How to Learn Item Representation for Cold-Start Multimedia Recommendation?

Xiaoyu Du
National University of Singapore
duxy.me@gmail.com

Xiang Wang*
National University of Singapore
xiangwang@u.nus.edu

Xiangnan He
University of Science and Technology of China
xiangnanhe@gmail.com

Zechao Li
Nanjing University of Science and Technology
zechao.li@njust.edu.cn

Jinhui Tang
Nanjing University of Science and Technology
jinhuitang@njust.edu.cn

Tat-Seng Chua
National University of Singapore
dcscts@nus.edu.sg

ABSTRACT

The ability of recommending cold items (that have no behavior history) is a core strength of multimedia recommendation compared with behavior-only collaborative filtering. To learn effective item representation, a key challenge lies in the discrepancy between training and testing, since the cold items only exist in the testing data. This means that the signal used to represent an item varies during training and testing — in the training stage, we can represent an item with both collaborative embedding and content embedding; whereas in the testing stage, we represent a cold item with content embedding only. Nevertheless, existing learning frameworks omit this critical discrepancy, resulting in suboptimal item representation for multimedia recommendation.

In this work, we pay special attention to cold items in multimedia recommender training. To address the discrepancy, we first represent an item with dual representation, *i.e.,* two vectors where one follows the traditional way that combines collaborative embedding and content embedding, and the other assumes that the item is cold by replacing the collaborative embedding with zero vector. We then propose a *Multi-Task Pairwise Ranking* (MTPR) framework for model training, which enforces the observed interactions ranking higher than the unobserved ones even if the item is assumed to be cold. As a general learning framework, Our MTPR is agnostic to the choice of the collaborative (and/or content) encoder. We demonstrate it on VBPR, a representative multimedia recommendation model based on matrix factorization. Extensive experiments on three datasets of diverse domains validate MTPR, which leads to better representation for both cold and non-cold items in the testing stage, thus improving the overall performance of multimedia recommendation. [1]

CCS CONCEPTS

• **Information systems → Recommender systems**.

KEYWORDS

Multimedia, Cold-start Recommendation, Multi-task Learning, Counterfactual Representation Learning

ACM Reference Format:
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 *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3413628

1 INTRODUCTION

Collaborative filtering (CF), as a prevalent technique in personalized recommendation, has been extensively studied. It relies heavily on historical interactions between users and items (*e.g.,* views, clicks, and ratings). The behavioral patterns are usually encoded as collaborative embeddings, which evolve from ID embeddings of matrix factorization [32], history embeddings of item-based encoders [18, 20], to holistic graph embeddings of recent graph-based encoders [4, 13, 44]. Despite the remarkable performance, such behavior-only CF methods suffer from cold-start issues — that is, they cannot generate collaborative embeddings of new-coming or unseen items, which never appear with the observed user interactions during model training.

Compared with behavior-only CF, multimedia recommendation is able to harness rich content of cold items to discover user interests on them. In particular, although no behavior history is available, there are multimodal information (*e.g.,* frames, audios, comments) associated with cold items (*e.g.,* images in Pinterest and Instagram, micro-videos in TikTok and Kuaishou). Early multimedia recommender models [1, 11, 40] exploit such content to generate content embeddings to reflect intrinsic similarity among items, thus transferring user preference from non-cold items to cold ones. One major limitation is that it ignores the collaborative signal, which leads to suboptimal representation ability. Having realized the limitation, recent works like VBPR [12], ACF [5], and MMGCN [47] incorporate collaborative embeddings into content embeddings to yield better item representations. Considering the content similarity of items and behavioral similarity of users simultaneously enable these methods to reconstruct the historical interactions well.

*Xiang Wang is the corresponding author.
[1] We release the experimental codes at: https://github.com/duxy-me/MTPR.

MM '20, October 12–16, 2020, Seattle, WA, USA
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3413628

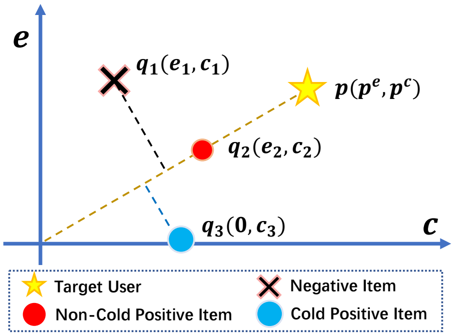


Figure 1: A demo space for the VBPR features. Owing to the discrepancy between training and testing, the cold item $\mathbf{q}_3$ is assigned with lower confidence than the negative non-cold $\mathbf{q}_1$, while $\mathbf{q}_3$ is more similar to the positive $\mathbf{q}_2$ *w.r.t.* content.

Nevertheless, we argue that these prior efforts [5, 12, 47] forgo the key challenge of discrepancy between training and testing — the signal used to represent a non-cold item in training and a cold item in testing varies significantly. To be more specific, in the training phase, these methods can represent an observed item with both collaborative embedding and content embedding, whereas in testing they have to use the zero vector as the collaborative embedding of a cold item. Clearly, such critical discrepancy results in inconsistent representations of items and further influences the ranking of cold items negatively, compared with the non-cold items. To illustrate how incomparable the cold and non-cold items are, we show a toy example of VBPR [12] in Figure 1, involving a target user $\mathbf{p}$, a negative non-cold item $\mathbf{q}_1$, a positive non-cold item $\mathbf{q}_2$ learned in the training, and a positive cold item $\mathbf{q}_3$ in the future testing data. Wherein, the coordinates $\mathbf{c}$ and $\mathbf{e}$ indicate the content embedding and collaborative embedding, respectively. For each item, its projection to the target user $\mathbf{p}$ (*i.e.,* the dashed yellow line) is the confidence in matching her interest (*cf.* the inner product adopted in VBPR to estimate user preference on an item). Obviously, owing to the discrepancy between training and testing, the cold item $\mathbf{q}_3$ is assigned with lower confidence than the non-cold but negative item $\mathbf{q}_1$, despite that $\mathbf{q}_3$ is more similar to the positive $\mathbf{q}_2$ *w.r.t.* content. Therefore, as they omit this critical discrepancy, existing learning frameworks result in suboptimal item representations for multimedia recommendation.

In this work, we pay special attention to cold items in multimedia recommender training, in order to address the discrepancy of item representations. Towards this end, we represent an item in training with dual representation, which comprises two vectors: 1) *normal representation*, which follows the traditional way to combine its collaborative embedding and content embedding and 2) *counterfactual representation*, which adopts the idea of *counterfactual thinking* and assumes that it is cold-start by replacing the collaborative embedding with the zero vector. Conceptually, counterfactual representations answer the question: *How would the representation change if one item observed during training becomes a cold item in testing?* and measure the discrepancy. Based on these two vectors, we then propose a new optimization framework, *Multi-Task Pairwise Ranking* (MTPR), for model training.

