achieves state-of-the-art performance on accuracy and robustness, with significant improvements over existing GCN-based multimodal frameworks.**

***Index Terms*—Multimodal, Recommendation, Graph Collaborative Filtering, Contrastive Learning.**

I. Introduction

The rapid expansion of the internet has led to significant information overload, which recommender systems aim to address by predicting user preferences [1], [2]. Although conventional recommender systems have been developed over many years [3], the inherent problem of data sparsity still challenges them. To address the data sparsity problem, numerous works have utilized multimodal fusion techniques to enrich the semantic representations of users and items [4]–[6]. Moreover, graph convolutional networks (GCNs) have been integrated into these models [7]–[9] to capture latent relationships between users and items, substantially improving their representational capabilities. Building on these advances, multimodal recommender systems also incorporate GCNs to enrich semantic information [10]. Additionally, recent works have explored the potential of hyper-graph structures [11] and diffusion models [12] in multimodal recommendation.

However, adopting GCNs inevitably introduces the over-smoothing problem, which lead to nodes to be too similar. And incorporating multimodal information will exacerbate this challenge, because GCNs aggregate information from neighbor nodes, resulting in feature uniformity. Consequently, nodes with numerous similar interactions are treated as almost identical nodes, causing a loss of their personalization. In other words, the over-smoothing problem will cause the model to be unable to fully utilize multimodal information, resulting in suboptimal recommendation performance.

To address this problem, we propose a novel framework named **RedN$^{\mathbf{n}}$D**, which retains the personalization of ego nodes by **Red**ucing **N**ode-**n**eighbor **D**iscrepancy. We refer to the ego nodes with numerous same or similar neighbor nodes as approximate nodes. This method aligns the ego node's neighbors with the ego node, leading to differentiated representations for the same or similar neighbors shared by approximate nodes. It enhances the similarity between the ego node and its neighbors, so that the neighbor representation also has the information of the ego node, increasing the attention of this information in the aggregation process, thereby retaining personalization. This strategy prevents node uniformity, and then alleviates the over-smoothing problem. From the representation perspective, our model reduces excessive clustering of nodes in the feature space, ensuring node representations remain diverse. From the preference perspective, our model allows users to retain their personalization and unique features, reducing the tendency to imitate similar users. Regarding the challenge of this framework: each layer is represented by a representation with the same dimension in GCN, and as layers increase, neighbor nodes have multiple representations, while the ego node has only one. Assigning equal weight to both will lead to neighbors receiving more attention. To address this, we average the representations of neighbor nodes across layers to balance attention between the ego node and its neighbors.

In summary, our key contributions are as follows:

- We introduce RedN$^{\mathrm{n}}$D, a GCN-based multimodal framework that mitigates the over-smoothing problem.
- We propose a novel alignment task between ego nodes and their neighbors to retain the personalization of ego nodes.

*Corresponding Author †Equal Contribution.

This work was supported by the National Natural Science Foundation of China (Grant No: 62072390, and 92270123), and the Research Grants Council, Hong Kong SAR, China (Grant No: 15203120, 15226221, 15209922, and 15210023).

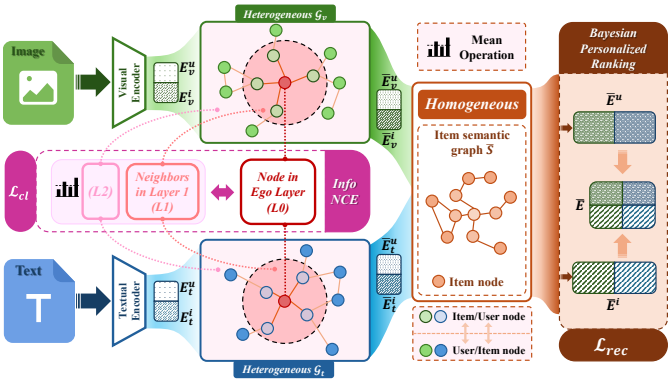


Fig. 1: Overall architecture of our RedN$^{\text{n}}$D.

