Multi-Modal Knowledge Distillation for Recommendation with Prompt-Tuning

Anonymous Author(s)

ABSTRACT

Multimedia online platforms (*e.g.*, Amazon, TikTok) have greatly benefited from the incorporation of multimedia content (*e.g.*, visual, textual, and acoustic modalities) into their personal recommender systems. These modalities provide intuitive semantics that facilitate modality-aware user preference modeling. However, two key challenges in multi-modal recommenders remain unresolved: i) The introduction of multi-modal encoders with a large number of additional parameters causes overfitting, given high-dimensional multi-modal features provided by extractors (*e.g.*, ViT, BERT). ii) As side information, media content inevitably introduces inaccuracies and redundancies, which skew the modality-interaction dependency from reflecting true user preference. To tackle these problems, we propose to simplify and empower recommenders through Multimedia Knowledge Distillation (MMKD) with the prompt-tuning that enables adaptive quality distillation. Specifically, MMKD conducts model compression through distilling u-i edge relationship and multi-modal node content from cumbersome teachers to relieve students from the additional feature reduction parameters. To bridge the semantic gap between multi-modal context and collaborative signals for empowering the overfitting teacher, soft prompt-tuning is introduced to perform student task-adaptive distillation for tackling redundancy. Additionally, to adjust the impact of inaccuracies in multimedia data, a disentangled multi-modal list-wise distillation is developed with modality-aware re-weighting mechanism. Experiments on real-world data demonstrate MMKD's superiority over existing techniques. Ablation tests confirm the effectiveness of key components. Additional tests show the efficiency and effectiveness.

ACM Reference Format:
Anonymous Author(s). 2023. Multi-Modal Knowledge Distillation for Recommendation with Prompt-Tuning. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Multimedia platforms (*e.g.*, Tiktok, Amazon) have grown in importance as tools for sharing and shopping online. Utilizing modalities (*e.g.*, soundtracks of videos, pictures of products) to identify customized user preferences for item ranking is the target of multi-modal recommendation system [5, 46]. Early works started with introducing visual content[9, 16] into Collaborative Filtering(CF)[18] framework and later works employed attention mechanism for tailored signals[35, 70]. After that, graph neural network (GNNs) methods (*e.g.*, MMGCN[64], GRCN [63]) became mainstream due to significant improvement by modeling high-order[58] relations. Based on previous progress, some multi-modal works focus on alleviating sparsity issue by constructing homogeneous graphs(*e.g.*, u-u[55], i-i[61, 74]) or introducing self-supervised tasks through joint training(*e.g.*, CLCRec [62], SLMRec [53], and MICRO [75]).

Despite the progress made in previous works, some key **issues** still remain explored for multimedia recommendation scenarios:

- **I1: Overfitting & Sparsity.** Current multimedia recommenders excel by employing advanced encoders to handle high-dimensional features from pre-trained extractors (CLIP-ViT[47], BERT[3]). The auxiliary modalities were originally intended to alleviate data sparsity, but actually inevitably lead to increased consumption [56]. For example, regarding feature extractors of Electronics (Sec. 4.1.1) dataset, the output dimension of SBERT[48] and CNNs[22] are 768 and 4,096, respectively. They are much larger than embedding dimensions of current methods[62, 64], *i.e.*, $d_m \gg d$. Retraining pre-trained models can change output dimensions, but will significantly impact performance due to different latent representations and hyperparameters. Besides, training pre-trained models demands significant computational resources and can take days to weeks on multiple GPUs. Therefore, current multi-modal works[5, 60] carry additional high-dimensional feature reduction layers. These additional parameters aggravated overfitting that already exists due to data sparsity, further increasing the difficulty of convergence and leading to suboptimal results[59].
- **I2: Noise & Semantic Gap.** As side information, multimedia content has inherent inaccuracies and redundancies when assisting user preference learning with collaborative relations. For example, a user may be attracted by a textual title, but the image content is unrelated; and the music in micro-videos might be for trends, not user preferences. Blindly relying on noisy modality data may mislead the u-i relation modeling. Besides, the multi-modal context and u-i collaborative relations are originally derived from two different distributions with a large semantic gap [60], which poses challenges in mining modality-aware user preference and even disrupts the existing sparse supervisory signals.

To cope with the above issues, we propose the following solutions: **I1:** Developing a multi-modal knowledge distillation (MMKD) recommendation framework to free the inference recommender from the additional feature reduction parameters, by using KD for model compression. This paradigm prevents overfitting while maintaining accuracy, which also boosts the critical online inference phase with fewer resources. Specifically, MMKD conducts model compression through distilling edge relationship (ranking KD, denoised modality-aware ranking KD), and node content (modality-aware embedding KD). The three types of KD respectively convey i) Pure knowledge through a modified KL divergence[29] based on BPR loss[49]; ii) Fine-grained modality-aware list-wise ranking knowledge; iii) Modality-aware embedding KD through SCE loss [20], an enhanced version of MSE. **I2:** Developing two modules to tackle issues '*Noise & Semantic Gap*' based on the KD framework: i) Semantic bridging soft prompt-tuning is meant to reduce the impact of redundancy by prompting teacher to deliver student-task adaptive knowledge. In

Conference'17, July 2017, Washington, DC, USA
2023. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Table 1: Model compression analysis. Time complexity comparison among SOTA GNN-enhanced multi-modal recommenders. i) the $\mathcal{R}(\cdot)$: time complexity of multi-modal feature reduction layer, by mapping high-dimensional features into dense embeddings, *i.e.*, $d_m \rightarrow d$. ii) the *GNNs*: time complexity of various GNN architectures in different models for message propagation.

Component	MMGCN [64]	GRCN [63]	LATTICE [74]	SLMRec [53]	BM3 [80]	MMKD
$\mathcal{R}(\cdot)$	$O(\sum_{m\in\mathcal{M}} \|\mathcal{I}\|(d_m+d)d_h)$	$O(\sum_{m\in\mathcal{M}} \|\mathcal{I}\|d_m d)$	$O(\sum_{m\in\mathcal{M}} \|\mathcal{I}\|d_m d)$	$O(\sum_{m\in\mathcal{M}} \|\mathcal{I}\|d_m d)$	$O(\sum_{m\in\mathcal{M}} \|\mathcal{I}\|d_m d)$	0
GNNs	$O(\sum_{m\in\mathcal{M}} L\|\mathcal{E}\|d^3)$	$O(\sum_{m\in\mathcal{M}} (\|\mathcal{I}\|^2 d + L\|\mathcal{E}\|d))$	$O(\sum_{m\in\mathcal{M}} \|\mathcal{I}\|^2 d_m + k\|\mathcal{I}\|log(\|\mathcal{I}\|) + L\|\mathcal{E}\|d)$	$O(\sum_{m\in\mathcal{M}} L\|\mathcal{E}\|d)$	$O(L\|\mathcal{E}\|d)$	$O(L\|\mathcal{E}\|d)$

other words, prompt-tuning module can bridge the semantic gap in two aspects: multi-modal content & collaborative signals, and student & frozen teacher. Technically, the module is incorporated into the teacher's reduction layer and constructs prompts based on multi-modal features. For optimization, the soft prompts train with both teacher and student, to adaptively guide students during the distillation when teacher is frozen. ii) Modality-aware disentangled denoising list-wise ranking KD is to adjust the influence of inaccuracies in modality-aware user preference. The decoupled KD process first separates the results of list-wise ranking based on modality-specific presentation. A re-weighting mechanism is then applied to adjust the influence of unreliable portions.

To summarize, the main contributions of this work are as follows:

- In this work, we propose a novel multi-modal KD framework MMKD for multimedia recommendation, which is able to produce a lightweight yet effective student inference recommender with minimal online inference time and resource consumption.
- We integrate prompt-tuning with multi-modal KD to bridge the semantic gap between modality content and collaborative signals. Additionally, by disentangling the modality-aware ranking logits, the impact of noise in multimedia data is adjusted.
- We conduct experiments to evaluate our model performance on real-world datasets. The results demonstrate our MMKD outperforms state-of-the-art baselines. The ablation studies and further analysis show the effectiveness of sub-modules.

To facilitate the result reproducibility, our implementation is released via link: https://anonymous.4open.science/r/MMKD.

2 PRELIMINARIES

Interaction Graph with Multi-Modal Context. Motivated by the effectiveness of graph-based recommenders, we represent user-item relationships as a bipartite graph $\mathcal{G} = (\{\mathcal{U}, \mathcal{I}\}, \mathcal{E}, \mathbf{X})$. Here, $\mathcal{U}, \mathcal{I}$ are users' set and items' set, respectively. The edges $\mathcal{E}$ in $\mathcal{G}$ can be represented by adjacency matrix $\mathbf{A} \in \mathbb{R}^{|\mathcal{U}|\times|\mathcal{I}|}$ with $\mathbf{A}_{[u,i]} = 1$ if the implicit feedback exists, otherwise $\mathbf{A}_{[u,i]} = 0$. Furthermore, each item $i \in \mathcal{I}$ is associated with multi-modal features $\mathbf{X}_i = \{\mathbf{x}_i^1, ..., \mathbf{x}_i^m, ..., \mathbf{x}_i^{|\mathcal{M}|}\}$, where $|\mathcal{M}|$ is the number of modalities, indexed by $m \in \mathcal{M}$. The feature $\mathbf{x}_i^m$ is a high-dimensional vector in $\mathbb{R}^{d_m}$ that captures the characteristics of modality m. Notably, the dimensions d_m of multimodal features are often much larger than those d of recommender representations, *i.e.*, $d_m \gg d$.

Task Formulation. The goal of multi-modal recommender systems is to learn a function that predicts the likelihood of a user adopting an item, given an interaction graph $\mathcal{G}$ with multi-modal context $\mathbf{X}$. The output of the predictive function is the learned preference score of a target user u over a non-interacted item i.

3 METHODOLOGY

MMKD conducts model compression to build a lightweight yet effective multi-modal recommender for resource-friendly online collaborative filtering. The overall model flow is shown in Fig. 1. Key components will be elaborated in following subsections.

3.1 Modality-aware Task-adaptive Modeling

3.1.1 **Teacher-Student in CF.** Knowledge distillation aims to compress a complex large model into a lightweight and effective small model. Inspired by this, our developed MMKD is to transfer modality-aware collaborative signals from cumbersome teacher to lightweight student. For optimization, we employ offline distillation [11] which is a two-stage process, for flexibility concerns. In the first stage, only the teacher is trained, and in the second stage, the teacher remains fixed while only the student is trained.

Teacher $\mathcal{T}$ follows pattern of current graph-based multi-modal encoders [60, 74], which encodes id-corresponding embeddings $\mathbf{E}_u^{\mathcal{T}}, \mathbf{E}_i^{\mathcal{T}}$ and modality-specific features $\mathbf{F}_u^m, \mathbf{F}_i^m$ through GNNs. The two types of encoded representations will be further distilled to student by our modality KD in Sec. 3.2.2, Sec. 3.2.3 and collaborative KD in Sec. 3.2.1. Teacher $\mathcal{T}$ encoding process can be as follows:

$$\{\mathbf{E}_u^{\mathcal{T}}, \mathbf{E}_i^{\mathcal{T}}\}, \quad \{\mathbf{F}_u^1, ..., \mathbf{F}_u^m, ..., \mathbf{F}_i^1, ..., \mathbf{F}_i^m ...\} = \mathcal{T}(\mathbf{A}, \mathbf{X}) \tag{1}$$

The two types of outputs respectively convey reliable collaborative signals and modality-aware user preferences to student. $\mathbf{F}_u^m \in \mathbb{R}^{|\mathcal{U}|\times d}, \mathbf{F}_i^m \in \mathbb{R}^{|\mathcal{I}|\times d}$ are compressed (*i.e.*, $d_m \rightarrow d$) from high-dimensional $\mathbf{X} \in \mathbb{R}^{|\mathcal{I}|\times d_m}$ from extractors (*e.g.*, BERT [27]).