Given a pair of positive and negative items (say, i and j), we associate it with two pairwise loss branches, each of which corresponds to one optimization task: 1) a pairwise ranking between their normal representations, which remains consistent to the classic BPR (Bayesian Personalized Ranking) loss [32]; and 2) a pairwise comparison between their counterfactual representations, which encourages the positive i to be ranked higher than the negative j, assuming these two items were cold-start during training. Besides, we introduce two regularization terms to make the representations across normal and counterfactual spaces comparable, so as to solve the discrepancy issue. Benefiting from such multi-task learning, our MTPR is able to effectively solve the discrepancy during training and testing. It is worthwhile highlighting that our MTPR is a general learning framework that is agnostic to the choice of the collaborative (and/or content) encoder, which can generate ID-, history-, or graph-based collaborative embeddings. We demonstrate it on VBPR with a matrix factorization (MF) encoder and conduct extensive experiments to validate the effectiveness of recommending cold items. In a nutshell, we summarize the main contributions as:

- We highlight the training-testing discrepancy for cold item representation in multimedia recommendation.
- We propose a novel generic learning framework MTPR, which devises dual item representations consisting of normal and counterfactual representations and optimizes them via four pairwise loss branches to solve the discrepancy.
- Extensive experiments on three public datasets validate MTPR, which leads to better representations for both cold and non-cold items in testing.

2 PRELIMINARY

In this section, we illustrate the technical preliminary. For ease of reading, we use bold uppercase letter to denote a matrix, bold lowercase letter to denote a vector, italic letter to denote a scalar, and calligraphic uppercase letter to denote a set.

2.1 Multimedia Recommendation

Multimedia recommendation methods aim to predict the preference of user u on item i. Recent works mostly take two vectors $\mathbf{p}_u$ and $\mathbf{q}_i$ to represent user u and item i, respectively. Then the preference score of user u on item i could be estimated with the function,

$$\hat{y}_{ui} = \phi(\mathbf{p}_u, \mathbf{q}_i). \tag{1}$$

The function $\phi(\cdot)$, user representation $\mathbf{p}_u$ and item representation $\mathbf{q}_i$ are three core components for recommendation with many variants in different scenarios. In multimedia recommendation, every item is associated with its content embedding, hence the item representation is mostly the combination of its collaborative embedding and content embedding:

$$\mathbf{q}_i = \rho(\mathbf{e}_i, \mathbf{c}_i), \tag{2}$$

where $\rho(\cdot)$ indicates the combination function, $\mathbf{e}_i$ indicates the collaborative embedding, and $\mathbf{c}_i$ indicates the content embedding which is usually defined as $\mathbf{c}_i = \mathbf{W} \cdot \mathbf{t}_i$, where $\mathbf{t}_i \in \mathbb{R}^F$ is the content feature of item i and $\mathbf{W} \in \mathbb{R}^{D \times F}$ denotes the projection matrix. In this paper, we take this as the definition of $\mathbf{c}_i$ by default.

Additionally, we introduce a typical implementation of $\rho(\cdot)$ as,

$$\rho(\mathbf{e}_i, \mathbf{c}_i) = \mathbf{e}_i \| \mathbf{c}_i, \tag{3}$$

where $\|$ is the vector concatenation operator. This is the item representation used in VBPR [12], and widely used as the raw item representation in recent works [36, 47].

2.2 Learning Paradigm

We use a matrix $\mathbf{R} \in \mathbb{R}^{M \times N}$ to denote the user-item behaviors, where M and N indicate the number of users and items, respectively. The (u, i)-th entry of $\mathbf{R}$ is denoted as r_{ui}, where $r_{ui} = 1$ indicates that the i-th item is in the behavioral history of the u-th user, 0 otherwise. In addition, we use $\mathcal{R} = \{(u, i) | r_{ui} = 1\}$ to denote the set of the observational data. The learning objective is to find an appropriate function $\phi(\mathbf{p}_u, \mathbf{q}_i)$ to predict the preference score $\hat{y}_{ui}$.

In order to optimize the prediction function, most recent works adopt Bayesian Personalized Ranking (BPR)[32], which is based on the personalized hypothesis that a user's preference on the observed interactions should be scored higher than that on the unobserved interactions. Thus it devotes to maximize the margin between the observed and unobserved interactions. Formally, its objective function is,

$$L_{BPR}(\mathcal{D}|\Theta) = \sum_{(u,i,j) \in \mathcal{D}} - \ln \sigma(\hat{y}_{ui} - \hat{y}_{uj}) + \lambda \|\Theta\|^2, \tag{4}$$

where $\mathcal{D} = \{(u, i, j) | (u, i) \in \mathcal{R}, (u, j) \notin \mathcal{R}\}$ indicates the set of training instances, $\sigma(\cdot)$ is the sigmoid function, and λ is the regularization parameters to prevent overfitting.

2.3 Discrepancy between Training and Testing

The key challenge of multimedia recommendation relies on the discrepancy between training and testing. The training process relies on the behavior history, where only the non-cold items could be involved. Hence the cold item representations composed of only content embedding have never been considered by the optimizing methods. However, the testing process compares them with the non-cold item representations, even though they are incomparable theoretically. This causes the suboptimal situation discussed in Figure 1, which severely influences the ranking performances.

Formally, as we introduced in section 2.1, the item is represented as $\mathbf{q}_i = \rho(\mathbf{e}_i, \mathbf{c}_i)$, where $\mathbf{e}_i$ indicates the collaborative embeddings learned from the behavior history. If item i is seen in training, its collaborative embedding is non-zero and informative, *i.e.,* $\mathbf{e}_i \neq \mathbf{0}$; if it is unseen in training, its collaborative embedding is assigned with an all-zero vector *i.e.,* $\mathbf{e}_i = \mathbf{0}$. Obviously, $\rho(\mathbf{0}, \mathbf{c}_i)$ reveals that the representation capability of cold items completely comes from their content embedding. This makes the non-cold and the cold item ranking seem like two separate tasks. To link the two tasks, we devise a multi-task method to bridge the cold and non-cold representations simultaneously.

3 METHOD

In this section, we introduce the proposed **M**ulti-**T**ask **P**airwise **R**anking framework (MTPR). As shown in Figure 2, the core components of MTPR are the dual item representations and the Multi-Task Pairwise Ranking loss function. We will illustrate them here and introduce the training and prediction measures.

3.1 Dual Item Representations

In section 2.3, we demonstrate the discrepancy between training and testing. The main problem is that the cold item representations never appear in the training process. Therefore, we devise the dual item representations to present an item from the perspective of counterfactual thinking. **Normal Representation** (N-rep for short) follows the traditional way and combines the item collaborative embedding and content embedding. **Counterfactual Representation** (C-rep for short) assumes that the item is cold-start and replaces the collaborative embedding with the all-zero vector. Formally, the dual representation of item i are,

$$\mathbf{q}_i^N = \rho(\mathbf{e}_i, \mathbf{c}_i), \quad \mathbf{q}_i^C = \rho(\mathbf{0}, \mathbf{c}_i), \tag{5}$$

where $\mathbf{q}^N$ and $\mathbf{q}^C$ indicate the N-rep and C-rep, respectively, $\mathbf{0} \in \{0\}^D$ is the all-zero vector. In order to make the N-rep and C-rep comparable, they are projected to a common space. Taking VBPR as the example, the dual representations are $\mathbf{q}_i^N = \mathbf{W}^q(\mathbf{e}_i \| \mathbf{c}_i)$ and $\mathbf{q}_i^C = \mathbf{W}^q(\mathbf{0} \| \mathbf{c}_i)$, where $\mathbf{W}^q$ is the projection matrix. Analogously, the user embeddings are projected to the same space via $\mathbf{W}^p$, *i.e.,* $\mathbf{p}_u^o = \mathbf{W}^p \mathbf{p}_u$, where $\mathbf{p}_u$ is his/her original representation. Figure 2 presents the flows, where the N-rep and C-rep components are separately colored red and blue.