- We conduct extensive experiments on three popular datasets, and provide visualization via t-SNE. These results demonstrate the effectiveness and efficiency of RedN$^{\text{n}}$D.

II. Methodology

In this section, we describe each component in RedN$^{\text{n}}$D, and the overall architecture of RedN$^{\text{n}}$D is shown in Figure 1.

A. Preliminary

Let $\mathcal{U} = \{u\}$ denotes the user set and $\mathcal{I} = \{i\}$ denotes the item set. Then, we denote the features of each modality as $E_m = \text{Con}(E_m^u, E_m^i) \in \mathbb{R}^{d_m \times (|\mathcal{U}|+|\mathcal{I}|)}$, where $m \in \mathcal{M}$ is the modality. This paper considers visual and textual modalities, denoted as $\mathcal{M} = \{v, t\}$. d_m is the dimension of the features, and $\text{Con}(\cdot)$ denotes concatenation operation.

B. Multimodal Information Encoder

Some previous works [10], [13], [14] find that both the user-item heterogeneous graph and the item-item homogeneous graph can significantly improve the performance of multimodal recommendations. Inspired by them, we propose a multimodal information encoder component to extract the representation of each modality.

Heterogeneous Graph. To capture high-order modality-specific features, we construct two **user-item graphs** $\mathcal{G} = \{\mathcal{G}_m \mid m \in \mathcal{M}\}$. Each graph $\mathcal{G}_m$ maintains the same graph structure and only retains the node features associated with each modality. Formally, the message propagation at l-th graph convolution layer can be formulated as:

$$E_m^u(l) = \sum_{i \in \mathcal{N}_u} \frac{1}{\sqrt{|\mathcal{N}_u|}\sqrt{|\mathcal{N}_i|}} E_m^i(l-1), \tag{1}$$

$$E_m^i(l) = \sum_{u \in \mathcal{N}_i} \frac{1}{\sqrt{|\mathcal{N}_u|}\sqrt{|\mathcal{N}_i|}} E_m^u(l-1), \tag{2}$$

where $E_m^{u/i}(l)$ represents the user or item representation in modality m at l-th graph convolution layer. $\mathcal{N}_{u/i}$ denotes the one-hop neighbors of u/i in $\mathcal{G}$. The final embedding for each modality m is calculated by element-wise summation:

$$\bar{E}_m^{u/i} = \sum_{l=0}^{L} E_m^{u/i}(l), \tag{3}$$

where L is the number of user-item graph layers.

Homogeneous Graph. We use k-NN to establish the **item-item graph** based on the item features for each modality m to extract significant semantic relations between items. Particularly, we calculate the similarity score $S_m^{i,i'}$ between item pair $(i, i') \in \mathcal{I}$ by the cosine similarity $\text{Sim}(\cdot)$ on their modality original features f_m^i and $f_m^{i'}$.

$$S_m^{i,i'} = \text{Sim}(f_m^i, f_m^{i'}) = \frac{(f_m^i)^\top f_m^{i'}}{\|f_m^i\| \, \|f_m^{i'}\|}. \tag{4}$$

We only retain the top-k neighbors:

$$\bar{S}_m^{i,i'} = \begin{cases} S_m^{i,i'} & \text{if } S_m^{i,i'} \in \text{ top-}k(S_m^{i,h} \mid h \in \mathcal{I}) \\ 0 & \text{otherwise} \end{cases}, \tag{5}$$

where $\bar{S}_m^{i,i'}$ represents the edge weight between item i and item i' within modality m. Thereafter, we further build a unified item-item graph $\bar{S}$ by aggregating all modality-specified graphs $\bar{S}_m$:

$$\bar{S} = \sum_{m \in \mathcal{M}} \alpha_m \bar{S}_m. \tag{6}$$

Inspired by [10], we freeze each item-item graph after initialization to remove the computation costs of the item-item graph during the training phase. Moreover, the α_m is a trainable parameter, which is initialized with equal value for each modality.