Student $\mathcal{S}$ utilizes lightweight LightGCN [17] to capture user-item collaborative relationship. The embedding process is conducted without computationally intensive encoding of multi-modal features. The encoding of student $\mathcal{S}$ can be summarized as:

$$\mathbf{E}_u^{\mathcal{S}}, \ \mathbf{E}_i^{\mathcal{S}} = \mathcal{S}(\mathbf{A}) \tag{2}$$

$\mathbf{E}_u^{\mathcal{S}}, \mathbf{E}_i^{\mathcal{S}}$ are the final user and item presentation used for online recommendation inference and for receiving teacher knowledge.

3.1.2 **Soft Prompt-Tuning as Semantic Bridge.** Modality content $\mathbf{X}$ inevitably includes ranking task-irrelevant redundancies, which not only confuse the target CF task but also exacerbate overfitting. Besides, the large semantic gap between general-purpose modality modeling and u-i interaction modeling also hinders true user preferences. Drawing inspiration from parameter efficient finetuning (PEFT) [31, 33], we employ soft prompt-tuning[31] as the solution. Specifically, we incorporate prompt $\mathbf{p}$ into teacher $\mathcal{T}(\cdot)$'s multi-modal feature reduction layer $\mathcal{R}(\cdot)$, to facilitate the extraction of collaborative signals from modalities. $\mathbf{p}$ is constructed by multi-modal features $\mathbf{X}$ and finetuned with student $\mathcal{S}(\cdot)$ to provide the frozen teacher $\mathcal{T}(\cdot)$ a sutdent-task related signals as a hint. The specific process can be divided into three steps: i) Construct the prompt; ii) Incorporate in teacher $\mathcal{T}(\cdot)$; iii) Conduct prompt-tuning.

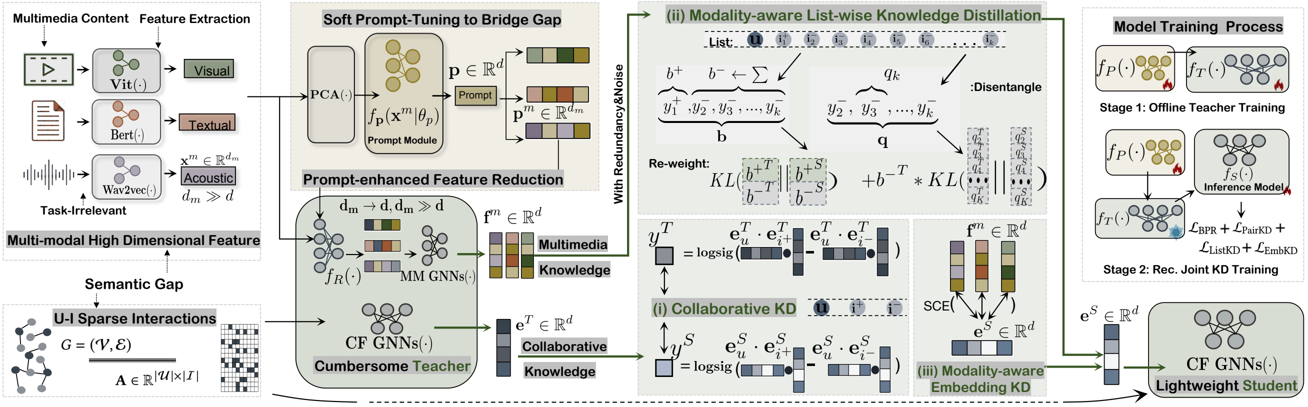
Multimedia Content
Feature Extraction
$\mathrm{Vit}(\cdot)$
Visual
$\mathrm{Bert}(\cdot)$
Textual
$\mathrm{Wav2vec}(\cdot)$
$\mathbf{x}^m \in \mathbb{R}^{d_m}$
Acoustic
$d_m \gg d$
Task-Irrelevant
Multi-modal High Dimensional Feature
Semantic Gap
U-I Sparse Interactions
$G = (\mathcal{V}, \mathcal{E})$
$\mathbf{A} \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{I}|}$
Soft Prompt-Tuning to Bridge Gap
$\mathbf{PCA}(\cdot)$
$f_{\mathbf{p}}(\mathbf{x}^m|\theta_p)$
Prompt Module
$\mathbf{p} \in \mathbb{R}^d$
Prompt
$\mathbf{p}^m \in \mathbb{R}^{d_m}$
Prompt-enhanced Feature Reduction
$\mathrm{d_m} \rightarrow \mathrm{d}, \mathrm{d_m} \gg \mathrm{d}$
$f_R(\cdot)$
MM GNNs$(\cdot)$
$\mathbf{f}^m \in \mathbb{R}^d$
Multimedia
Knowledge
CF GNNs$(\cdot)$
Cumbersome Teacher
$\mathbf{e}^T \in \mathbb{R}^d$
Collaborative
Knowledge
With Redundancy&Noise
(ii) Modality-aware List-wise Knowledge Distillation
List: $\mathbf{u}$ i_1^+ i_2^- i_3^- i_4^- i_5^- i_6^- . . . i_k^-
b^+ $b^- \leftarrow \sum$
$y_1^+, y_2^-, y_3^-, ..., y_k^-$
$\mathbf{b}$
q_k
$y_2^-, y_3^-, ..., y_k^-$
$\mathbf{q}$
:Disentangle
Re-weight:
$KL(\begin{smallmatrix} b^{+T} \\ b^{-T} \end{smallmatrix} || \begin{smallmatrix} b^{+S} \\ b^{-S} \end{smallmatrix}) + b^{-T} * KL(q_2^T, q_3^T, q_4^T, \cdots, q_K^T || q_2^S, q_3^S, q_4^S, \cdots, q_K^S)$
y^T
$= \mathrm{logsig}(\mathbf{e}_u^T \cdot \mathbf{e}_{i+}^T - \mathbf{e}_u^T \cdot \mathbf{e}_{i-}^T)$
(i) Collaborative KD
$\mathbf{u}$ $\mathbf{i}^+$ $\mathbf{i}^-$
y^S
$= \mathrm{logsig}(\mathbf{e}_u^S \cdot \mathbf{e}_{i+}^S - \mathbf{e}_u^S \cdot \mathbf{e}_{i-}^S)$
$\mathbf{f}^m \in \mathbb{R}^d$
SCE( )
$\mathbf{e}^S \in \mathbb{R}^d$
(iii) Modality-aware Embedding KD
$\mathbf{e}^S \in \mathbb{R}^d$
CF GNNs$(\cdot)$
Lightweight Student
Model Training Process
$f_P(\cdot)$
$f_T(\cdot)$
Stage 1: Offline Teacher Training
$f_P(\cdot)$
$f_S(\cdot)$
Inference Model
$f_T(\cdot)$
$\mathcal{L}_{\mathrm{BPR}} + \mathcal{L}_{\mathrm{PairKD}} + \mathcal{L}_{\mathrm{ListKD}} + \mathcal{L}_{\mathrm{EmbKD}}$
Stage 2: Rec. Joint KD Training

Figure 1: MMKD is to learn a lightweight recommender with minimal online consumption, including three types of KD: i) ranking KD; ii) denoised modality-aware ranking KD; iii) modality-aware embedding KD. Besides, prompt-tuning is for adaptive task-relevant KD; disentangling and re-weighting are introduced to adjust the impact of noise in modalities.

Prompt Construction. To better incorporate semantics to prompt module $\mathcal{P}(\cdot)$, we initialize $\mathbf{p}$ using semantic content[51], instead of vanilla initialization (*e.g.*, Xavier [10], uniform). Refer to Prefix-tuning[33], $\mathcal{P}(\cdot)$ is a feedforward layer takes soft prompt $\mathbf{p}$ as input which aggregates information from multi-modal item features $\mathbf{x}^m$. The process of obtaining prompt vectors is as follows:

$$\mathbf{p} = \mathcal{P}(\mathbf{x}^m|\theta_{\mathcal{P}}) = \mathcal{P}\left(\frac{1}{|\mathcal{M}|}\sum_{m\in\mathcal{M}}^{|\mathcal{M}|}\eta(\mathbf{x}^m)\right) \tag{3}$$

$\eta(\cdot)$ denotes the dimensionality reduction function (*e.g.*, PCA) for multi-modal features. The learned prompt $\mathbf{p}$ will be incorporated into the teacher's inference process. The soft prompt module $\mathcal{P}(\cdot)$ will offer adaptive cues to the teacher once the student $\mathcal{S}(\cdot)$ is trained and the teacher $\mathcal{T}(\cdot)$ is frozen.

Prompt-guided Teacher. Having obtained prompt $\mathbf{p}$, we apply it to the feature reduction layer $\mathcal{R}(\cdot)$ in teacher $\mathcal{T}(\cdot)$ for enhancing the overfitting teacher, while simultaneously conducting student-task adaptive knowledge distillation through the frozen teacher. To be specific, we transform our prompt $\mathbf{p}$ into modality-specific module, *i.e.*, $\mathbf{p} \longrightarrow \mathbf{p}^m$, which allows the prompt to capture modality-specific information. Next, our method leverages a simple yet effective add operator, inspired by [21], to integrate the modality-specific prompt $\mathbf{p}^m$ into the teacher's multi-modal feature encoding layer. Formally, this prompt integration process can be given as follows:

$$\mathbf{f}^m = \mathcal{R}(\mathbf{x}^m, \mathbf{p}^m|\theta_{\mathcal{R}}) = \varrho((\mathbf{x}^m + \lambda_1 * \mathbf{p}^m)\mathbf{W}_{\mathcal{R}}^m + \mathbf{b}_{\mathcal{R}}^m) \tag{4}$$

$\mathcal{R}(\cdot)$ takes high-dimensional multi-modal features $\mathbf{x}^m$ and modality-specific prompts $\mathbf{p}^m$ as inputs, and output modality-specific embeddings $\mathbf{f}^m \in \mathbb{R}^d$. The modality-specific prompt $\mathbf{p}^m \in \mathbb{R}^{d_m}$ is obtained by reshaping (*i.e.*, $d \longrightarrow d_m$) from $\mathbf{p}$ through $\mathbf{p}^m = \mathbf{p} \cdot \mathbf{p}^T\mathbf{x}^m$, and adjusted by factor λ_1. To prevent overfitting caused by numerous parameters high-dimensional features $\mathbf{x}^m$'s reduction, dropout $\varrho(\cdot)$ is applied here. The filter parameters $\mathbf{W}_{\mathcal{R}}^m$ and $\mathbf{b}_{\mathcal{R}}^m$ are used to map modality-specific features to their respective embedding space.

By this way, the feature reduction $\mathcal{R}$ will be strengthened due to: i) bridging the gap between modality content and collaborative signals, extracting modality-aware user preferences; ii) facilitating knowledge distillation process by making modality-aware student constrained prompt $\mathbf{p}$ participate in teacher's inference.

Table 2: Summary of Key Notations.