The dual representations have some desired properties:

- The N-rep $\mathbf{q}_i^N$ corresponds to the non-cold item representations that are the same as the representations defined in Equation 2. It would inherit the advantages of the traditional models.
- The C-rep $\mathbf{q}_i^C$ corresponds to the item representations when item i is a cold item, which is significantly related to the performance of cold items.
- The N-rep and C-rep of a cold item are identical, *i.e.,* $\mathbf{q}_i^N = \mathbf{q}_i^C$, because the collaborative embedding $\mathbf{e}_i = \mathbf{0}$ when item i is cold.

If the N-rep and C-rep become comparable by modeling their relations during the training process, the non-cold items and cold items could also be comparable according to the properties of the dual representations. Therefore, we could say that we succeed in bridging the representations of non-cold and cold items via the dual representations. Next, we will introduce the Multi-Task Pairwise Ranking to model the relations between N-rep and C-rep.

3.2 Multi-Task Pairwise Ranking

Multi-Task Pairwise Ranking (MTPR) consists of four pairwise branches, to model the correlations between N-rep and C-rep, as shown in Figure 2. Given the training pair $(u, i, j) \in \mathcal{D}$, where $\mathcal{D} = \{(u, i, j) | (u, i) \in \mathcal{R} \wedge (u, j) \notin \mathcal{R}\}$, we capture the dual item representations, where $\mathbf{q}_i^N$ and $\mathbf{q}_i^C$ are the dual representations of the positive item i, and $\mathbf{q}_j^N$, $\mathbf{q}_j^C$ are the dual representations of the negative item j. Then we estimate the user preferences on both items i and j with the normal measure and counterfactual measure,

$$\begin{aligned} \hat{y}_{ui}^N = \phi(\mathbf{p}_u^o, \mathbf{q}_i^N), \quad \hat{y}_{ui}^C = \phi(\mathbf{p}_u^o, \mathbf{q}_i^C), \\ \hat{y}_{uj}^N = \phi(\mathbf{p}_u^o, \mathbf{q}_j^N), \quad \hat{y}_{uj}^C = \phi(\mathbf{p}_u^o, \mathbf{q}_j^C). \end{aligned} \tag{6}$$

Following the basic idea in BPR, we aim to score the positive item higher than the negative one. Thus we organize four pairwise comparisons as,

$$\begin{aligned} L_{NN}(u, i, j) = - \ln \sigma(\hat{y}_{ui}^N - \hat{y}_{uj}^N), \quad L_{CC}(u, i, j) = - \ln \sigma(\hat{y}_{ui}^C - \hat{y}_{uj}^C), \\ L_{NC}(u, i, j) = - \ln \sigma(\hat{y}_{ui}^N - \hat{y}_{uj}^C), \quad L_{CN}(u, i, j) = - \ln \sigma(\hat{y}_{ui}^C - \hat{y}_{uj}^N). \end{aligned} \tag{7}$$

The four terms reflect two tasks and two constraints:

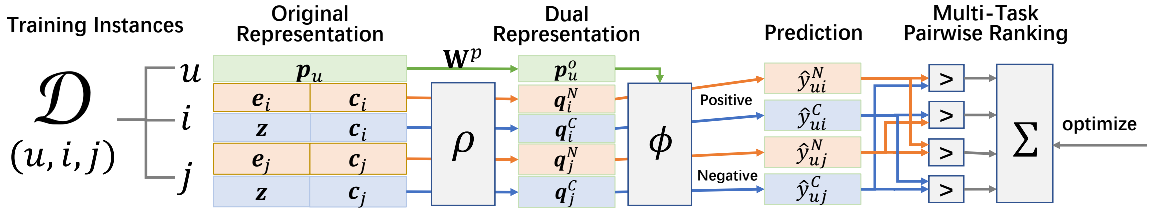


Figure 2: Framework of MTPR. The user, N-rep and C-rep components are colored green, red and blue, respectively.

(1) The term $L_{NN}(u, i, j)$ focuses on the pairwise ranking task on the normal representations such as the VBPR method [12]. Due to the use of collaborative embedding, $L_{NN}(u, i, j)$ concentrates on the ranking task on the non-cold items.
(2) The term $L_{CC}(u, i, j)$ indicates the pairwise ranking task on the counterfactual representations that consist of content embedding, such as the DUIF method [11]. The counterfactual representation performs consistently for the non-cold items and cold items, thus $L_{CC}(u, i, j)$ tries to deal with the overall ranking task.
(3) The remaining two terms $L_{NC}(u, i, j)$ and $L_{CN}(u, i, j)$ are the key to build the correlations between N-rep and C-rep. They constraint the comparison across the normal and counterfactual representations when optimizing the previous two tasks.

Finally, with the four terms, the objective function of MTPR is,

$$
\begin{aligned}
L_{MTPR}(\mathcal{D}|\Theta_*) = &\sum_{(u,i,j)\in\mathcal{D}} L(u, i, j) \\
&+\lambda_e\|\Theta_e\|^2 + \lambda_w\|\Theta_w\|^2 + \lambda_q\|\Theta_q\|^2,
\end{aligned}
\tag{8}
$$

where λ_* denote the regularization factors, Θ_e indicates the embeddings, Θ_q indicates the item projection matrix, while Θ_w indicates the rest of projection parameters, and

$$
\begin{aligned}
L(u, i, j) = &L_{NN}(u, i, j) + L_{CC}(u, i, j) \\
&+ L_{NC}(u, i, j) + L_{CN}(u, i, j).
\end{aligned}
\tag{9}
$$

3.3 Training and Prediction

To explore how the framework works, we demonstrate MTPR on VBPR [12], which uses a classic matrix factorization (MF) encoder.

Initialization. At the beginning of the training phase, the user collaborative embeddings $\mathbf{p}_u$ is randomly initialized, and the item collaborative embeddings $\mathbf{q}_i$ are initialized according to the type of the items. For the non-cold items, their collaborative embeddings are initialized randomly as usual. For the cold items, since they are not interacted by any users, their collaborative embeddings are initialized with zero vectors, which indicate they do not have collaborative information. The remaining projection matrices, such as $\mathbf{W}$, $\mathbf{W}^p$ and $\mathbf{W}^q$, are randomly initialized.