C. Multimodel Fusion

We calculate the entire user and item representations:

$$\bar{E}^{u/i} = \text{Con}(\beta_m \bar{E}_m^{u/i} \mid m \in \mathcal{M}), \tag{7}$$

where the attention weight β_m is a trainable parameter, which is initialized with equal value for each modality. Then we enhance $\bar{E}^i$ based on $\bar{S}$, the final item-item graph embedding, and fuse visual and textual modalities:

$$\bar{E}^i = \bar{E}^i + \bar{S} \cdot \bar{E}^i, \quad \bar{E} = \text{Con}(\bar{E}^u, \bar{E}^i). \tag{8}$$

D. Node-neighbor Discrepancy Reduction

Equation 3 shows that as the number of layers in the GCN increases, more layers will be aggregated, making nodes with numerous similar interactions are treated as almost identical nodes. This aggregating will lead to feature uniformity, causing the over-smoothing problem. To address this problem, our proposed method retain the personalization of ego nodes by reducing the discrepancy between ego nodes and their neighbors.

We also consider that the average strategy will give the neighbor's representation the same attention as the ego node so that the alignment operation is fair and reasonable. Through the analysis above, we obtain $\tilde{e}_m^n$ by averaging the representation of each convolution layer l, except the ego layer representation $\hat{e}_m^n$ for modality m:

$$\hat{e}_m^n = e_m^n(0), \quad \tilde{e}_m^n = \frac{1}{L} \sum_{l=1}^{L} e_m^n(l), \tag{9}$$

where L is the total number of convolution layers.

We employ contrastive learning adopting the InfoNCE [15] loss function to align the neighbors of the ego node with the ego node. Formally:

$$\mathcal{L}_{cl}^m = -\sum_{n \in \mathcal{N}} \log \frac{\exp\left(\hat{e}_m^{n\top} \tilde{e}_m^n / \tau\right)}{\sum_{n' \in \mathcal{N}} \exp\left(\hat{e}_m^{n\top} \tilde{e}_m^{n'} / \tau\right)}, \tag{10}$$

$$\mathcal{L}_{cl} = \sum_{m \in \mathcal{M}} \mathcal{L}_{cl}^{m}. \tag{11}$$

It enhances the similarity between the neighbors and the ego nodes involved in the aggregation process, thereby strengthening the aggregated information while preserving the unique features of the ego nodes. This strategy effectively prevents node uniformity and mitigates the over-smoothing problem.

E. Optimization

We adopt LightGCN [7] as the backbone model and employ the Bayesian Personalized Ranking (BPR) loss [3] as the primary optimization objective. The BPR loss is specifically designed to improve the predicted preference distinction between positive and negative items for each triplet $(u, p, n) \in \mathcal{D}$, where $\mathcal{D}$ represents the training dataset. In this context, the positive item p is one with which user u has interacted, while the negative item n is randomly selected from the set of items that user u has not interacted with. Formally:

$$\mathcal{L}_{rec} = \sum_{(u,p,n) \in \mathcal{D}} -\log(\sigma(y_{u,p} - y_{u,n})) + \lambda \cdot \|\mathbf{\Theta}\|_2^2, \tag{12}$$

where σ represents the sigmoid function, and λ controls the strength of L_2 regularization, and $\mathbf{\Theta}$ denotes the parameters subject to regularization. The terms $y_{u,p}$ and $y_{u,n}$ correspond to the ratings of user u for the positive item p and the negative item n, respectively, computed as $e_u^T \cdot e_p$ and $e_u^T \cdot e_n$. The final loss function is given by:

$$\mathcal{L} = \mathcal{L}_{rec} + \lambda_c \mathcal{L}_{cl}, \tag{13}$$

where λ_c is the balancing hyper-parameter.

III. Experiment

In this section, we conduct comprehensive experiments to evaluate the performance of our RedN$^{\text{n}}$D on three widely used real-world datasets. The following four questions can be well answered through experiment results: **RQ1**: How does our RedN$^{\text{n}}$D model compare to state-of-the-art recommendation models in terms of accuracy? **RQ2**: What impact do the key components of our RedN$^{\text{n}}$D model have on its overall performance? **RQ3**: How do different hyper-parameters affect the results achieved by our RedN$^{\text{n}}$D model? **RQ4:** Does our RedN$^{\text{n}}$D alleviate the over-smoothing problem?