Notations	Explanations
$\mathcal{G}, \mathcal{V}, \mathcal{E}$	Interaction graph, Node set, Edge set
$\mathbf{x}^m \in \mathbb{R}^{d_m}, \mathbf{f}^m \in \mathbb{R}^d$	High dimensional/ Densified feature of $\mathcal{T}$
$\mathbf{E}_u^{\mathcal{T}} \in \mathbb{R}^{\mathcal{U}\times d}, \mathbf{E}_u^{\mathcal{S}} \in \mathbb{R}^{\mathcal{U}\times d}$	Final user embedding of teacher/student
$\mathcal{T}(\cdot), \mathcal{S}(\cdot), \mathcal{P}(\cdot), \mathcal{R}(\cdot)$	Teacher, Student, Prompt Module, Reduction
$\mathbf{b}, \mathbf{q}$	Binarized/Re-weighted knowledge
$b^+/b^-, q_k$	Binarized/Re-weighted single score

* We use uppercase bold letters (*e.g.*, $\mathbf{X}$) to denote matrices, lowercase bold letters (*e.g.*, $\mathbf{x}$) to denote vectors and light letters to denote scalar values. Algorithm pseudo code in Supplementary.

Soft Prompt-tuning Paradigm. In KD's soft prompt tuning, we consider the cumbersome teacher as the pre-trained model, and the process is split into two stages. During teacher training, the prompt module $\mathcal{P}(\cdot)$ undergoes gradient descent with teacher $\mathcal{T}(\cdot)$, affecting the teacher's inference process. During student training, we employ offline knowledge distillation[11], freezing the teacher's parameters $\theta_{\mathcal{T}}$ and updating the prompt module $\mathcal{P}(\cdot)$ again according to the student's recommended loss, which allows the prompt $\mathbf{p}$ to provide additional guidance to the feature reduction process and distill task-relevant knowledge from teacher $\mathcal{T}(\cdot)$.

3.2 Modality & Ranking Knowledge Distillation

To comprehensively obtain quality collaborative signal and modality-aware user preference from teacher $\mathcal{T}(\cdot)$, we have designed three types of KD paradigms to convey knowledge from different perspectives: i) Ranking KD; ii) Denoised Modality-aware Ranking KD; and iii) Modality-aware Embedding KD.

3.2.1 **Pure Ranking KD.** As a ranking task, teacher $\mathcal{T}(\cdot)$ ought to convey task-relevant collaborative relations. To this end, we propose to utilize prediction logits in ranking objectives such as BPR[49] for KD optimization. Specifically, we distill valid ranking knowledge from the ultimate representation $\mathbf{E}_u^{\mathcal{T}}, \mathbf{T}_i^{\mathcal{T}}$ constrained by the classical pair-wise ranking BPR loss. Pair-wise score $y_{\text{pair}}^{\mathcal{T}}$ and $y_{\text{pair}}^{\mathcal{S}}$ are is taken as a logit of KD loss for teacher and student, respectively. The classic KL loss logit represents multi-class scores, while y_{pair} represents a binary classification logit for determining whether i_+ is better than i_- for user u. Our KD paradigm with the pairwise ranking loss can be formally presented as follows:

$$\mathcal{L}_{\text{PairKD}}(\Theta_{\mathcal{S}};\Theta_{\mathcal{P}}) = -\sum_{(u,i^+,i^-)}^{|\mathcal{E}_{\text{bpr}}|} y^{\mathcal{T}}_{\text{pair}}(\log y^{\mathcal{T}}_{\text{pair}} - \log y^{\mathcal{S}}_{\text{pair}}) \quad (5)$$

$$y_{\text{pair}} = \log(\text{sigmoid}(\mathbf{e}_u \cdot \mathbf{e}_{i^+} - \mathbf{e}_u \cdot \mathbf{e}_{i^-}))$$

where $\mathcal{L}_{\text{PairKD}}$ represents the pair-wise ranking KD objective. $\theta_S; \theta_P$ means that both student $\mathcal{S}(\cdot)$ and prompt module $\mathcal{P}(\cdot)$ parameters are updated with the loss $\mathcal{L}_{\text{PairKD}}(\cdot)$. In each step, MMKD samples a batch of triplets $\mathcal{E}_{\text{bpr}} = \{(u, i^+, i^-)|\mathbf{A}_{[u,i^+]} = 1, \mathbf{A}_{[u,i^-]} = 0\}$, where u denotes the target user. Here, i^+ and i^- denotes the positive item and negative item of BPR loss, respectively. In this way, teacher model $\mathcal{T}(\cdot)$ imparts collaborative expertise to student model $\mathcal{S}(\cdot)$, offering rich implicit knowledge in a different solution space [28] to help the student escape from local optima [7, 14].

3.2.2 Denoised Modality-aware Ranking Disentangled KD.

Previously encoded multi-modal content $\mathbf{f}^m_u, \mathbf{f}^m_i$ in teacher $\mathcal{T}(\cdot)$ contains noise and can affect the modality-aware user preferences modeling. To conduct accuracy and fine-grained distillation while reducing the impact of task-irrelevant parts, we design a denoised modality-aware KD. Specifically, we calculate the list-wise score using $\mathbf{f}^m_u, \mathbf{f}^m_i$ to perform modality-aware ranking KD. In addition, to further reduce the impact of noise, we reformulate KD loss into a weighted sum of the disentangled parts.

Disentangling Modality-aware List-wise Score. For a K samples ranking list, the predicted logits can be denoted as $\mathbf{y}_{\text{list}} = [y^+_1; y^-_2, y^-_3, ..., y^-_k, ..., y^-_K]$, where y^+ and y^- are the scores of the observed edge $\mathbf{A}^+$ and unobserved edge $\mathbf{A}^-$, respectively. MMKD take each score in $\mathbf{y}_{\text{list}}$ as logit in KL divergence for distilling informative tacit knowledge[11]. The modality-aware list-wise logits then can be reformulated into two parts $KL(\mathbf{b}^{\mathcal{T}}\|\mathbf{b}^{\mathcal{S}})$ and $KL(\mathbf{q}^{\mathcal{T}}\|\mathbf{q}^{\mathcal{S}})$. $\mathbf{b}^{\mathcal{T}}$ deliver overall user preference to $\mathbf{b}^{\mathcal{S}}$; $\mathbf{q}^{\mathcal{T}}$ deliver fine-grained list-wise ranking prefer to $\mathbf{q}^{\mathcal{S}}$. More specifically, $\mathbf{b} = [b^+, b^-] \in \mathbb{R}^{1\times 2}$ represents the binary logits of observed set $\{y^+_1\}$ and unobserved set $\{y^-_2, y^-_3, ..., y^-_k, ..., y^-_K\}$, that softened by softmax:

$$\mathbf{b}: \quad b^+ = \frac{exp(y^+_1)}{\sum_{k=1}^{K} exp(y_k)}; \quad b^- = \frac{\sum_{k=2}^{K} exp(y^-_k)}{\sum_{k=1}^{K} exp(y_k)} \quad (6)$$

Note that, $\hat{y}$ is the sum of unobserved sets. Meanwhile, we declare $\mathbf{p} = [\hat{y}_2, \hat{y}_3, ..., \hat{y}_k, ..., \hat{y}_K] \in \mathbb{R}^{1\times K}$ to independently model logits among unobserved set (i.e., without considering y^+). Each element is calculated by:

$$\mathbf{q}: \quad q_k = \frac{exp(y^-_k)}{\sum_{k=2}^{K} exp(y_k)} \quad (7)$$

Re-weighting Modality-aware List-wise Score. Afterward, lower scores are assigned to those uncertain user-item relationships to down-weight their influence in the KD process. This allows MMKD to focus on the most reliable signals from the teacher model for denoised knowledge transfer. The vanilla KL-Divergence can be disentangled and re-weighted through the following derivation [1] :

$$KL(\mathbf{y}^{\mathcal{T}}_{\text{list}}\|\mathbf{y}^{\mathcal{S}}_{\text{list}}) = b^{+\mathcal{T}}\log(\frac{b^{+\mathcal{T}}}{b^{+\mathcal{S}}}) + \sum_{k=2,i\neq 1}^{K} y_k^{-\mathcal{T}}\log(\frac{y_k^{-\mathcal{T}}}{y_k^{-\mathcal{S}}}) \quad (8)$$

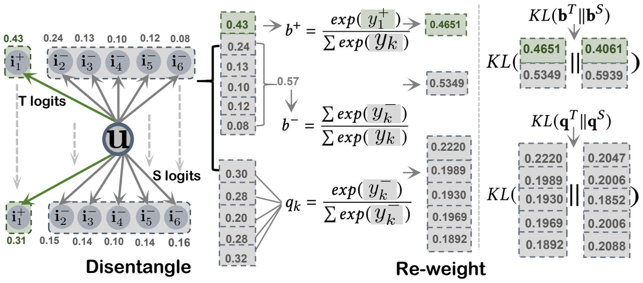
0.43
0.24
0.13
0.10
0.12
0.08
i^+_1
i^-_2
i^-_3
i^-_4
i^-_5
i^-_6
T logits
u
S logits
0.31
0.15
0.14
0.10
0.14
0.16
Disentangle
0.43
0.24
0.13
0.10
0.12
0.08
0.57
0.30
0.28
0.20
0.28
0.32
$b^+ = \frac{exp(y^+_1)}{\sum exp(y_k)}$
$b^- = \frac{\sum exp(y^-_k)}{\sum exp(y_k)}$
$q_k = \frac{exp(y^-_k)}{\sum exp(y^-_k)}$
0.4651
0.5349
0.2220
0.1989
0.1930
0.1969
0.1892
Re-weight
$KL(\mathbf{b}^T\|\mathbf{b}^S)$
KL(0.4651 0.5349 || 0.4061 0.5939)
$KL(\mathbf{q}^T\|\mathbf{q}^S)$
KL(0.2220 0.1989 0.1930 0.1969 0.1892 || 0.2047 0.2006 0.1852 0.2006 0.2088)

Figure 2: Calculation Example of Disentangled KD

According to Eq. 6 and Eq. 7, we can derive $q_k = y_k/b^-$. Thus, Eq. 8 can be rewritten as follows (detailed derivations in Appendix):

$$= b^{+\mathcal{T}}\log(\frac{b^{+\mathcal{T}}}{b^{+\mathcal{S}}}) + b^{-\mathcal{T}} \sum_{k=2,i\neq 1}^{K} q_k{}^{\mathcal{T}}(\log(\frac{q_k{}^{\mathcal{T}}}{q_k{}^{\mathcal{S}}}) + \log(\frac{b^{-\mathcal{T}}}{b^{-\mathcal{S}}})) \quad (9)$$

$$= \underbrace{b^{+\mathcal{T}}\log(\frac{b^{+\mathcal{T}}}{b^{+\mathcal{S}}}) + b^{-\mathcal{T}}\log(\frac{b^{-\mathcal{T}}}{b^{-\mathcal{S}}})}_{KL(\mathbf{b}^{\mathcal{T}}\|\mathbf{b}^{\mathcal{S}})} + (1 - b^{+\mathcal{T}}) \underbrace{\sum_{k=2,i\neq 1}^{K} q_{k}{}_{k}^{\mathcal{T}}\log(\frac{q_k{}^{\mathcal{T}}}{q_k{}^{\mathcal{S}}})}_{KL(\mathbf{q}^{\mathcal{T}}\|\mathbf{q}^{\mathcal{S}})}$$

Then, we can reformulate our disentangled knowledge distillation paradigm with the awareness of multi-modalities as follows:

$$\mathcal{L}_{\text{ListKD}} = -\sum_{m\in\mathcal{M}}^{|\mathcal{M}|} KL(\mathbf{b}^{\mathcal{T}}\|\mathbf{b}^{\mathcal{S}}) + (b^{+\mathcal{T}} - 1)KL(\mathbf{q}^{\mathcal{T}}\|\mathbf{q}^{\mathcal{S}}) \quad (10)$$

List-wise ranking KD loss $\mathcal{L}_{\text{ListKD}}$ is reformulated as a weighted sum of two terms for adjustably transferring reliable knowledge and enhancing the accuracy of modality-relevant user preference.