Optimization. According to the Equations 2 and 5, there are two sets of parameters to be trained in our models, where $\mathcal{E} = \{\mathbf{p}_u, \mathbf{e}_i\}$ is the set of collaborative embeddings functioning on specific user or item, while $\mathcal{W} = \{\mathbf{W}, \mathbf{W}^p, \mathbf{W}^q\}$ is the set of projection matrices shared by all users or items. Since their gradients would have different characteristics, during the training process, we use different optimizers with different learning rates with regularization factors to optimize.

Table 1: Statistics of the experimental datasets.

Dataset	Amazon	Tiktok	Movielens
#User	27,044	28,750	55,485
#Item	86,506	38,102	5,986
#Interaction	201,279	449,593	1,184,023
#Cold Item	17,696	6,171	2,000
Cold Ratio	20.5%	16.2%	33.4%
Visual Dimension	64	128	128
Acoustic Dimension	-	128	128
Textual Dimension	-	-	100

Ranking. To recommend the appropriate items for a specific user u, the items are ranked via their user preference score, which is computed via the item N-rep in our model. The characteristics discussed in Section 3.1 reveal that the N-rep is the same as the C-rep for cold items, while it consists of more information for the non-cold items. Thus, the preference scores are computed with the unified prediction function,

$$
\hat{y}_{ui} = \hat{y}_{ui}^N =< \mathbf{p}_u, \mathbf{q}_i^N > . \tag{10}
$$

4 EXPERIMENTS

As the key contribution of MTPR is the dual item representation and the MTPR loss, we conduct abundant experiments to answer the following research questions,

RQ1 How does our MTPR perform compared with the state-of-the-art methods?
RQ2 How do the dual item representations (optimized with MTPR) contribute to the model performance?
RQ3 How do the multi-modal content features impact the results?
RQ4 How does MTPR optimize an effective model?

At the beginning, we introduce the experimental settings including the datasets, evaluation protocols, baselines and parameter settings.

4.1 Experimental Settings

4.1.1 Datasets. We conduct our experiments on three public datasets, Amazon, Tiktok, and Movielens, the statistics of which are listed in Table 1. In order to generate cold-start validation and testing items, we have re-organized the datasets.

Amazon[2] records the visiting histories on the products of Amazon. We use the men's clothing category to evaluate the task of image-based recommendation. Following previous work [12], we take the review histories as implicit feedback and remove the users with less than five interactions. Every item is correlated to an image, which is originally represented by sparse CNN features.

[2] http://jmcauley.ucsd.edu/data/amazon/.

We use PCA [48] to compact them and finally obtain a 64-d visual feature for each image.

Tiktok[3] records the viewing history on the micro-videos of Tiktok. The dataset provides visual and acoustic features officially. We randomly sample 100,000 items from the original dataset, collect their interactions, and remove the users with less than five interactions and the items with less than two interactions. We randomly sample 6,000 items to be cold-start items, the interactions of which are removed from the training set. In addition, after splitting the dataset, we find 171 unpopular items turn out to be unseen in the training set. Thus, the number of cold items is 6,171.

Movielens[4] records the visiting history on movies information. We generate the multimodal features from the pictures, voices, and titles following previous work [47]. We also compact the visual features to 128-d vectors via PCA. We randomly sample 2,000 items to be cold-start items, the interactions of which are removed from the training set.

4.1.2 Evaluation Protocols. For each user, we randomly sample 80%, 10%, and 10% interactions for training, validation and testing, respectively. Cold items only exist in validation and testing sets. We ensure that each user has at least one item in validation and testing sets, respectively. These observed interactions are treated as positive samples. Besides, we randomly sample some negative samples and pair totally 1,000 items for each user in validation and testing, respectively. Each method generate their predictions for the user-item pairs.

We adopt two metrics to evaluate the predictions, Area Under Curve (AUC) [12] and Normalized Discounted Cumulative Gain (NDCG) [14]. AUC reflects the global improvements over the testing items and NDCG@K reflects the improvements over Top-K ranks. In our experiments, we compute the AUC, NDCG@5, and NDCG@10 for each user and report the average scores. Particularly, in order to explore the impacts on non-cold and cold items, we separate the positive items as non-cold positive items and cold positive items, and report their average scores separately.

4.1.3 Baselines. Our MTPR is a general learning framework agnostic to the choice of the collaborative (and/or content) encoder. To justify the effectiveness of our method, We demonstrate it on VBPR, a representative multimedia recommendation model based on matrix factorization. To make a fair comparison,we select the state-of-the-art matrix factorization based multimedia recommendation methods as our baseline methods,

- **BPR** [32] represents the items with the collaborative embeddings and optimizes them with a pair-wise loss. It is a competitive method built on ID-based item representations.
- **CBPR** is a pairwise impelmentation of DUIF [11], which represents the items with the item content. It reflects the impact of item content without collaborative information.
- **VBPR** [12] is the state-of-the-art matrix factorization based method for cold-start items. It incorporates the ID-based embeddings and content-based embeddings in order to give considerations to both the non-cold and cold items.
- **AMR** [36] is a competitive method that introduces adversarial noise in the training process to increase the robustness and performances of multimedia recommendation methods. We follow the original paper and implement the matrix factorization based AMR to obtain the results.

4.1.4 Parameter Setting. We implement the methods based on PyTorch [28]. Empirically, we adopt Adagrad [8] to optimize the user and item embeddings which are private features, and Adam [19] to optimize the rest parameters that are generally used in every prediction. We use the batch size of 512 and the representation dimension D of 32. For all models including baselines and our method, we conduct the grid search for learning rate in $\{1^{-4}, 1^{-3}, 1^{-2}, 1^{-1}\}$. We also tune the regularization factors in Equation 8. λ_e, λ_w and λ_q of $[0, 1^{-4}, 1^{-2}, 1]$ are tested. The collaborative embeddings in all the methods are initialized with random values or pre-trained BPR embeddings. The rest parameters are randomly initialized. We train every method for 100 epochs, use an early-stop strategy referred to the validation scores, and report the corresponding testing scores.

4.2 Performances Comparison (RQ1)

We select the set of parameters that attain the best NDCG@5 score in the validation set, and report their testing results in Table 2. From the table, we have the following observations,

(1) MTPR outperforms BPR, CBPR, VBPR, and AMR in all the cases. Although it adopts VBPR as the original representations, with the multi-task pairwise ranking loss, MTPR achieves a high improvement on both global performance (*i.e.,* AUC) and top ranks (*i.e.,* NDCG). In addition, MTPR not only improves the cold representations but also does well in non-cold items. This verifies that MTPR is an effective measure to bridge the collaborative information and content embeddings, and offers simultaneous considerations to both non-cold and cold items.

(2) Generally, the models incorporating collaborative embeddings have higher NDCG scores. The situation reflects the advantages of collaborative information on Top-K recommendations. In contrast, CBPR, the model that uses content embeddings only, outperforms other methods on the NDCG scores of cold items. The situation reflects the importance of content embeddings to cold items. According to the effect of the two representations, the key to cold-start recommendation is therefore to effectively incorporate the two representations. In such way, MTPR does well compared with the baseline methods.