A. Datasets and Evaluation Metrics

To assess the performance of our proposed RedN$^{\text{n}}$D in the recommendation task, we perform comprehensive experiments on three widely used Amazon datasets [16]: Baby, Sports, and Office. These datasets offer both product descriptions and images. In line with prior works [17], we preprocess the raw data with a 5-core setting for both items and users. Additionally, we utilize pre-extracted 4096-dimensional visual features and obtain 384-dimensional textual features using a pre-trained sentence transformer [18]. For a fair evaluation, we employ two widely recognized metrics: Recall@K (R@K) and NDCG@K (N@K). We present the average metrics for all users in the test dataset for both K = 10 and K = 20. We adhere to the standard procedure [10] with a random data split of 8:1:1 for training, validation, and testing.

TABLE I: Statistics of datasets.

Datasets	#Users	#Items	#Interactions	Sparsity
Baby	19,445	7,050	160,792	99.88%
Sports	35,598	18,357	296,337	99.95%
Office	4,905	2,420	53,258	99.55%

B. Baselines and Experimental Settings

To evaluate the effectiveness of our proposed RedN$^{\text{n}}$D, we compare it with state-of-the-art recommendation models, categorized into two groups: conventional recommendation models (**MF-BPR** [3], **LightGCN** [7], **SimGCL** [19], and **LayerGCN** [9]) and multimodal recommendation models (**VBPR** [4], **MMGCN** [17], **DualGNN** [20], **LATTICE** [13], **FREEDOM** [10], **SLMRec** [21], **BM3** [22], **MMSSL** [23], **LLMRec** [24], **LGMRec** [11], and **DiffMM** [12]).

We implement RedN$^{\text{n}}$D and all baseline models using MMRec [18]. For the general configuration, we initialize the embeddings using Xavier initialization [25] with a dimension of 64 and set the learning rate to 1e-4. All models are optimized using the Adam optimizer [26]. To ensure a fair comparison, we conduct a comprehensive grid search for each baseline according to the settings specified in their respective papers. For RedN$^{\text{n}}$D, we perform a grid search over the regularization hyper-parameter λ in {1e-2, 1e-3, 1e-4}, the balancing hyper-parameter λ_c in {1e-2, 1e-3, 1e-4}, and the value of k in {5, 10, 15, 20} for top-k in constructing the item-item graph. Early stopping is set to 20 epochs to ensure convergence. In line with [18], we update the best record based on Recall@20 on the validation dataset.

C. Overall Performance (RQ1)

Detailed experiment results are shown in Table II. The optimal results are highlighted in bold, while the suboptimal ones are underlined. Based on these results, we observed that our RedN$^{\text{n}}$D outperforms the strongest baselines, achieving 3.76%(R@10), 4.63%(R@20) improvement on the Baby dataset, 6.95%(R@10), 4.96%(R@20) improvement on the Sports dataset, and 10.78%(R@10), 9.90%(R@20) improvement on the Office dataset, which demonstrates the effectiveness of our RedN$^{\text{n}}$D.

D. Ablation Study (RQ2)

Table III highlights the impact of various excitation strategies. RedN^{1}D means only one neighbor layer of the ego node, and RedN^{2}D, RedN^{3}D, RedN^{4}D means the ego node has two, three, or four neighbor layers. This ablation study demonstrates that the choice of neighbor layer number significantly influences the model's representation capability and performance. Within the experimental range, the three-layer has the best performance.

E. Hyper-parameter Analysis (RQ3)

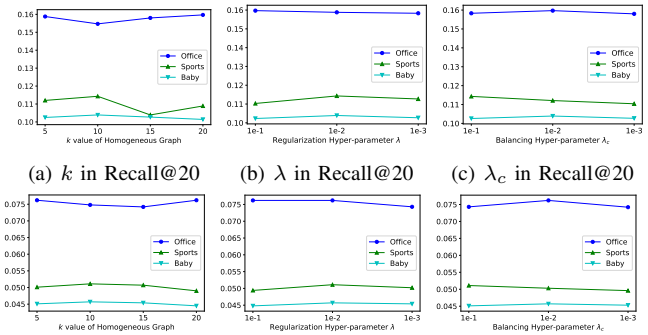
To examine the sensitivity of RedN$^{\text{n}}$D to hyper-parameters, we evaluated its performance on three datasets with different hyper-parameter values. Figure 2(a) and Figure 2(d) indicate that the optimal k-value for constructing the homogeneous graph is 10 for the Baby and Sports datasets, whereas 20 is