3.2.3 Modality-aware Embedding Distillation.

In addition to the logit-based knowledge distillation, we propose to enhance our MMKD framework with embedding-level distillation. To achieve embedding alignment in our MMKD, we employ the Scale Cosine Error (SCE) [20] loss function for robust training instead of Mean Square Error (MSE). This is because MSE is sensitive and unstable, which can lead to training collapse [20] because of varied feature vector norms and the curse of dimensionality [6]. The utilization of the SCE-based loss $\mathcal{L}_{\text{EmbKD}}$ for embedding-level knowledge distillation can take the following forms:

$$\mathcal{L}_{\text{EmbKD}} = \sum_{m\in\mathcal{M}}^{|\mathcal{M}|} \frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}}(1 - \frac{\mathbf{e}^{\mathcal{S}}_i \cdot \mathbf{f}^m_i}{\|\mathbf{e}^{\mathcal{S}}_i\| \times \|\mathbf{f}^m_i\|})^{\gamma}, \gamma \geq 1 \quad (11)$$

$\mathcal{L}_{\text{EmbKD}}$ is averaged over all user and item nodes in the interaction graph $\mathcal{G}$. The final representation outputted by the student model $\mathcal{S}(\cdot)$ is denoted as $\mathbf{e}^{\mathcal{S}}_i \in \mathbf{E}^{\mathcal{S}}_i$, while the encoded multi-modal feature from the teacher function $\mathcal{T}(\cdot)$ are denoted as $\mathbf{f}^m_i \in \mathbf{F}^m_i$. The scaling factor γ is an adjustable hyper-parameter.

3.2.4 Model Joint Training of MMKD.

We train our recommender using a multi-task learning scheme to jointly optimize MMKD with the following tasks: i) the main user-item interaction prediction task, represented by $\mathcal{L}_{\text{BPR}}$; ii) the pair-wise robust ranking KD $\mathcal{L}_{PairKD}$; iii) the modality-aware list-wise disentangled KD $\mathcal{L}_{ListKD}$; iv) modality-aware embedding KD $\mathcal{L}_{EmbKD}$. The overall

[1] We omit the temperature τ of softmax [19] without loss of generality

loss function $\mathcal{L}$ is given as follows:

$$\mathcal{L} = \mathcal{L}_{\text{BPR}} + \lambda_2 \cdot \mathcal{L}_{\text{PairKD}} + \lambda_3 \cdot \mathcal{L}_{\text{ListKD}} + \lambda_4 \cdot \mathcal{L}_{\text{EmbKD}} \quad (12)$$

$$\mathcal{L}_{\text{BPR}} = \sum_{u,i^+,i^-}^{|\mathcal{E}_{\text{bpr}}|} -\log\left(\text{sigmoid}(\mathbf{e}_u \cdot \mathbf{e}_{i^+} - \mathbf{e}_u \cdot \mathbf{e}_{i^-})\right) + \|\Theta\|^2 \quad (13)$$

where λ_2, λ_3, and λ_4 are parameters for loss term weighting. The last term $\|\Theta\|^2$ is the weight-decay regularization against over-fitting.

3.3 Model Complexity Analysis

The time complexity of current state-of-the-art graph-based multimodal recommender mainly consists of two parts: i) Modality feature reduction layer $\mathcal{R}(\cdot)$: The multimodal recommendation models inevitably need to incorporate feature reduction layers, as shown in Tab. 1. Most models employ a linear layer $O(\sum_{m\in\mathcal{M}} \times|\mathcal{I}|\times d_m \times d)$ or MLP transformation $O(\sum_{m\in\mathcal{M}} \times|\mathcal{I}| \times (d_m + d) \times d_h)$. $|\mathcal{I}|$ is the number of items. However, our inference model avoids the densification layer, due to the developed multi-modal knowledge distillation recommendation framework. ii) GNNs operations: Our inference model utilizes the LightGCN architecture solely in the graph convolutional component, resulting in the lowest consumption level $O(L\times|\mathcal{E}|\times d)$ among current graph-based recommendation models, where L is the number of GNNs layers and $|\mathcal{E}|$ denotes the number of observed interactions. Other models (*e.g.*, LATTICE, SLMRec, MICRO) also use lightweight architectures. However, GRCN and LATTICE require reconstruction operations that consume $|\mathcal{I}|^2 \times d$ and $|\mathcal{I}|^2 \times d_m$, respectively. The difference between them is that the weights of the reconstructed edges are based on densification d and the original high dimension d_m, respectively. LATTICE also takes $O(k \times |\mathcal{I}| \times log(|\mathcal{I}|))$ to retrieve top-k most similar items for each item. We summarize the computational complexity of the graph-based multimodal methods in Tab. 1

4 EVALUATION

4.1 Experimental Settings

4.1.1 **Dataset.** We conduct experiments on three multi-model recommendation datasets and summarize their statistics in Tab. 3.

- **Netflix:** This dataset contains user-item interactions from the Netflix platform. To construct the multi-model content, we crawled the movie posters based on the provided movie titles. The CLIP-ViT model [47] was used as the image feature extractor and BERT [27] is pre-trained for text feature encoding. We have released our pre-processed Netflix dataset, which includes the posters, to facilitate further research.
- **Tiktok:** This micro-video dataset [60] contains interactions with three types of modality features: visual, acoustic, and textual. The 128-dimensional visual and acoustic features were extracted from micro-videos desensitization, while the textual features were extracted from the captions using the Sentence-BERT model [48].
- **Electronics:** This dataset is based on the Electronics review data from Amazon. The visual modality includes 4,096-dimensional features that were extracted using pre-trained convolutional neural networks [15]. For the textual modality, we utilized Sentence-BERT [48] to combine various item attributes, such as title, descriptions, categories, and brands, into a compact 1024-d vector.

Table 3: Statistics of experimented datasets with multi-modal item Visual (V), Acoustic (A), Textual (T) contents.

Dataset	Netflix		Tiktok			Electronics	
Modality	V	T	V	A	T	V	T
Feat. Dim.	512	768	128	128	768	4096	1024
User	43,739		14,343			41,691	
Item	17,239		8,690			21,479	
Interaction	609,341		276,637			359,165	
Sparsity	99.919%		99.778%			99.960%	

* Tiktok: https://www.biendata.xyz/competition/icmechallenge2019/
* Electronics: http://jmcauley.ucsd.edu/data/amazon/links.html

4.1.2 **Evaluation Protocols.** We use two widely adopted metrics for top-K item recommendation task: Recall@K (R@K) and Normalized Discounted Cumulative Gain (N@K). We set K to 20 and 50 to evaluate the performance of our approach and several state-of-the-art baselines. We adopted the all-ranking strategy for evaluation, following the settings used in previous works [61, 63]. To conduct significance analysis, p-values were calculated using the results of our proposed approach and the best-performing baseline.

4.1.3 **Hyperparameter Settings.** We implemented our model framework using PyTorch and initialized model parameters using the Xavier initializer. We employ the AdamW optimizer [42] for both teacher $\mathcal{T}(\cdot)$ and student $\mathcal{S}(\cdot)$. The optimizer of the student will simultaneously optimize the parameters of both the student $\mathcal{S}(\cdot)$ and the prompt module $\mathcal{P}(\cdot)$, which is similar to that of the teacher's optimization. We search for the learning rates of $\mathcal{T}(\cdot)$ and $\mathcal{S}(\cdot)$ within the ranges of $[3.5e^{-4}, 9.8e^{-3}]$ and $[2.5e^{-4}, 8.5e^{-4}]$ respectively. The decay of the L_2 regularization term is tuned from $\{2.5e^{-3}, 7.4e^{-3}, 2.1e^{-2}\}$ for three datasets. All baselines are evaluated based on their source code and original papers, and the corresponding parameter tuning is conducted under a unified process.

4.1.4 **Baselines.** To comprehensively evaluate the performance of our proposed approach, we compared it against several state-of-the-art baselines from different research lines.

i) **Collaborative Filtering Models**

- **BPR-MF** [49]: It presents a generic optimization criterion, BPR-Opt, for personalized ranking that outperforms standard learning techniques for matrix factorization and adaptive kNN.
- **NGCF** [58]: It introduces GNNs to CF framework to model high-order information. The newly proposed embedding propagation layer allows the embeddings of users and items to interact with long-range information to harvest the collaborative signal.
- **LightGCN** [17]: It simplifies the graph convolution to remove the transformation and activation modules for model simplification.

ii) **Multi-Modal Recommender Systems**

- **VBPR** [16]: It proposes a Matrix Factorization approach to incorporate visual signals into prediction of user's preference for personalized ranking with implicit feedback.
- **MMGCN** [64]: It is built upon the graph-based information propagation framework with a multi-modal GNN, so as to guide representation learning of user preference in each modality.
- **GRCN** [63]: It designs adaptive refinement module to identify and prune potential false positive edges in the interaction structure, by considering multi-modal item characteristics.

- **LATTICE** [74]: This method discovers latent relationships between modalities using modality-aware structure learning layers to supplement collaborative signals for recommendation.
- **CLCRec** [62]: It studies the cold-start recommendation task and maximizes the mutual dependencies between item content and collaborative signals using contrastive learning.
- **SLMRec** [53]: This work captures multi-modal patterns in data by generating multiple views of individual items and using contrastive learning to distill additional supervised signals.
- **BM3** [80]: It leverages a latent representation dropout mechanism instead of graph augmentation to generate the target view of users/items for contrastive task without negative samples.

4.2 Performance Comparison

Tab. 4 presents the results of all methods on three datasets, with the results of MMKD and the best baseline highlighted in bold and underlined, respectively. Based on results and our analysis, we make the following key observations and conclusions:

- The proposed MMKD consistently outperforms both general collaborative filtering (CF) models and state-of-the-art multi-modal recommendation methods on all three datasets, demonstrating its effectiveness in multimedia recommendation. In particular, our proposed approach outperforms the best baselines by 9.75%, 13.26%, and 11.79% in terms of NDCG@50. The improved outcomes are attributed to our designed multi-modal knowledge distillation enhanced by prompt-tuning, which not only bridges the semantic gap during the multi-modal knowledge transfer, but also eliminates the impact of noise and redundancy of modality data. Furthermore, our results support the idea that multi-modal recommender systems perform better than general CF models, due to the incorporation of multi-modal context for assisting user preference learning under sparse data.
- Our MMKD achieves competitive results with a lightweight architecture and tailored transferred knowledge, suggesting that there may be noise in the multi-modal data. This finding confirms our motivation that directly incorporating multi-modal information into the user representations may introduce noise, which can misguide the encoding of modality-aware user preferences. To address this issue, our proposed approach disentangles the soft labels of collaborative relations during the knowledge distillation, which effectively alleviates the noise of multi-modal content by transferring more informative signals into the student model.
- Multi-modal recommendation methods often exhibit significant performance fluctuations on different datasets, due to overfitting. These models are highly influenced by the quality of model features as well as the number of interactions. For instance, LATTICE performs worse on Netflix with many interactions, which we attribute to the introduction of noise by the homogeneous co-current graph. In contrast, GRCN achieves superior performance on Netflix by identifying and removing false-positive edges in user-item graphs. BM3 and CLCRec do not use classical negative sampling of BPR and perform better on datasets with more implicit feedback than on Tiktok and Electronics. We speculate that this is because negative samples do not necessarily indicate users' dislikes. It may simply be due to the item not being presented.