(3) The representations incorporating the content embeddings could not only improve the cold item representations but also promote the non-cold item representations. The typical signal is that VBPR and MTPR outperform BPR in most non-cold evaluation cases. The exceptions are caused by the model selection measure, because we select the results with the highest overall NDCG@5, which would be significantly impacted by the cold item scores. In addition, the improvements on non-cold items correspond to the dataset. The incorporation of content embeddings significantly benefits the model for Amazon, while it rarely affects the models for Tiktok and Movielens. This phenomenon is caused by the number of interactions in the datasets. On average, the items in Amazon have less than three interactions, while the items in Tiktok and

[3] http://ai-lab-challenge.bytedance.com/tce/vc/.

[4] https://grouplens.org/datasets/movielens/.

Table 2: **The performances of the recommendation methods. non-cold, Cold, and All indicate the performance of the sets of non-cold items, cold items and overall items, respectively. RI is the relative improvement of MTPR over baselines. The best score and second best score are annotated with bold font and underline, respectively.**

Metric	Model	Amazon				Tiktok				Movielens			
		Non-Cold	Cold	**All**	RI (%)	Non-Cold	Cold	**All**	RI (%)	Non-Cold	Cold	**All**	RI (%)
AUC	BPR	0.5434	0.5047	0.5310	44.4	0.8662	0.6879	0.7670	15.0	0.9435	0.6231	0.8074	10.5
	CBPR	0.7002	0.7214	0.7066	8.5	0.6913	0.6826	0.6864	28.6	0.8026	0.6681	0.7478	19.3
	VBPR	0.6981	0.7218	0.7054	8.7	0.8930	0.7501	0.8119	8.7	0.9503	0.6696	0.8317	7.3
	AMR	0.6753	0.7104	0.6867	11.7	0.8975	0.7592	0.8189	7.8	0.9516	0.6758	0.8350	6.9
	MTPR	0.7603	0.7816	**0.7669**	-	0.8691	0.8940	**0.8824**	-	0.9347	0.8380	**0.8922**	-
NDCG@5	BPR	0.0583	0.0000	0.0393	65.0	0.1941	0.0000	0.0961	7.4	0.2101	0.0000	0.1445	13.1
	CBPR	0.0314	0.0395	0.0344	88.4	0.0208	0.0210	0.0240	329.6	0.0754	0.0141	0.0608	168.9
	VBPR	0.0437	0.0368	0.0425	52.3	0.1946	0.0018	0.0969	6.6	0.2160	0.0004	0.1479	10.4
	AMR	0.0636	0.0145	0.0485	33.6	0.1968	0.0027	0.0987	4.6	0.2191	0.0006	0.1512	8.1
	MTPR	0.0843	0.0213	**0.0648**	-	0.1950	0.0132	**0.1032**	-	0.2368	0.0030	**0.1634**	-
NDCG@10	BPR	0.0700	0.0000	0.0468	69.3	0.2278	0.0000	0.1127	10.8	0.2494	0.0000	0.1687	12.5
	CBPR	0.0424	0.0519	0.0460	72.4	0.0287	0.0291	0.0330	278.2	0.0948	0.0206	0.0759	150.1
	VBPR	0.0557	0.0493	0.0541	46.5	0.2318	0.0030	0.1152	8.5	0.2584	0.0009	0.1743	8.9
	AMR	0.0749	0.0236	0.0590	34.4	0.2346	0.0049	0.1177	6.1	0.2629	0.0010	0.1778	6.7
	MTPR	0.1004	0.0327	**0.0793**	-	0.2263	0.0241	**0.1249**	-	0.2787	0.0065	**0.1898**	-

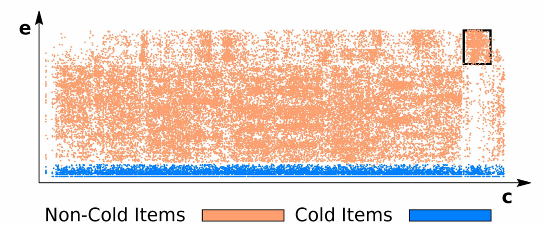


Figure 3: The t-SNE visualization of VBPR item representations in TikTok. c- and e-axis indicate the content embeddings and collaborative embeddings, respectively.

Movielens have more than 11 interactions. More interactions bring stronger collaborative information, so the effect of extra content embeddings is less.

4.3 Effects of MTPR (RQ2)

The core components of MTPR is the dual item representation and the multi-task pairwise ranking loss. We conduct the following experiments to explore the effectiveness of the two components.

4.3.1 Effectiveness of Dual Item Representations. The dual item representation is composed of two vectors, N-rep and C-rep, and both are used in the training process. In the testing process, the final item representation is the N-rep, as discussed in section 3.1. In order to demonstrate the properties of item representations, we visualize the items representations of BPR, VBPR and MTPR.

Firstly, we take Tiktok dataset to show the visualization of VBPR item representations in Figure 3. We use t-SNE [26] to project the content embeddings and collaborative embeddings of Tiktok to 1-d space separately, and draw the points on the 2-d map. The **c**-axis and **e**-axis indicate the content embeddings and the collaborative embeddings, respectively. The blue points and the orange points indicate the cold items and the non-cold items, respectively. From the figure, we have three key observations,

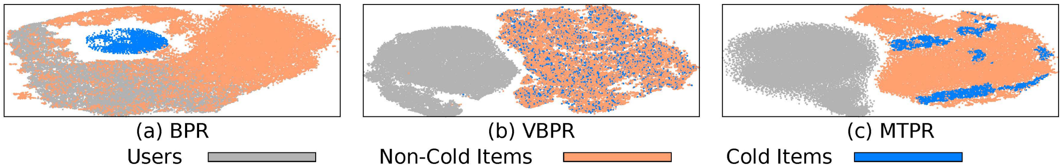
- The cold items are completely detached from the non-cold items in the **e**-axis, which means the lack of collaborative embedding of cold items makes them incomparable with the non-cold items.
- The situation that both the cold items and non-cold items cover most of the **c** range indicates that most of the cold items have similar content as the non-cold items. However, due to the limitation of **e**, their representations are separated from their similar non-cold items.
- The collaborative embeddings and the content embeddings of the non-cold items show obvious correlations, such as the cluster in the black rectangle at the top-right corner of the point map. This cluster corresponds to a set of items which consist of content and collaborative features. However, the cold items that have similar content embeddings are located far away from the rectangle and unable to take advantage of the collaborative knowledge. This is another evidence of the limitation of VBPR representation.

Next, we also use the Tiktok dataset to show the user and item distributions from BPR, VBPR and MTPR in Figure 4. We leverage t-SNE [26] to project the user and item representations to a 2-d space, and plot them in a point map. The blue, orange and gray points indicate the cold items, non-cold items, and the users, respectively.

According to the meaning of t-SNE [26], a cluster of the points reflect the similarity among them. The situation that the points of users and items belong to different clusters indicates that all the methods have produced informative representations. However, Due to the different item distributions of the methods, the effects of their representations are much different. We then analyze their item distributions respectively.

BPR The cold items are detached from the non-cold items. In most cases, they would be treated as unrelated sets. Hence, no effective connections between the non-cold items and cold items could be captured. However, the observation that the cold items are surrounded by the users and non-cold items indicate that they still have a reasonable position in the rank and could contribute a bit to the results.