TABLE II: Performance comparison of baselines and RedN^{n}D(our) in terms of Recall@K(R@K) and NDCG@K(N@K). The superscript * indicates the improvement is statistically significant where the p-value is less than 0.01.

Model (Source)	Baby				Sports				Office			
	R@10	R@20	N@10	N@20	R@10	R@20	N@10	N@20	R@10	R@20	N@10	N@20
MF-BPR (UAI'09)	0.0357	0.0575	0.0192	0.0249	0.0432	0.0653	0.0241	0.0298	0.0572	0.0951	0.0331	0.0456
LightGCN (SIGIR'20)	0.0479	0.0754	0.0257	0.0328	0.0569	0.0864	0.0311	0.0387	0.0791	0.1189	0.0459	0.0583
SimGCL (SIGIR'22)	0.0513	0.0804	0.0273	0.0350	0.0601	0.0919	0.0327	0.0414	0.0799	0.1239	0.0470	0.0595
LayerGCN (ICDE'23)	0.0529	0.0820	0.0281	0.0355	0.0594	0.0916	0.0323	0.0406	0.0825	0.1213	0.0486	0.0593
VBPR (AAAI'16)	0.0423	0.0663	0.0223	0.0284	0.0558	0.0856	0.0307	0.0384	0.0692	0.1084	0.0422	0.0531
MMGCN (MM'19)	0.0378	0.0615	0.0200	0.0261	0.0370	0.0605	0.0193	0.0254	0.0558	0.0926	0.0312	0.0413
DualGNN (TMM'21)	0.0448	0.0716	0.0240	0.0309	0.0568	0.0859	0.0310	0.0385	0.0887	0.1350	0.0505	0.0631
LATTICE (MM'21)	0.0547	0.0850	0.0292	0.0370	0.0620	0.0953	0.0335	0.0421	0.0969	0.1421	0.0562	0.0686
FREEDOM (MM'23)	0.0627	0.0992	0.0330	0.0424	0.0717	0.1089	0.0385	0.0481	0.0974	0.1445	0.0549	0.0669
SLMRec (TMM'22)	0.0529	0.0775	0.0290	0.0353	0.0663	0.0990	0.0365	0.0450	0.0790	0.1252	0.0475	0.0599
BM3 (WWW'23)	0.0564	0.0883	0.0301	0.0383	0.0656	0.0980	0.0355	0.0438	0.0715	0.1155	0.0415	0.0533
MMSSL (WWW'23)	0.0613	0.0971	0.0326	0.0420	0.0673	0.1013	0.0380	0.0474	0.0794	0.1273	0.0481	0.0610
LLMRec (WSDM'24)	0.0621	0.0983	0.0324	0.0422	0.0682	0.1000	0.0363	0.0459	0.0809	0.1299	0.0492	0.0621
LGMRec (AAAI'24)	0.0639	0.0989	0.0337	0.0430	0.0719	0.1068	0.0387	0.0477	0.0959	0.1402	0.0514	0.0663
DiffMM (MM'24)	0.0623	0.0975	0.0328	0.0411	0.0671	0.1017	0.0377	0.0458	0.0733	0.1183	0.0439	0.0560
RedN^{n}D (Our)	**0.0663***	**0.1039***	**0.0361***	**0.0457***	**0.0769***	**0.1143***	**0.0409***	**0.0511***	**0.1079***	**0.1597***	**0.0622***	**0.0762***
Improv.	*3.76%*	*4.63%*	*7.12%*	*6.28%*	*6.95%*	*4.96%*	*5.68%*	*6.24%*	*10.78%*	*10.52%*	*10.68%*	*11.08%*

TABLE III: Ablation Study.