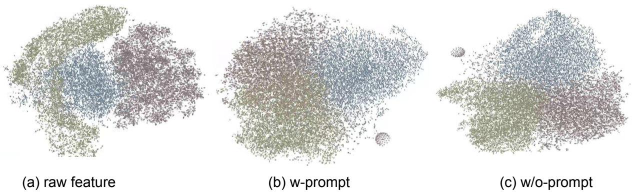

Figure 3: t-SNE Visualization on Tiktok for raw high dimensional multi-modal features $\mathbf{X}^m$, modality-specific representations $\mathbf{F}^m$ of MMKD and $\mathbf{F}^m$ of variant *w/o*-Prompt.

4.3 Ablation and Effectiveness Analyses

To justify the effectiveness of the proposed key components, we designed four variants of our MMKD and compared their performance against the original approach. The results in terms of Recall@20 and NDCG@20 are shown in Table 5. Further convergence analysis is provided in Supplementary. In order to gain deeper insights into the efficacy of the key components, we also conducted a further visualization analysis. Variant details are presented below:

- *w/o*-Prompt: This variant disables the prompt-tuning module to evaluate its impact on bridging the semantic gap during the teacher-student knowledge distillation process.
- *w/o*-PairKD: This variant examines the effect of ranking-based distillation for collaborative knowledge by removing the pair-wise knowledge distillation loss term $\mathcal{L}_{\text{PairKD}}$ from the joint loss.
- *w/o*-ListKD: The modality-aware disentangled knowledge distillation is not included to re-weight the soft-labels for alignment with the fine-grained knowledge decoupling.
- *w/o*-disentangle: This variant preserves the list-wise distillation in Sec. 3.2.2, while removing the disentangled part. Aiming to validate the utility of extracting more informative signals from modality features $\mathbf{f}^m$ with the list-wise objective, as well as the necessity of decoupling the transferred knowledge.

4.3.1 **Numerical Results.** As can be seen in the Tab. 5: (1) For variant *w/o*-Prompt, its performance on all three datasets has decreased compared to MMKD. This suggests that the removal of prompt-tuning may lead to the semantic gap for knowledge distillation. The modality-aware projection may also be in an overfitting state and can be limited to encode recommendation task-relevant multi-modal context without prompt-tuning enhancement. (2) The variant *w/o*-PairKD shows a decrease in performance compared to MMKD when pair-wise KD is disabled, demonstrating the strength of $\mathcal{L}_{\text{PKD}}$ in distilling ranking-based signals for model alignment. (3) Modality-aware list-wise distillation can finely extract quality modality-aware collaborative relationships, which helps in multi-modal recommendation. Therefore, the variant *w/o*-ListKD is inferior to the MMKD results. (4) The item-centric modality features are heavily biased against the preferences of the user. As a result, the variant *w/o*-disentangle performs poorly without disentangling and re-weighing distilled soft labels.

4.3.2 **Visualization Analysis.** As shown in Fig. 3, We conducted a visual analysis of modality-specific features on the TikTok dataset to intuitively understand the influence of introducing prompt-tuning for bridging the teacher model and the student model. Specifically, we applied t-SNE with PCA initialization to reduce the dimensionality of both the modality-specific densified features $\mathbf{f}_i^m \in \mathbb{R}^{|\mathcal{I}| \times d}$ ($w-$, $w/o-$ prompt-tuning) obtained from the feature densification layer, and the original multi-modal high-dimensional

Table 4: Performance comparison of baselines on different datasets in terms of *Recall*@20/50, and *NDCG*@20/50.

Baseline	Netflix				Tiktok				Electronics			
	R@20	N@20	R@50	N@50	R@20	N@20	R@50	N@50	R@20	N@20	R@50	N@50
MF-BPR	0.1583	0.0578	0.2396	0.0740	0.0488	0.0177	0.1038	0.0285	0.0211	0.0081	0.0399	0.0117
NGCF	0.1617	0.0612	0.2455	0.0767	0.0604	0.0206	0.1099	0.0296	0.0241	0.0095	0.0417	0.0128
LightGCN	0.1605	0.0609	0.2449	0.0768	0.0612	0.0211	0.1119	0.0301	0.0259	0.0101	0.0428	0.0132
VBPR	0.1661	0.0621	0.2402	0.0729	0.0525	0.0186	0.1061	0.0289	0.0234	0.0095	0.0409	0.0125
MMGCN	0.1685	0.0620	0.2486	0.0772	0.0629	0.0208	0.1221	0.0305	0.0273	0.0114	0.0445	0.0138
GRCN	0.1762	0.0661	0.2669	0.0868	0.0642	0.0211	0.1285	0.0311	0.0281	0.0117	0.0518	0.0158
LATTICE	0.1654	0.0623	0.2531	0.0770	0.0675	0.0232	0.1401	0.0362	0.0340	0.0135	0.0641	0.0184
CLCRec	0.1801	0.0719	0.2789	0.0892	0.0657	0.0214	0.1329	0.0329	0.0300	0.0118	0.0559	0.0169
SLMRec	0.1743	0.0682	0.2878	0.0869	0.0669	0.0221	0.1363	0.0342	0.0331	0.0132	0.0624	0.0180
BM3	0.1792	0.0720	0.2842	0.0923	0.0660	0.0225	0.1351	0.0343	0.0336	0.0141	0.0637	0.0195
MMKD	**0.1864**	**0.0743**	**0.3054**	**0.1013**	**0.0737**	**0.0258**	**0.1517**	**0.0410**	**0.0369**	**0.0155**	**0.0691**	**0.0218**
p-value	$1.60e^{-6}$	$5.90e^{-5}$	$2.99e^{-7}$	$1.11e^{-6}$	$1.41e^{-4}$	$5.59e^{-4}$	$5.00e^{-6}$	$1.29e^{-5}$	$3.24e^{-5}$	$2.96e^{-6}$	$7.51e^{-7}$	$4.63e^{-6}$

Table 5: Ablation study on key components of MMKD

Data	Netflix		Tiktok		Electronics	
Metrics	R@20	N@20	R@20	N@20	R@20	N@20
w/o-Prompt	0.1665	0.0662	0.0681	0.0240	0.0280	0.0117
w/o-PairKD	0.1774	0.0689	0.0692	0.0242	0.0277	0.0112
w/o-ListKD	0.1690	0.0487	0.0673	0.0234	0.0331	0.0136
w/o-disentangle	0.1712	0.0693	0.0706	0.0249	0.0353	0.0141
MMKD	**0.1864**	**0.0743**	**0.0737**	**0.0258**	**0.0369**	**0.0155**

Table 6: Model compactness and inference efficiency. "Time" indicates the average recommendation time for each epoch. "Memory" represents GPU memory usage. "Params" denotes the number of parameters. "Ratio" indicates the relative parameter size compared to the teacher. We use PyTorch with CUDA from RTX 3090 GPU and Intel Xeon W-2133 CPU.

Dataset	T-Model	Time	Memory	# Params	Ratio
Netflix	Teacher	42.6s	2.95GB	24.91M	-
	LATTICE	61.0s	18.24GB	24.06M	96.59%
	BM3	39.5s	2.69GB	24.06M	96.59%
	MMKD	23.3s	2.03GB	1.95M	7.83%
Electronics	Teacher	30.8s	5.02GB	99.04M	-
	LATTICE	45.1s	37.69GB	98.39M	99.34%
	BM3	31.8s	4.57GB	98.39M	99.34%
	MMKD	13.9s	3.81GB	2.67M	2.70%

* LATTICE out of memory on Electronics dataset, and we completed its experiment on A100.

features $\mathbf{x}^m \in \mathbb{R}^{|\mathcal{I}|\times d_m}$ into a 2-dimensional space. The results show that the original features $\mathbf{x}^m$ of diverse modalities exhibit significant differences in their vector space representation, with clear distinctions among different modalities, highlighting their association with distinct distributions. For modality-specific features $\mathbf{f}_i^m$, there are more overlaps in the prompt-tuning version, while the non-prompt-tuning version w/o–Prompt remains more confined to a modality-specific space. This suggests that prompt-tuning effectively strengthens the encoding of modality-specific features by extracting common user preferences pertaining to multiple ranking tasks while reducing the task-irrelevant features characteristic.

4.4 Study on Resource Consumption

In this section, we investigate the resource utilization of the teacher, student, and several baselines (BM3 and LATTICE) in terms of training time, storage, parameter count, and student-to-teacher parameter ratio for model compression. The specific numerical results on Netflix and Electronics are reported in Tab. 6. Results show that Our student model exhibits significantly lower inference and recommendation time consumption than other models, likely due to their larger size, which requires more time during gradient descent parameter updates. Additionally, BM3 has a self-supervised joint training task, and LATTICE has to dynamically learn homogeneous graphs, which increase computational time consumption. We find that the calculation of KL-Divergence in our model does not significantly increase time consumption, resulting in lower latency.

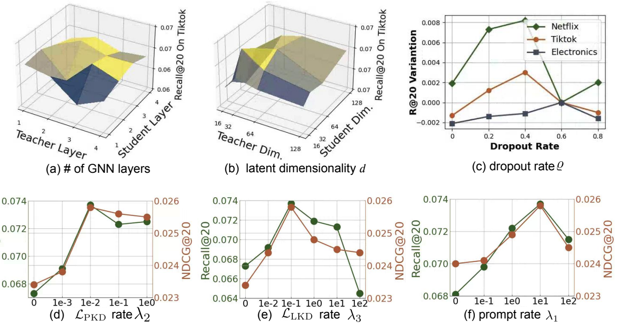

Figure 4: Impact study of hyperparameters in MMKD.

Moreover, the results show that our model has low storage consumption, with a much lower parameter quantity compared to other models, such as LATTICE which needs to dynamically calculate and store item-time relationships, incurring significant overhead. The numerical value of 'ratio=11.24% or 2.70%' indicates the effectiveness of our model as a compression algorithm. Supplementary provides model evaluation results with online incremental learning.

4.5 Impact Study of Hyperparameters

This section investigates the influence of several important hyperparameters in our proposed MMKD. We report the evaluation results in Fig. 4 and examine the effect of one hyperparameter at a time while keeping other parameters at their default settings.

- Representation Dimensionality d: We investigated the influence of representation dimensionality d on both the student $\mathcal{S}(\cdot)$ and teacher $\mathcal{T}(\cdot)$, with respect to the impact on recommendation system outcomes. We selected values of d from [16, 32, 64, 128], and found that the model's performance saturates when the number of hidden units reaches approximately 64 for the student. Notably, when the dimensions of the teacher and student are the same, the student's results are better. This is because the score of KD is obtained by the inner product of the representations, and the dimension size determines the scale of the score. Having the same scale level leads to more accurate KD.