VBPR As discussed in Figure 3, the non-cold items with similar content embeddings would cluster, such as the items in the black rectangle. Thus the cold items with similar content embeddings should cluster well too. In other words, the cluster indicates the effectiveness of item correlations. However, the cold items


(a) BPR

(b) VBPR

(c) MTPR

Figure 4: The item distribution visualized by t-SNE. The ideal distributions is the items would have obvious clusters and the non-cold items and cold items could be mixed, as the distribution of MTPR.

Table 3: AUC of VBPR, SBPR, DBPR and MTPR.

Model	Amazon			Tiktok			Movielens		
	Non-Cold	Cold	All	Non-Cold	Cold	All	Non-Cold	Cold	All
VBPR	0.646	0.652	0.648	0.933	0.705	0.838	0.879	0.748	0.807
SBPR	0.715	0.616	0.683	0.865	0.804	0.831	0.957	0.613	0.809
DBPR	0.752	0.688	0.731	0.890	0.790	0.834	0.909	0.772	0.854
MTPR	0.760	0.782	**0.767**	0.869	0.894	**0.882**	0.935	0.838	**0.892**

represented by VBPR are scattered randomly on the cloud map. That means that in such a space, the correlations of the cold items are meaningless, and are incomparable with the non-cold items.

MTPR The apparent cluster of cold items indicates that the correlations inside the cold items are effectively represented. In addition, the situation that the cold items are blended with the non-cold items means that MTPR has successfully modeled the correlations between the cold and non-cold items. This verifies the effectiveness of MTPR.

4.3.2 Effectiveness of multi-task pairwise ranking. In order to verify the necessity of the four pairwise ranking branches, we devise two simplified training loss named Single-branch BPR (SBPR) and Double-branch BPR (DBPR), where SBPR optimizes the models with only L_{NN} and DBPR optimizes the models with L_{NN} and L_{CC}. Actually, SBPR indicates the single task solution, similar to VBPR [12]. DBPR indicates the double-task solution without the constraint terms, as a simple composition of the tasks of VBPR [12] and CBPR [11].

From the AUC scores of these training loss on the three datasets presented in Table 3, we have two observations. 1) The performances of SBPR and VBPR are similar because SBPR only linearly projects the VBPR representations to a common space. This suggests that the projection might not the key to improve the item representations; 2) The situation that DBPR outperforms SBPR on Amazon and Movielens verifies the effectiveness of introducing C-rep into the training process. With DBPR, the N-rep and C-rep could transfer their information via their interacted user and their shared content embeddings. 3) Lacking the comparison across the non-cold and cold items limits the improvement of DBPR, which is verified by its worse performance than MTPR over all datasets.

4.4 Multimodal Features (RQ3)

The multimodal features are significant for the predicting results, since they are the only features for cold items. We explore the impact of different modal features on Movielens by using different combination of modal features. We train the models with acoustic features, visual features and textual features, and their combination, respectively, and list the results in Table 4.

The results reveal the contributions of the three modal features for MTPR. The NDCG scores show that the acoustic and visual features could help the cold items increase their ranks.

Table 4: Impact of multimodal features tested on Movielens. A-F, V-F, and T-F indicate the Acoustic feature, Visual feature, and Textual feature, respectively. C-F indicates the feature composed of A-F, V-F, and T-F.

Feat	AUC			NDCG@5			NDCG@10		
	Non-Cold	Cold	All	Non-Cold	Cold	All	Non-Cold	Cold	All
A-F	0.9306	0.8194	0.8830	0.2210	0.0014	0.1534	0.2641	0.0038	0.1804
V-F	0.9370	0.7890	0.8731	0.2284	0.0015	0.1573	0.2721	0.0026	0.1848
T-F	0.9273	0.7941	0.8692	0.2335	0.0001	0.1611	0.2780	0.0008	0.1878
C-F	0.9347	0.8380	**0.8922**	0.2368	0.0030	**0.1634**	0.2787	0.0065	**0.1898**

Table 5: Parameter initialization with Random Values vs Parameter Initialization with Pre-trained collaborative Embedding. R and P indicate the *R*andom initialization and *P*retrained initialization, respectively.

Metric	Init	Amazon			Tiktok			Movielens		
		Non-Cold	Cold	**All**	Non-Cold	Cold	**All**	Non-Cold	Cold	**All**
AUC	R	0.760	0.782	0.767	**0.883**	0.888	**0.885**	0.937	0.803	0.879
	P	**0.766**	**0.825**	**0.785**	0.869	**0.894**	0.882	**0.935**	**0.838**	**0.892**
NDCG@5	R	**0.084**	**0.021**	**0.065**	0.173	0.013	0.093	0.190	**0.004**	0.133
	P	0.075	0.020	0.058	**0.195**	**0.013**	**0.103**	**0.237**	0.003	**0.163**
NDCG@10	R	**0.100**	**0.033**	**0.079**	0.207	0.024	0.116	0.232	**0.008**	0.159
	P	0.090	0.032	0.071	**0.226**	**0.024**	**0.125**	**0.279**	0.007	**0.190**

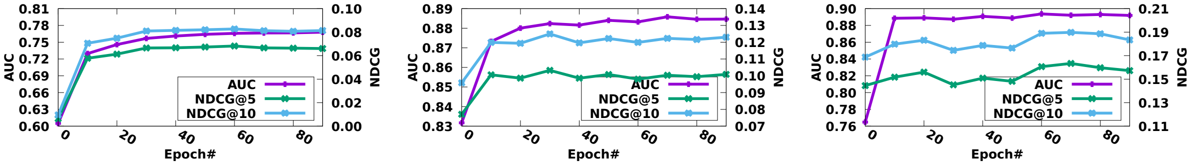
Correspondingly, their AUC scores are better than that of the textual feature. The non-cold test of textual features on NDCG outperforms those of the acoustic and visual features. This means that the textual features make the non-cold items more distinguishable at the top ranks. Overall, by combining all three modalities, the model could achieve the best performance on all metrics.

4.5 Training Details (RQ4)

We discuss the impact of different initialization measures, and present the training curves to illustrate the training process.

4.5.1 Parameter Initialization. We empirically adopt two measures to initialize the collaborative embeddings and list the results in Table 5. Measure R initializes the embedding with random values, while measure P uses the embeddings trained by BPR to directly introduce the informative features from the user-item interactions. The results reveal that the impacts of the initialization measures rely on the quality of the pre-trained embeddings. As discussed in Table 2, BPR performs poor on Amazon, indicating that its trained embeddings might contain some noisy information which impacts the NDCG scores on Amazon negatively. In contrast, the NDCG scores on the other two datasets are significantly improved by introducing the informative embeddings.

4.5.2 Training Process. In Figure 5, we plot the curve of the testing performances of the models reported in Table 2. The models are evaluated at every 10 epochs. All the curves change in the same trend. They achieve a nice performance in 10 epochs, oscillate slightly in the following training process, and go stable eventually. This verifies that MTPR could converge stably and it is easy to train.



(a) Amazon

(b) Tiktok

(c) Movielens

Figure 5: The scores of MTPR on the test data during the training process.