Variant	Baby		Sports		Office	
	R@20	N@20	R@20	N@20	R@20	N@20
RedN^{1}D	0.1011	0.0438	0.1098	0.0487	0.1576	0.0734
RedN^{2}D	0.1027	0.0450	0.1113	0.0496	0.1581	0.0757
RedN^{3}D	**0.1039**	**0.0457**	**0.1143**	**0.0511**	**0.1597**	**0.0762**
RedN^{4}D	0.1031	0.0452	0.1137	0.0506	0.1593	0.0758



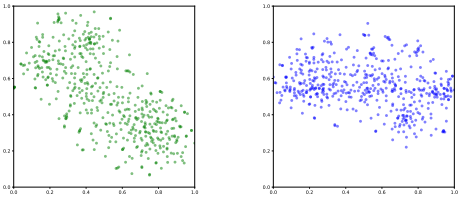
(a) k in Recall@20 (b) λ in Recall@20 (c) λ_c in Recall@20

(d) k in NDCG@20 (e) λ in NDCG@20 (f) λ_c in NDCG@20

Fig. 2: Effect of hyper-parameters: k, λ and λ_c.

preferred for the Office dataset. Additionally, Figure 2(b) and Figure 2(e) reveal that setting λ to 1e-2 yields the best results for the Baby and Sports datasets. In contrast, the Office dataset shows optimal performance at 1e-1, reflecting a different trend compared to the other two datasets. Furthermore, as illustrated in Figure 2(c) and Figure 2(f), Office and Baby datasets share the same optimal value of 1e-2 for λ_c, while Sports achieves the best performance at 1e-1.

The Office dataset achieves the best results with a larger k value, indicating that utilizing more neighbors to construct the item-item graph benefits this dataset. This is attributed to the Office dataset's low sparsity, providing more neighbors with similar semantics, thereby improving the effectiveness of the item-item graph. The hyper-parameters λ control the strength of L_2 regularization, and λ_c balancing the attention allocation between self-supervision task and recommendation



(a) RedN^{n}D. (b) RedN^{n}D w/o CL.

Fig. 3: Visualization via t-SNE.

task. It is worth noting that being flexible in choosing the value of hyper-parameters will allow us to adapt our model to multiple datasets. Although the optimal settings of λ and λ_c vary, the performance differences are minimal, demonstrating the robustness and stability of RedN^{n}D on different data sets.

F. Visualization (RQ4)

Figure 3 visualizes the final 2D embeddings of the optimal model using t-SNE [27]. The green one is the final representation for RedN^{n}D, and the blue one is the final representation for RedN^{n}D without CL. The final representation of RedN^{n}D is more discrete than that of RedN^{n}D without CL, which indicates that the over-smoothing problem is alleviated.

IV. Conclusion

In this paper, we proposed RedN^{n}D, a novel framework designed to mitigate the over-smoothing problem in GCN-based multimodal recommendation systems. By reducing the discrepancy between ego nodes and their neighbors, RedN^{n}D retains the personalization of ego nodes. Comprehensive experiments conducted on three widely used datasets validate the effectiveness of RedN^{n}D, demonstrating significant improvements in recommendation accuracy and robustness compared to existing recommendation frameworks. These results show RedN^{n}D mitigates the over-smoothing challenge in GCN-based models, and highlight its potential in advancing the development of multimodal recommendation systems.

REFERENCES

[1] J. Xu, Z. Chen, Z. Ma, J. Liu, and E. C. Ngai, "Improving consumer experience with pre-purify temporal-decay memory-based collaborative filtering recommendation for graduate school application," *IEEE Transactions on Consumer Electronics*, 2024.

[2] J. Xu, Z. Chen, J. Li, S. Yang, H. Wang, and E. C.-H. Ngai, "Aligngroup: Learning and aligning group consensus with member preferences for group recommendation," *arXiv preprint arXiv:2409.02580*, 2024.

[3] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "Bpr: Bayesian personalized ranking from implicit feedback," *arXiv preprint arXiv:1205.2618*, 2012.