- Depth of GNNs L: We examine the influence of the depth of the GNNs in the range of [1, 2, 3, 4]. The results show that the teacher's performance improves as the layer count increases, while the student's performance remains moderate. We speculate that this is because the teacher needs to encode useful knowledge with high-order relationships, and our modality-aware ranking-based KD effectively transfers quality knowledge to the student.
- Dropout Ratio of Teacher's Modality Encoding Layer: We investigate the influence of the dropout ratio of the teacher's modality encoding layer, which ranges from 0 to 1. Our results show that without dropout, the teacher's performance drops sharply, indicating overfitting in the multi-modal feature encoder. A higher dropout rate is required for datasets with higher original feature dimensions, confirming the risk of overfitting the multi-modal feature with existing modality encoders.
- Pair/List-wise KD Loss Weight λ_2, λ_3: The pair/list-wise KD loss weights (λ_2, λ_3) indicate the strength of collaborative knowledge distillation and disentangled modality-aware knowledge distillation, respectively. We vary the weights in the range of [0, $1e$-2, $1e$-1, $1e0$, $1e1$, $1e2$]. Evaluation results show that absent or small weights significantly decrease the model's performance.
- Prompt Rate λ_1: It controls the soft-token rate. Our results show that without a soft-token rate, the model's performance significantly decreases, indicating that prompt-tuning enables teachers to generate more helpful knowledge for students. We speculate that this is because the prompt module optimizes alongside the student learning, leading to better recommendation performance.

5 RELATED WORK

Multi-Modal Recommender Systems. Researchers have explored using multi-modal content to enhance recommenders. Pioneering studies, such as VBPR[16] have incorporated visual features into CF frameworks. After that, GNN-enhanced multi-modal recommenders capture high-order connectivity by incorporating modality signals. Inspired by the success of self-supervised learning, recent multimedia recommenders, such as MMGCL [73], SLMRec [53], use data augmentation strategies to enhance the representation learning, PAMD[13] uses contrastive learning to align modalities. Additionally, some other works focus on a different scenario for recommending multimedia content (*e.g.*, fashion [66], news[50, 65], micro-video [23, 38]) with sequential learning [34, 79]. For the cold start scenario, several studies (*e.g.*, MTPR [4], MML [45], and INP [36]) have used multi-modal content to improve item representations. For debiasing in multi-modal scenario, CausalRec [46] uses a causal graph for visual debiasing while InvRL[5], CGKR[44] addresses spurious correlations in multimedia content. Despite their effectiveness, most of them are built upon cumbersome multi-modal feature encoding which limits their scalability in practice.

Knowledge Distillation for Recommendation. KD in recommendation has sparked various research directions [54, 68]. Many studies have explored the distillable aspects of recommenders. For relation-based KD, HTD[25] distilled the topological knowledge built upon the relations in the teacher embedding space and [78] distilled structured knowledge from a differentiable path-based recommendation model. DE-RRD[24] conducts both feature-based KD and relation-based KD. Besides, different distillation paradigms are also of interest to researchers[71, 72]. LUME[71] uses multi-teacher ensemble and debiased KD to aggregate knowledge from multiple pretrained teachers. CrossDistil[72], [30] encourage mutual distillation between models. KD can bring benefits for scenario-specific recommenders[26, 52, 57]. For the recommender with heterogeneous information, HetComp[26] transfers the ensemble knowledge of heterogeneous teachers to a lightweight student and DESIGN[52] integrates information from the homogenous graph and trains two auxiliary smaller models. Recommender with privileged feature also can leverage KD[57]. For example, student of PGD[57] can learn the distilled output with privileged CF embeddings to tackle the cold start problem. And there also are some efforts focus on debias when conducting KD[2, 71].

We introduce KD in multi-modal scenario for two purposes: i) Solve the problem of large coding parameter models caused by high-dimensional output features in multi-modal scenes; ii) reduce the impact of noise in modal content and emphasize the transfer of knowledge that is useful and relevant for downstream tasks.

Prompt Learning. Prompt learning has become a emerging research direction in the context of large pre-trained models [1, 37]. It allows these models to be repurposed for different tasks without requiring additional training [39, 43]. The core idea behind prompt learning is to generate effective prompts that guide the model to generate the expected output for specific downstream tasks [12, 31]. Recent studies explores prompt learning in various domains, such as graph mining, knowledge graph representation learning and recommendation [8, 40, 76]. For example, GraphPrompt [41] defines the paradigm of prompts on graphs. To transfer knowledge graph semantics into task data, KGTransformer [77] regards task data as a triple prompt for tuning. Additionally, prompt-based learning has also been introduced to benefit recommender systems to enhance model explanation [32], fairness [67], sequence learning [69]. Motivated by these research lines, we propose a novel multi-modal prompt learning approach that can adaptively guide knowledge distillation for simple yet effective multimedia recommendation.

6 CONCLUSION

The objective of this work is to simplify and enhance multi-modal recommenders using a novel modality-aware KD framework empowered by prompt-tuning. In order to effectively transfer task-relevant knowledge from the teacher to the student model, we introduce a learnable prompt module that dynamically bridges the semantic gap between the multi-modal context encoding in the teacher model and the collaborative relation modeling in the student model. Additionally, our proposed framework, called MMKD, aims to disentangle the informative collaborative relationships, thereby enabling augmented knowledge distillation. Through extensive experiments, we demonstrate that MMKD significantly improves model efficiency while maintaining superior accuracy compared to state-of-the-art solutions. Our future work plans to integrate LLMs with multi-modal context encoding for performance enhancement.

REFERENCES

[1] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, et al. 2020. Language models are few-shot learners. *Neural Information Processing Systems* 33 (2020), 1877–1901.

[2] Gang Chen, Jiawei Chen, Fuli Feng, Sheng Zhou, and Xiangnan He. 2023. Unbiased Knowledge Distillation for Recommendation. In *ACM International Conference on Web Search and Data Mining*. 976–984.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[4] Xiaoyu Du, Xiang Wang, Xiangnan He, Zechao Li, Jinhui Tang, and Tat-Seng Chua. 2020. How to learn item representation for cold-start multimedia recommendation?. In *ACM Multimedia Conference*. 3469–3477.

[5] Xiaoyu Du, Zike Wu, Fuli Feng, Xiangnan He, and Jinhui Tang. 2022. Invariant Representation Learning for Multimedia Recommendation. In *ACM Multimedia Conference*. 619–628.

[6] Jerome H Friedman. 1997. On bias, variance, 0/1âloss, and the curse-of-dimensionality. *Data mining and knowledge discovery* 1 (1997), 55–77.

[7] Rong Ge and Tengyu Ma. 2017. On the optimization landscape of tensor decompositions. *Neural Information Processing Systems* 30 (2017).

[8] Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. 2022. Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). In *Proceedings of the 16th ACM Conference on Recommender Systems*. 299–315.

[9] Xue Geng, Hanwang Zhang, Jingwen Bian, et al. 2015. Learning image and user features for recommendation in social networks. In *IEEE/CVF International Conference on Computer Vision*. 4274–4282.

[10] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*. JMLR Workshop and Conference Proceedings, 249–256.

[11] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision* 129 (2021), 1789–1819.

[12] Yuxian Gu, Xu Han, Zhiyuan Liu, and Minlie Huang. 2021. Ppt: Pre-trained prompt tuning for few-shot learning. *arXiv preprint arXiv:2109.04332* (2021).

[13] Tengyue Han, Pengfei Wang, Shaozhang Niu, and Chenliang Li. 2022. Modality matches modality: Pretraining modality-disentangled item representations for recommendation. In *Proceedings of the ACM Web Conference 2022*. 2058–2066.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 770–778.

[15] Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *proceedings of the 25th international conference on world wide web*. 507–517.

[16] Ruining He and Julian McAuley. 2016. VBPR: visual bayesian personalized ranking from implicit feedback. In *Association for the Advancement of Artificial Intelligence*, Vol. 30.

[17] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *ACM Special Interest Group on Information Retrieval*. 639–648.

[18] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *ACM International World Wide Web Conference*. 173–182.

[19] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* (2015).

[20] Zhenyu Hou, Xiao Liu, Yukuo Cen, Yuxiao Dong, Jie Tang, et al. 2022. Graphmae: Self-supervised masked graph autoencoders. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

[21] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. 2022. Visual prompt tuning. In *ECCV*. Springer, 709–727.

[22] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. 2014. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*. 675–678.

[23] Hao Jiang, Wenjie Wang, Yinwei Wei, Zan Gao, Yinglong Wang, and Liqiang Nie. 2020. What aspect do you like: Multi-scale time-aware user interest modeling for micro-video recommendation. In *ACM Multimedia Conference*. 3487–3495.

[24] SeongKu Kang, Junyoung Hwang, Wonbin Kweon, and Hwanjo Yu. 2020. DE-RRD: A knowledge distillation framework for recommender system. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 605–614.

[25] SeongKu Kang, Junyoung Hwang, Wonbin Kweon, and Hwanjo Yu. 2021. Topology distillation for recommender system. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 829–839.

[26] SeongKu Kang, Wonbin Kweon, Dongha Lee, Jianxun Lian, et al. 2023. Distillation from Heterogeneous Models for Top-K Recommendation. In *ACM International World Wide Web Conference*. 801–811.

[27] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics*. 4171–4186.

[28] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. 2017. On large-batch training for deep learning: Generalization gap and sharp minima. *International Conference on Learning Representations* (2017).

[29] Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics* 22, 1 (1951), 79–86.

[30] Wonbin Kweon, SeongKu Kang, and Hwanjo Yu. 2021. Bidirectional distillation for top-K recommender system. In *Proceedings of the Web Conference 2021*. 3861–3871.

[31] Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. *Conference on Empirical Methods in Natural Language Processing* (2021).

[32] Lei Li, Yongfeng Zhang, and Li Chen. 2023. Personalized prompt learning for explainable recommendation. *Transactions on Information Systems (TOIS)* 41, 4 (2023), 1–26.

[33] Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190* (2021).

[34] Jiahao Liang, Xiangyu Zhao, Muyang Li, Zijian Zhang, Wanyu Wang, Haochen Liu, and Zitao Liu. 2023. MMMLP: Multi-modal Multilayer Perceptron for Sequential Recommendations. In *Proceedings of the ACM Web Conference 2023*. 1109–1117.

[35] Fan Liu, Huilin Chen, Zhiyong Cheng, Anan Liu, Liqiang Nie, and Mohan Kankanhalli. 2022. Disentangled multimodal representation learning for recommendation. *IEEE Transactions on Multimedia* (2022).

[36] Huafeng Liu, Liping Jing, Dahai Yu, Mingjie Zhou, and Michael Ng. 2022. Learning Intrinsic and Extrinsic Intentions for Cold-start Recommendation with Neural Stochastic Processes. In *ACM Multimedia Conference*. 491–500.

[37] Peng Liu, Lemei Zhang, and Jon Atle Gulla. 2023. Pre-train, Prompt and Recommendation: A Comprehensive Survey of Language Modelling Paradigm Adaptations in Recommender Systems. *arXiv preprint arXiv:2302.03735* (2023).

[38] Yiyu Liu, Qian Liu, Yu Tian, Changping Wang, Yanan Niu, Yang Song, and Chenliang Li. 2021. Concept-Aware Denoising Graph Neural Network for Micro-Video Recommendation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 1099–1108.

[39] Yong Liu, Susen Yang, Chenyi Lei, Guoxin Wang, Haihong Tang, Juyong Zhang, Aixin Sun, and Chunyan Miao. 2021. Pre-training graph transformer with multimodal side information for recommendation. In *Proceedings of the 29th ACM International Conference on Multimedia*. 2853–2861.

[40] Zhuang Liu, Yunpu Ma, Matthias Schubert, Yuanxin Ouyang, and Zhang Xiong. 2022. Multi-Modal Contrastive Pre-training for Recommendation. In *Proceedings of the 2022 International Conference on Multimedia Retrieval*. 99–108.

[41] Zemin Liu, Xingtong Yu, Yuan Fang, and Xinming Zhang. 2023. GraphPrompt: Unifying Pre-Training and Downstream Tasks for Graph Neural Networks. *ACM International World Wide Web Conference* (2023).