5 RELATED WORKS

5.1 Multimedia Recommendation

As a key branch of recommendation approaches [37, 43], collaborative filtering (CF) generates personalized collaborative features by analyzing the user-item interaction history [20, 50]. The core is to present the users and items with latent factor vectors (*i.e.,* embeddings), which are mostly optimized via the matrix factorization methods [15, 24, 39]. Recently, deep neural networks boost the emergence of generalized matrix factorization methods which take in the embeddings and predict user preferences via neural networks. NCF [14] proposes a two-pathway MLP neural network to generate the predictions. ConvNCF [6] proposes a convolutional neural network to deeply explore the ability of embedding representations. Graph neural networks are also used to capture better collaborative features [44, 45].

In multimedia recommendation tasks, the items are always associated with multimedia content [7, 23, 35], which could accurately describe the characteristics of the items themselves [29, 38]. DUIF [11] takes CNN features to represent the images. Kristoffersen *et al.* [21] propose a joint embedding composed of content and context. MMGCN [47] introduces the graph neural network in the multimedia domain by leveraging the multimedia content embeddings. Yang *et al.* [49] build multiple graphs to improve the multimedia features. Huang *et al.* [16] use a knowledge graph to enhance the explainable sequential recommender. Wang *et al.* [46] recommend tags according to the social properties of users and tags. However, these methods are mostly built on non-cold items and neglect the performances of cold items.

5.2 Cold-start Recommendation

The items that are not interacted by users are termed cold-start items or cold items. They widely exist in online platforms. As they perform much different from the non-cold items in collaborative perspective, to rank the two types of items jointly has been a significant challenge. Vartak *et al.* [41] figure that the personalized recommendation problem is equivalent to a few-shot learning task [9, 17, 30], where the representations of users and items could be generated through a small amount of interaction history (*i.e.,* support data) in the warm-up process [22, 27]. However, due to the necessity of the warm-up process, these methods are not suited for the completely cold items.

As the collaborative embeddings represent the strong collaborative representations for recommendation, many works attempt to link the users and the cold items via context relations [2, 33, 40]. These structures severely rely on the external information. Notable that the content embeddings are very common in multimedia recommendation tasks and they could effectively represent the completely cold items. Some works try to map the collaborative embeddings with the input of content embeddings [3, 10]. In practice, the collaborative and content embeddings reflect two different perspectives of the items. Thus, VBPR [12] concatenates the two kinds of embeddings in prediction and achieves a great performance. AMR [36] improves the VBPR representations by using adversarial training [31, 51].

Some methods noticed the influence of varying and missing features [25, 34, 42]. For example, DropoutNet [42] leverages a dropout-like structure to balance the strengths between collaborative and content features. We argue that this data-augmentation way is insufficient to entangle the relationships among collaborative and content embeddings. Distinct from these methods, we propose the multi-task pair-wise ranking method to address the problem from the counterfactual perspective.

6 CONCLUSION

In this paper, we proposed a new framework MTPR to enhance the cold-start multimedia recommendation. Specifically, we first devised dual item representations consisting of two vectors, where N-rep indicates the normal representation that corresponds to all item representations and C-rep indicates the counterfactual representation that assumes the item is cold. We then proposed multi-task pairwise ranking, a loss function composed of four pairwise ranking branches, to optimize the representations with the target of modeling the correlations between non-cold and cold items. Experimental results on three public multimedia datasets verify the effectiveness of our method.

This work presents a fundamental step towards effectively fusing collaborative and content information for multimedia recommendation. As many models take collaborative and content embeddings to represent an item for multimedia recommendation, we plan to extend the influence of MTPR on a wider range of models, such as neural collaborative filtering and graph-based models. Moreover, we will involve more content information, such as multimedia conversations with users and multimedia knowledge graph of items, to assist the cold-start recommendation. In addition, we would like to explore the challenges inherent in cold-start recommendation, such as algorithmic fairness, evaluation metrics, and long-tail issues.

ACKNOWLEDGEMENTS

This research is supported by the National Research Foundation, Singapore under its International Research Centres in Singapore Funding Initiative, the National Key Research and Development Program of China under Grant 2018AAA0102002, and the National Natural Science Foundation of China (Grant No. 61732007, and U19A2079). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore.

REFERENCES

[1] Iman Barjasteh, Rana Forsati, Farzan Masrour, Abdol-Hossein Esfahanian, and Hayder Radha. 2015. Cold-start item and user recommendation with decoupled completion and transduction. In *RecSys*. 91–98.

[2] Iman Barjasteh, Rana Forsati, Dennis Ross, Abdol-Hossein Esfahanian, and Hayder Radha. 2016. Cold-start recommendation with provable guarantees: A decoupled approach. *TKDE* 28, 6 (2016), 1462–1474.

[3] Oren Barkan, Noam Koenigstein, Eylon Yogev, and Ori Katz. 2019. CB2CF: a neural multiview content-to-collaborative filtering model for completely cold item recommendations. In *RecSys*. 228–236.

[4] Rianne van den Berg, Thomas N Kipf, and Max Welling. 2017. Graph convolutional matrix completion. *arXiv preprint arXiv:1706.02263* (2017).

[5] Jingyuan Chen, Hanwang Zhang, Xiangnan He, Liqiang Nie, Wei Liu, and Tat-Seng Chua. 2017. Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention. In *SIGIR*. 335–344.

[6] Xiaoyu Du, Xiangnan He, Fajie Yuan, Jinhui Tang, Zhiguang Qin, and Tat-Seng Chua. 2019. Modeling Embedding Dimension Correlations via Convolutional Neural Collaborative Filtering. *TOIS* 37, 4 (2019), 1–22.

[7] Xiao-Yu Du, Yang Yang, Liu Yang, Fu-Min Shen, Zhi-Guang Qin, and Jin-Hui Tang. 2017. Captioning videos using large-scale image corpus. *Journal of Computer Science and Technology* 32, 3 (2017), 480–493.

[8] John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research* 12, Jul (2011), 2121–2159.

[9] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*. JMLR. org, 1126–1135.

[10] Zeno Gantner, Lucas Drumond, Christoph Freudenthaler, Steffen Rendle, and Lars Schmidt-Thieme. 2010. Learning attribute-to-feature mappings for cold-start recommendations. In *2010 IEEE International Conference on Data Mining*. IEEE, 176–185.

[11] Xue Geng, Hanwang Zhang, Jingwen Bian, and Tat-Seng Chua. 2015. Learning image and user features for recommendation in social networks. In *ICCV*. 4274–4282.

[12] Ruining He and Julian McAuley. 2016. VBPR: visual bayesian personalized ranking from implicit feedback. In *AAAI*.

[13] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, YongDong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. In *SIGIR* (Virtual Event, China) *(SIGIR '20)*. Association for Computing Machinery, New York, NY, USA, 639–648. https://doi.org/10.1145/3397271.3401063

[14] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *WWW*. 173–182.

[15] Xiangnan He, Hanwang Zhang, Min-Yen Kan, and Tat-Seng Chua. 2016. Fast matrix factorization for online recommendation with implicit feedback. In *SIGIR*. 549–558.