[4] R. He and J. McAuley, "Vbpr: visual bayesian personalized ranking from implicit feedback," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, no. 1, 2016.

[5] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proceedings of the 26th international conference on world wide web*, 2017, pp. 173–182.

[6] J. Chen, H. Zhang, X. He, L. Nie, W. Liu, and T.-S. Chua, "Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention," in *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*, 2017, pp. 335–344.

[7] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang, "Lightgcn: Simplifying and powering graph convolution network for recommendation," in *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, 2020, pp. 639–648.

[8] J. Xu, Z. Chen, J. Li, S. Yang, H. Wang, X. Hu, and E. C.-H. Ngai, "Fourierkan-gcf: Fourier kolmogorov-arnold network – an effective and efficient feature transformation for graph collaborative filtering," *arXiv preprint arXiv:2406.01034*, 2024.

[9] X. Zhou, D. Lin, Y. Liu, and C. Miao, "Layer-refined graph convolutional networks for recommendation," in *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 2023, pp. 1247–1259.

[10] X. Zhou and Z. Shen, "A tale of two graphs: Freezing and denoising graph structures for multimodal recommendation," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 935–943.

[11] Z. Guo, J. Li, G. Li, C. Wang, S. Shi, and B. Ruan, "Lgmrec: Local and global graph learning for multimodal recommendation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 8, 2024, pp. 8454–8462.

[12] Y. Jiang, L. Xia, W. Wei, D. Luo, K. Lin, and C. Huang, "Diffmm: Multi-modal diffusion model for recommendation," *Proceedings of the 32ed ACM International Conference on Multimedia*, 2024.

[13] J. Zhang, Y. Zhu, Q. Liu, S. Wu, S. Wang, and L. Wang, "Mining latent structures for multimedia recommendation," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 3872–3880.

[14] J. Xu, Z. Chen, S. Yang, J. Li, H. Wang, and E. C.-H. Ngai, "Mentor: Multi-level self-supervised learning for multimodal recommendation," *arXiv preprint arXiv:2402.19407*, 2024.

[15] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[16] J. McAuley, C. Targett, Q. Shi, and A. Van Den Hengel, "Image-based recommendations on styles and substitutes," in *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, 2015, pp. 43–52.

[17] Y. Wei, X. Wang, L. Nie, X. He, R. Hong, and T.-S. Chua, "Mmgcn: Multi-modal graph convolution network for personalized recommendation of micro-video," in *Proceedings of the 27th ACM international conference on multimedia*, 2019, pp. 1437–1445.

[18] X. Zhou, "Mmrec: Simplifying multimodal recommendation," *arXiv preprint arXiv:2302.03497*, 2023.

[19] J. Yu, H. Yin, X. Xia, T. Chen, L. Cui, and Q. V. H. Nguyen, "Are graph augmentations necessary? simple graph contrastive learning for recommendation," in *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, 2022, pp. 1294–1303.

[20] Q. Wang, Y. Wei, J. Yin, J. Wu, X. Song, and L. Nie, "Dualgnn: Dual graph neural network for multimedia recommendation," *IEEE Transactions on Multimedia*, 2021.

[21] Z. Tao, X. Liu, Y. Xia, X. Wang, L. Yang, X. Huang, and T.-S. Chua, "Self-supervised learning for multimedia recommendation," *IEEE Transactions on Multimedia*, 2022.

[22] X. Zhou, H. Zhou, Y. Liu, Z. Zeng, C. Miao, P. Wang, Y. You, and F. Jiang, "Bootstrap latent representations for multi-modal recommendation," in *Proceedings of the ACM Web Conference 2023*, 2023, pp. 845–854.

[23] W. Wei, C. Huang, L. Xia, and C. Zhang, "Multi-modal self-supervised learning for recommendation," in *Proceedings of the ACM Web Conference 2023*, 2023, pp. 790–800.

[24] W. Wei, X. Ren, J. Tang, Q. Wang, L. Su, S. Cheng, J. Wang, D. Yin, and C. Huang, "Llmrec: Large language models with graph augmentation for recommendation," in *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, 2024, pp. 806–815.

[25] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.

[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[27] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.