[42] Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

[43] Zaiqiao Meng, Siwei Liu, Craig Macdonald, and Iadh Ounis. 2021. Graph neural pre-training for enhancing recommendations using side information. *arXiv preprint arXiv:2107.03936* (2021).

[44] Shanlei Mu, Yaliang Li, Wayne Xin Zhao, Jingyuan Wang, Bolin Ding, and Ji-Rong Wen. 2022. Alleviating spurious correlations in knowledge-aware recommendations through counterfactual generator. In *Proceedings of the 45th International ACM ACM Special Interest Group on Information Retrieval Conference on Research and Development in Information Retrieval*. 1401–1411.

[45] Xingyu Pan, Yushuo Chen, Changxin Tian, Zihan Lin, Jinpeng Wang, He Hu, and Wayne Xin Zhao. 2022. Multimodal meta-learning for cold-start sequential recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 3421–3430.

[46] Ruihong Qiu, Sen Wang, Zhi Chen, et al. 2021. Causalrec: Causal inference for visual debiasing in visually-aware recommendation. In *ACM Multimedia Conference*. ACM, 3844–3852.

[47] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. PMLR, 8748–8763.

[48] Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Conference on Empirical Methods in Natural Language Processing*.

[49] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*.

[50] Shaoyun Shi, Weizhi Ma, Zhen Wang, Min Zhang, Kun Fang, Jingfang Xu, Yiqun Liu, and Shaoping Ma. 2021. WG4Rec: Modeling textual content with word

graph for news recommendation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 1651–1660.

[51] Mingchen Sun, Kaixiong Zhou, Xin He, Ying Wang, and Xin Wang. 2022. Gppt: Graph pre-training and prompt tuning to generalize graph neural networks. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1717–1727.

[52] Ye Tao, Ying Li, Su Zhang, Zhirong Hou, and Zhonghai Wu. 2022. Revisiting Graph based Social Recommendation: A Distillation Enhanced Social Graph Network. In *Proceedings of the ACM Web Conference 2022*. 2830–2838.

[53] Zhulin Tao, Xiaohao Liu, Yewei Xia, Xiang Wang, Lifang Yang, Xianglin Huang, and Tat-Seng Chua. 2022. Self-supervised Learning for Multimedia Recommendation. *Transactions on Multimedia (TMM)* (2022).

[54] Chenyang Wang, Zhefan Wang, Yankai Liu, Yang Ge, Weizhi Ma, Min Zhang, Yiqun Liu, Junlan Feng, Chao Deng, and Shaoping Ma. 2022. Target Interest Distillation for Multi-Interest Recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 2007–2016.

[55] Qifan Wang, Yinwei Wei, Jianhua Yin, Jianlong Wu, Xuemeng Song, and Liqiang Nie. 2021. DualGNN: Dual Graph Neural Network for Multimedia Recommendation. *Transactions on Multimedia (TMM)* (2021).

[56] Qinyong Wang, Hongzhi Yin, Hao Wang, Quoc Viet Hung Nguyen, Zi Huang, and Lizhen Cui. 2019. Enhancing collaborative filtering with generative augmentation. In *Proceedings of the 25th ACM SIGACM SIGKDD Conference on Knowledge Discovery and Data Mining International Conference on Knowledge Discovery & Data Mining*. 548–556.

[57] Shuai Wang, Kun Zhang, Le Wu, Haiping Ma, Richang Hong, et al. 2021. Privileged graph distillation for cold start recommendation. In *ACM Special Interest Group on Information Retrieval*. 1187–1196.

[58] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In *ACM Special Interest Group on Information Retrieval*.

[59] Yuhao Wang, Xiangyu Zhao, Bo Chen, Qidong Liu, Huifeng Guo, Huanshuo Liu, Yichao Wang, Rui Zhang, and Ruiming Tang. 2023. PLATE: A Prompt-Enhanced Paradigm for Multi-Scenario Recommendations. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1498–1507.

[60] Wei Wei, Chao Huang, Lianghao Xia, and Chuxu Zhang. 2023. Multi-Modal Self-Supervised Learning for Recommendation. In *ACM International World Wide Web Conference*. 790–800.

[61] Yinwei Wei, Xiang Wang, Xiangnan He, Liqiang Nie, Yong Rui, and Tat-Seng Chua. 2021. Hierarchical user intent graph network for multimedia recommendation. *Transactions on Multimedia (TMM)* (2021).

[62] Yinwei Wei, Xiang Wang, Qi Li, Liqiang Nie, Yan Li, Xuanping Li, et al. 2021. Contrastive learning for cold-start recommendation. In *ACM Multimedia Conference*. 5382–5390.

[63] Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, and Tat-Seng Chua. 2020. Graph-refined convolutional network for multimedia recommendation with implicit feedback. In *ACM Multimedia Conference*. 3541–3549.

[64] Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, Richang Hong, and Tat-Seng Chua. 2019. MMGCN: Multi-modal graph convolution network for personalized recommendation of micro-video. In *ACM Multimedia Conference*. 1437–1445.

[65] Chuhan Wu, Fangzhao Wu, Yongfeng Huang, and Xing Xie. 2023. Personalized news recommendation: Methods and challenges. *ACM Transactions on Information Systems* 41, 1 (2023), 1–50.

[66] Yaxiong Wu, Craig Macdonald, and Iadh Ounis. 2022. Multi-Modal Dialog State Tracking for Interactive Fashion Recommendation. In *ACM Conference on Recommender Systems*. 124–133.

[67] Yiqing Wu, Ruobing Xie, Yongchun Zhu, Fuzhen Zhuang, Ao Xiang, et al. 2022. Selective fairness in recommendation via prompts. In *ACM Special Interest Group on Information Retrieval*. 2657–2662.

[68] Xin Xia, Hongzhi Yin, Junliang Yu, Qinyong Wang, Guandong Xu, and Quoc Viet Hung Nguyen. 2022. On-Device Next-Item Recommendation with Self-Supervised Knowledge Distillation. In *ACM Special Interest Group on Information Retrieval*. 546–555.

[69] Xin Xin, Tiago Pimentel, Alexandros Karatzoglou, Pengjie Ren, Konstantina Christakopoulou, and Zhaochun Ren. 2022. Rethinking Reinforcement Learning for Recommendation: A Prompt Perspective. In *ACM Special Interest Group on Information Retrieval*. 1347–1357.

[70] Jiahao Xun, Shengyu Zhang, Zhou Zhao, Jieming Zhu, Qi Zhang, Jingjie Li, Xiuqiang He, Xiaofei He, Tat-Seng Chua, and Fei Wu. 2021. Why do we click: visual impression-aware news recommendation. In *Proceedings of the 29th ACM International Conference on Multimedia*. 3881–3890.

[71] Guipeng Xv, Xinyi Liu, Chen Lin, Hui Li, Chenliang Li, and Zhenhua Huang. 2022. Lightweight Unbiased Multi-teacher Ensemble for Review-based Recommendation. In *ACM International Conference on Information and Knowledge Management*. 4620–4624.

[72] Chenxiao Yang, Junwei Pan, Xiaofeng Gao, Tingyu Jiang, Dapeng Liu, and Guihai Chen. 2022. Cross-task knowledge distillation in multi-task recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 4318–4326.

[73] Zixuan Yi, Xi Wang, Iadh Ounis, and Craig Macdonald. 2022. Multi-modal Graph Contrastive Learning for Micro-video Recommendation. In *ACM Special Interest Group on Information Retrieval*. 1807–1811.

[74] Jinghao Zhang, Yanqiao Zhu, Qiang Liu, Shu Wu, Liang Wang, et al. 2021. Mining Latent Structures for Multimedia Recommendation. In *ACM Multimedia Conference*. 3872–3880.

[75] Jinghao Zhang, Yanqiao Zhu, Qiang Liu, Mengqi Zhang, Shu Wu, and Liang Wang. 2022. Latent Structure Mining with Contrastive Modality Fusion for Multimedia Recommendation. *IEEE Transactions on Knowledge and Data Engineering* (2022).

[76] Lingzi Zhang, Xin Zhou, and Zhiqi Shen. 2023. Multimodal Pre-training Framework for Sequential Recommendation via Contrastive Learning. *arXiv preprint arXiv:2303.11879* (2023).

[77] Wen Zhang, Yushan Zhu, Mingyang Chen, Yuxia Geng, Yufeng Huang, Yajing Xu, Wenting Song, and Huajun Chen. 2023. Structure Pretraining and Prompt Tuning for Knowledge Graph Transfer. *ACM International World Wide Web Conference* (2023).

[78] Yuan Zhang, Xiaoran Xu, Hanning Zhou, and Yan Zhang. 2020. Distilling structured knowledge into embeddings for explainable and accurate recommendation. In *Proceedings of the 13th international conference on web search and data mining*. 735–743.

[79] Xiaolin Zheng, Jiajie Su, Weiming Liu, and Chaochao Chen. 2022. DDGHM: Dual Dynamic Graph with Hybrid Metric Training for Cross-Domain Sequential Recommendation. In *ACM Multimedia Conference*. 471–481.

[80] Xin Zhou, Hongyu Zhou, Yong Liu, Zhiwei Zeng, Chunyan Miao, Pengwei Wang, Yuan You, and Feijun Jiang. 2022. Bootstrap latent representations for multimodal recommendation. In *ACM International World Wide Web Conference*.

7 SUPPLEMENTARY MATERIAL

In our supplementary materials, we begin by outlining the key steps involved in learning our proposed model. We then provide detailed derivations of the decoupling collaborative relationships to supplement our model's ability to distill more informative signals and reduce noise in the multi-modal knowledge distillation paradigm. Furthermore, we present additional experiments on real-time incremental learning scenarios to further demonstrate the validity of our proposed MMKD framework in practical scenarios. Finally, we provide a list of the source code used for the baseline methods.