[16] Xiaowen Huang, Quan Fang, Shengsheng Qian, Jitao Sang, Yan Li, and Changsheng Xu. 2019. Explainable Interaction-driven User Modeling over Knowledge Graph for Sequential Recommendation. In *MM*. 548–556.

[17] Muhammad Abdullah Jamal and Guo-Jun Qi. 2019. Task agnostic meta-learning for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 11719–11727.

[18] Santosh Kabbur, Xia Ning, and George Karypis. 2013. Fism: factored item similarity models for top-n recommender systems. In *KDD*. 659–667.

[19] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[20] Yehuda Koren. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *KDD*. 426–434.

[21] Miklas S Kristoffersen, Jacob L Wieland, Sven E Shepstone, Zheng-Hua Tan, and Vinoba Vinayagamoorthy. 2019. Deep Joint Embeddings of Context and Content for Recommendation. *arXiv preprint arXiv:1909.06076* (2019).

[22] Hoyeop Lee, Jinbae Im, Seongwon Jang, Hyunsouk Cho, and Sehee Chung. 2019. MeLU: Meta-Learned User Preference Estimator for Cold-Start Recommendation. In *KDD*. 1073–1082.

[23] Zechao Li and Jinhui Tang. 2015. Weakly supervised deep metric learning for community-contributed image retrieval. *IEEE Transactions on Multimedia* 17, 11 (2015), 1989–1999.

[24] Defu Lian, Cong Zhao, Xing Xie, Guangzhong Sun, Enhong Chen, and Yong Rui. 2014. GeoMF: joint geographical modeling and matrix factorization for point-of-interest recommendation. In *KDD*. 831–840.

[25] Fan Liu, Zhiyong Cheng, Changchang Sun, Yinglong Wang, Liqiang Nie, and Mohan Kankanhalli. 2019. User Diverse Preference Modeling by Multimodal Attentive Metric Learning. In *Proceedings of the 27th ACM International Conference on Multimedia*. 1526–1534.

[26] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, Nov (2008), 2579–2605.

[27] Feiyang Pan, Shuokai Li, Xiang Ao, Pingzhong Tang, and Qing He. 2019. Warm Up Cold-start Advertisements: Improving CTR Predictions via Learning to Learn ID Embeddings. In *SIGIR*. 695–704.

[28] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *NeurIPS*. 8024–8035.

[29] Guo-Jun Qi. 2016. Hierarchically gated deep networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2267–2275.

[30] Guo-Jun Qi, Wei Liu, Charu Aggarwal, and Thomas Huang. 2016. Joint intermodal and intramodal label transfers for extremely rare or unseen classes. *IEEE transactions on pattern analysis and machine intelligence* 39, 7 (2016), 1360–1373.

[31] Guo-Jun Qi, Liheng Zhang, Hao Hu, Marzieh Edraki, Jingdong Wang, and Xian-Sheng Hua. 2018. Global versus localized generative adversarial nets. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1517–1525.

[32] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618* (2012).

[33] Martin Saveski and Amin Mantrach. 2014. Item cold-start recommendations: learning local collective embeddings. In *RecSys*. 89–96.

[34] Shaoyun Shi, Min Zhang, Xinxing Yu, Yongfeng Zhang, Bin Hao, Yiqun Liu, and Shaoping Ma. 2019. Adaptive Feature Sampling for Recommendation with Missing Content Feature Values. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 1451–1460.

[35] Xiangbo Shu, Guo-Jun Qi, Jinhui Tang, and Jingdong Wang. 2015. Weakly-shared deep transfer networks for heterogeneous-domain knowledge propagation. In *MM*. 35–44.

[36] Jinhui Tang, Xiaoyu Du, Xiangnan He, Fajie Yuan, Qi Tian, and Tat-Seng Chua. 2020. Adversarial Training Towards Robust Multimedia Recommender System. *TKDE* 32, 5 (2020), 855–867.

[37] Jinhui Tang, Xiangbo Shu, Zechao Li, Yu-Gang Jiang, and Qi Tian. 2019. Social anchor-unit graph regularized tensor completion for large-scale image retagging. *TPAMI* 41, 8 (2019), 2027–2034.

[38] Jinhui Tang, Xiangbo Shu, Guo-Jun Qi, Zechao Li, Meng Wang, Shuicheng Yan, and Ramesh Jain. 2017. Tri-clustered tensor completion for social-aware image tag refinement. *TPAMI* 39, 8 (2017), 1662–1674.

[39] Dan-Dan Tu, Cheng-Chun Shu, and Hai-Yan Yu. 2013. Using unified probabilistic matrix factorization for contextual advertisement recommendation. *Ruanjian Xuebao/Journal of Software* 24, 3 (2013), 454–464.

[40] Zhen Tu, Yali Fan, Yong Li, Xiang Chen, Li Su, and Depeng Jin. 2019. From fingerprint to footprint: cold-start location recommendation by learning user interest from app data. *IMWUT* 3, 1 (2019), 1–22.

[41] Manasi Vartak, Arvind Thiagarajan, Conrado Miranda, Jeshua Bratman, and Hugo Larochelle. 2017. A meta-learning perspective on cold-start recommendations for items. In *NeuIPS*. 6904–6914.

[42] Maksims Volkovs, Guangwei Yu, and Tomi Poutanen. 2017. Dropoutnet: Addressing cold start in recommender systems. In *Advances in neural information processing systems*. 4957–4966.

[43] Jingdong Wang, Zhe Zhao, Jiazhen Zhou, Hao Wang, Bin Cui, and Guojun Qi. 2012. Recommending Flickr groups with social topic model. *Information retrieval* 15, 3-4 (2012), 278–295.

[44] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural graph collaborative filtering. In *SIGIR*. 165–174.

[45] Xiang Wang, Dingxian Wang, Canran Xu, Xiangnan He, Yixin Cao, and Tat-Seng Chua. 2019. Explainable reasoning over knowledge graphs for recommendation. In *AAAI*, Vol. 33. 5329–5336.

[46] Xueting Wang, Yiwei Zhang, and Toshihiko Yamasaki. 2019. User-Aware Folk Popularity Rank: User-Popularity-Based Tag Recommendation That Can Enhance Social Popularity. In *MM*. 1970–1978.

[47] 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 *MM*. 1437–1445.

[48] Svante Wold, Kim Esbensen, and Paul Geladi. 1987. Principal component analysis. *Chemometrics and intelligent laboratory systems* 2, 1-3 (1987), 37–52.

[49] Xun Yang, Xiaoyu Du, and Meng Wang. 2020. Explainable reasoning over knowledge graphs for recommendation. In *AAAI*.

[50] Yongfeng Zhang, Qingyao Ai, Xu Chen, and W Bruce Croft. 2017. Joint representation learning for top-n recommendation with heterogeneous information sources. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. 1449–1458.

[51] Yiru Zhao, Zhongming Jin, Guo-jun Qi, Hongtao Lu, and Xian-sheng Hua. 2018. An adversarial approach to hard triplet generation. In *ECCV*. 501–517.