7.1 Pseudocode of MMKD's Learning Procedure

In this subsection, we outline the key steps involved in learning our proposed MMKD with the following presented pseudocode:

```
Algorithm 1: Framework of MMKD
    Input: Bipartite interaction graph G=( { U, I },E, $\mathbf{X}$), with
           modality-specific node attribute
           $\mathbf{X}^m \in \mathbb{R}^{|\mathcal{I}| \times d_m} (\mathbf{X}^m \in \mathbf{X})$ and normalized
           user-item interaction matrix $\mathbf{A}$.
    Output: User's modality-aware interactive preference
            $\hat{y}_{u,i} (i \in \mathcal{I}, u \in \mathcal{U})$.
    Initialize: Xavier initialized parameters of
      $\mathcal{T}(\cdot), \mathcal{S}(\cdot), \mathcal{P}(\cdot), \mathcal{R}(\cdot)$.
    // Offline Teacher Training
    for epoch $\leftarrow 0, 1, ...$ do
        for step $\leftarrow 0, 1, ...$ do
            // Teacher Inference
            $\mathbf{p} \longleftarrow Eq. 3$
            $\mathbf{E}_u^{\mathcal{T}}, \mathbf{E}_i^{\mathcal{T}}, \quad \mathbf{F}_u^1, ..., \mathbf{F}_u^m, ..., \mathbf{F}_i^1, ..., \mathbf{F}_i^m ... \longleftarrow \mathcal{T}(\mathbf{A}, \mathbf{X})$
            // Teacher Objective
            $\mathcal{L}_{\text{BPR}} \longleftarrow \mathbf{E}_u^{\mathcal{T}}, \mathbf{E}_i^{\mathcal{T}}$
            Gradient descent for $\mathcal{T}(\cdot)$ and $\mathcal{P}(\cdot)$.
        end
    end
    // Student Training
    for epoch $\leftarrow 0, 1, ...$ do
        for step $\leftarrow 0, 1, ...$ do
            // Teacher Inference (as above)
            ...
            // Student Inference
            $\mathbf{E}_u^{\mathcal{S}}, \mathbf{E}_i^{\mathcal{S}} \longleftarrow \mathcal{S}(\mathbf{E}_u^{\mathcal{T}}, \mathbf{E}_i^{\mathcal{T}}, \mathbf{A})$
            // Knowledge Distillation
            $\mathcal{L}_{\text{PairKD}} \longleftarrow \mathbf{E}_u^{\mathcal{T}}, \mathbf{E}_i^{\mathcal{T}}, \mathbf{E}_u^{\mathcal{S}}, \mathbf{E}_i^{\mathcal{S}}$
            $\mathcal{L}_{\text{ListKD}} \longleftarrow \mathbf{F}_u^1, ..., \mathbf{F}_u^m, ..., \mathbf{F}_i^1, ..., \mathbf{F}_i^m ..., \mathbf{E}_u^{\mathcal{S}}, \mathbf{E}_i^{\mathcal{S}}$
            $\mathcal{L}_{\text{EKD}} \longleftarrow \mathbf{F}_u^1, ..., \mathbf{F}_u^m, ..., \mathbf{F}_i^1, ..., \mathbf{F}_i^m ..., \mathbf{E}_u^{\mathcal{S}}, \mathbf{E}_i^{\mathcal{S}}$
            // Student Objective
            L $\longleftarrow \mathcal{L}_{\text{BPR}} + \lambda_2 \cdot \mathcal{L}_{\text{PairKD}} + \lambda_3 \cdot \mathcal{L}_{\text{ListKD}} + \lambda_4 \cdot \mathcal{L}_{\text{EmbKD}}$
            Gradient descent for $\mathcal{S}(\cdot)$ and $\mathcal{P}(\cdot)$.
        end
    end
```

7.2 Derivations of Disentangling Ranking Score

In our MMKD, we propose to improve our collaborative knowledge distillation approach by incorporating modality-aware user interaction patterns and decoupling the transferred knowledge based on its degree of uncertainty. By considering the reliability of observed user-item relations, including interactions and non-interactions, we can better identify the most informative collaborative signals from the teacher model. To achieve this, we will utilize a modality-aware decoupled KD with a listwise loss, which will allow us to more flexibly handle collaborative signals related to different modalities.

The details presented below are related to the mathematical derivation discussed in Section 3.2.2:

$$
\begin{aligned}
KL(\mathbf{y}_{\text{list}}^T \| \mathbf{y}_{\text{list}}^S) &= \sum_{k=1}^{K} y_k^T log(\frac{y_k^T}{y_k^S}) \qquad (14) \\
&= y_1^{+T} log(\frac{y_1^{+T}}{y_1^{+S}}) + \sum_{k=2,k\neq1}^{K} y_k^{-T} log(\frac{y_k^{-T}}{y_k^{-S}}) \\
&= b^{+T} log(\frac{b^{+T}}{b^{+S}}) + \sum_{k=2,k\neq1}^{K} y_k^{-T} log(\frac{y_k^{-T}}{y_k^{-S}})
\end{aligned}
$$

Eq.12 can be expressed as $q_k = y_k / b^-$ by applying Eq. 6 and Eq. 7, as the class index k is not affected by b^{-T} and b^{-S}.

$$
\begin{aligned}
&= b^{+T} log(\frac{b^{+T}}{b^{+S}}) + \sum_{k=2,k\neq1}^{K} y_k^{-T} log(\frac{y_k^{-T}}{y_k^{-S}}) \qquad (15) \\
&= b^{+T} \log(\frac{b^{+T}}{b^{+S}}) + \sum_{k=2,k\neq1}^{K} b^{-T} q_k^T (\log(\frac{b^{-T} q_k^T}{b^{-T} q_k^S})) \\
&= b^{+T} \log(\frac{b^{+T}}{b^{+S}}) + \sum_{k=2,k\neq1}^{K} b^{-T} q_k^T (\log(\frac{b^{-T}}{b^{-S}}) + \log(\frac{q_k^T}{q_k^S})) \\
&= b^{+T} \log(\frac{b^{+T}}{b^{+S}}) + \sum_{k=2,k\neq1}^{K} b^{-T} q_k^T \log(\frac{b^{-T}}{b^{-S}}) + \sum_{k=2,k\neq1}^{K} b^{-T} q_k^T (\log(\frac{q_k^T}{q_k^S})) \\
&= b^{+T} \log(\frac{b^{+T}}{b^{+S}}) + b^{-T} \log(\frac{b^{-T}}{b^{-S}}) \sum_{k=2,k\neq1}^{K} q_k^T + \sum_{k=2,k\neq1}^{K} b^{-T} q_k^T (\log(\frac{q_k^T}{q_k^S})) \\
&= b^{+T} \log(\frac{b^{+T}}{b^{+S}}) + b^{-T} \log(\frac{b^{-T}}{b^{-S}}) + \sum_{k=2,k\neq1}^{K} b^{-T} q_k^T (\log(\frac{q_k^T}{q_k^S})) \\
&= b^{+T} \log(\frac{b^{+T}}{b^{+S}}) + b^{-T} \log(\frac{b^{-T}}{b^{-S}}) + (1 - b^{+T}) \sum_{k=2,k\neq1}^{K} q_k^T (\log(\frac{q_k^T}{q_k^S}))
\end{aligned}
$$

Consequently, $KL(\mathbf{y}_{\text{list}}^T \| \mathbf{y}_{\text{list}}^S)$ may be rewritten as follows:

$$
= \underbrace{b^{+T} \log(\frac{b^{+T}}{b^{+S}}) + b^{-T} \log(\frac{b^{-T}}{b^{-S}})}_{KL(\mathbf{b}^T \| \mathbf{b}^S)} + (1 - b^{+T}) \underbrace{\sum_{k=2,i\neq1}^{K} {q_k}_k^T \log(\frac{{q_k}^T}{{q_k}^S})}_{KL(\mathbf{q}^T \| \mathbf{q}^S)} \qquad (16)
$$

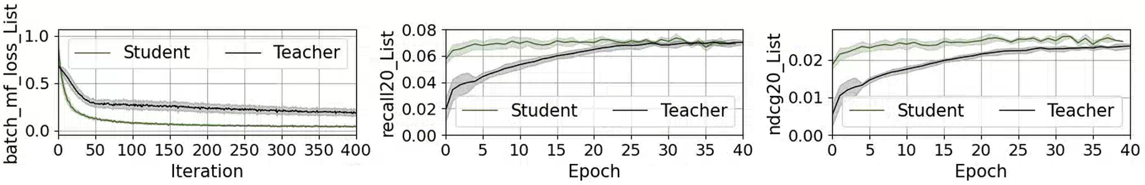

Figure 5: Training curves of MMKD framework in terms of Recall@20, NDCG@20, and $\mathcal{L}$ on Tiktok dataset.

By discriminating the knowledge transferred from the teacher model in terms of its uncertainty, our model can effectively manage the collaborative signals associated with different modalities, thereby reducing the noise and improving the accuracy of the transferred knowledge in our multi-modal knowledge distillation.

7.3 Supplementary Experiments

7.3.1 **Incremental Learning over New Data.** To validate the efficiency and effectiveness of our lightweight inference model (*i.e.*, student) for making recommendation in real-life online platforms, we simulate a scenario where the model is fine-tuned with a small number of new interactions and used for predicting dynamic user preferences movies from Netflix data.

Table 7: Performance of model adaptation to new data. The time cost (measured by ms) is calculated by training the model and making inference with recommendation results. Better results can be obtained by our model compared with baselines in terms of both efficiency and accuracy.

Methods	Recall@20	NDCG@20	Time
LATTICE	0.1210	0.0487	1,605.26ms
BM3	0.1381	0.0574	790.0ms
MMKD	0.1454	0.0586	482.5ms

Table 7 demonstrates that the lightweight inference model outperforms the baseline methods (BM3 and LATTICE) with cumbersome multi-modal encoding frameworks in terms of efficiency. Our designed multi-modal knowledge distillation paradigm allows our model to achieve comparable recommendation accuracy to more complex multi-modal recommender systems (such as BM3), while significantly improving performance compared to LATTICE. This is because our model effectively transfers multi-modal collaborative relationships from the teacher model, thereby preserving modality-aware user preferences in the smaller student model.

The experiments performed under incremental learning recommendation environment indicate that a lightweight inference model can better adapt to new recommendation data with much lower computational cost, while performing comparably to more cumbersome baselines in terms of Recall@20 and NDCG@20. This further confirms the performance superiority of our MMKD in practical recommender systems, providing an efficient and effective inference model for real-time recommendation.

7.3.2 **Convergence Analysis.** In this section, we use convergence analysis to examine the effects of our multi-modal knowledge distillation recommendation system on the effectiveness of model training. The convergence process with respect to Recall@20 and $\mathcal{L}$ for each epoch is shown in Fig. 5.

7.3.3 **Parameter Size Comparison.** By analyzing the parameter statistics of both the teacher and student models, we can observe that our MMKDeffectively compresses the student model, resulting in a significantly smaller parameter size.

Table 8: # Parameters of Teacher and Student Models.

# Parameters: Teacher (Netflix)		
Weight Name	Weight Shape	Number
image_trans.weight	torch.Size([32, 512])	16384
image_trans.bias	torch.Size([32])	32
text_trans.weight	torch.Size([32, 768])	24576
text_trans.bias	torch.Size([32])	32
user_id_embedding.weight	torch.Size([43739, 32])	1399648
item_id_embedding.weight	torch.Size([43739, 32])	1399648
image_embedding.weight	torch.Size([17239, 32])	8826368
text_embedding.weight	torch.Size([17239, 32])	13239552
batch_norm.weight	torch.Size([32])	32
batch_norm.bias	torch.Size([32])	32
The total number of parameters:		24906304
The parameters of Model Teacher_Model:		24.906304M

# Parameters: Student: LightGCN (Netflix)		
Weight Name	Weight Shape	Number
user_id_embedding.weight	torch.Size([43739, 32])	1399648
item_id_embedding.weight	torch.Size([17239, 32])	551648
The total number of parameters:		1951296
The parameters of Model Student_LightGCN:		1.951296M

* Correct the value in Tab.6.

# Parameters: Student: GCN (Netflix)		
Weight Name	Weight Shape	Number
layer_list.0.user_weight	torch.Size([32, 32])	1024
layer_list.0.item_weight	torch.Size([32, 32])	1024
The total number of parameters:		2048
The parameters of Model Student_GCN:		0.002048M

7.3.4 **Baseline Model Implementations.** The publicly available source codes for the baselines can be found at the following URLs:

- **BPR-MF**: https://github.com/gamboviol/bpr.git
- **NGCF**: https://github.com/huangtinglin/NGCF-PyTorch.git
- **LightGCN**: https://github.com/kuandeng/LightGCN.git
- **VBPR**: https://github.com/DevilEEE/VBPR.git
- **MMGCN**: https://github.com/weiyinwei/MMGCN.git
- **GRCN**: https://github.com/weiyinwei/GRCN.git
- **LATTICE**: https://github.com/CRIPAC-DIG/LATTICE.git
- **CLCRec**: https://github.com/weiyinwei/CLCRec.git
- **SLMRec**: https://github.com/zltao/SLMRec.git
- **BM3**: https://github.com/enoche/BM3.git