Contrastive Learning for Cold-Start Recommendation

Yinwei Wei
weiyinwei@hotmail.com
National University of Singapore

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

Qi Li
iliqi@mail.sdu.edu.cn
Shandong University

Liqiang Nie
nieliqiang@gmail.com
Shandong University

Yan Li
liyan@kuaishou.com
Beijing Kuaishou Technology Co., Ltd.

Xuanping Li
lixuanping@kuaishou.com
Beijing Kuaishou Technology Co., Ltd.

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

ABSTRACT

Recommending purely cold-start items is a long-standing and fundamental challenge in the recommender systems. Without any historical interaction on cold-start items, the collaborative filtering (CF) scheme fails to leverage collaborative signals to infer user preference on these items. To solve this problem, extensive studies have been conducted to incorporate side information of items (e.g., content features) into the CF scheme. Specifically, they employ modern neural network techniques (e.g., dropout, consistency constraint) to discover and exploit the coalition effect of content features and collaborative representations. However, we argue that these works less explore the mutual dependencies between content features and collaborative representations and lack sufficient theoretical supports, thus resulting in unsatisfactory performance on cold-start recommendation.

In this work, we reformulate the cold-start item representation learning from an information-theoretic standpoint. It aims to maximize the mutual dependencies between item content and collaborative signals. Specifically, the representation learning is theoretically lower-bounded by the integration of two terms: mutual information between collaborative embeddings of users and items, and mutual information between collaborative embeddings and feature representations of items. To model such a learning process, we devise a new objective function founded upon contrastive learning and develop a simple yet effective Contrastive Learning-based Cold-start Recommendation framework (CLCRec). In particular, CLCRec consists of three components: contrastive pair organization, contrastive embedding, and contrastive optimization modules. It allows us to preserve collaborative signals in the content representations for both warm and cold-start items. Through extensive experiments on four publicly accessible datasets, we observe that CLCRec achieves significant improvements over state-of-the-art approaches in both warm- and cold-start scenarios.

KEYWORDS

Contrastive Learning, Cold-start Recommendation, Recommender System, Collaborative Filtering, Multimedia Recommendation

1 INTRODUCTION

Collaborative Filtering (CF) has exhibited remarkable success in personalized recommendation. Learning quality representations of users and items from historical interactions (e.g., clicks, rates, views) is central to modern CF models [14, 15, 36, 48]. We denote the representations encoding collaborative signals as collaborative embeddings, such as ID embeddings in MF [36], interaction history embeddings in SVD++ [23], and interaction graph embeddings in LightGCN [14]. Such collaborative embeddings reflect behavioral similarity of users, thus inferring user preferences on items effectively. Clearly, historical user-item interactions are of crucial importance to establish high-quality collaborative embeddings. However, there are a large amount of newly coming items, which never appear in the history, posing the serious challenge of cold-start recommendation. This is because unavailable interaction fails the learning of collaborative embedding for these cold-start items.

To address this problem, one common solution is incorporating item content (e.g., images, videos, attributes) into the CF scheme. The content features are supposed to cooperate with collaborative representations, form the coalition, and offer better predictions. Such coalition effects memorized in CF models allow us to transfer collaborative knowledge to cold-start items. In this research line, extensive studies have been conducted, which can be roughly categorized into two types: robustness-based [8, 34, 38, 43] and constraint-based [2, 3, 26, 37, 54]. Despite success, each type of methods suffers from some inherent limitations:

- From the viewpoint of robust learning, robustness-based methods frame the cold-start items as the corrupted forms of warm
In a nutshell, while being powered by the advance of neural network techniques (e.g., dropout [40] and constraint [54]), these existing methods are insufficient to boost cold-start item recommendation.

In this work, we aim to explore the in-depth and theoretical aspect of cold-start representation learning. Specifically, we reformulate the learning of feature representation as optimizing its maximum posterior estimator under the observed interactions and learned collaborative embeddings. This optimization allows us to distill meaningful information from collaborative embeddings, and encode them into the feature representations for both warm and cold-start items. More formally, the lower bound of the posterior estimator is the integration of two terms: (1) mutual information between user and item collaborative embeddings (short for U-I) and (2) mutual information between feature representation and collaborative embedding (short for R-E). Towards the maximum of mutual information, we propose a novel objective function founded upon contrastive learning [4, 5, 33, 41] from an information-theoretic standpoint.

To achieve our proposed objective function, we further develop a general framework, named Contrastive Learning-based Cold-start Recommendation (CLCRec). Technically, it consists of three key components. (1) We first conduct contrastive pair organization. For the U-I loss, we treat observed historical interactions as positive user-item pairs, while unobserved interactions as negative user-item pairs. Meanwhile, for the R-E loss, we adopt the self-discrimination task of items, where one item paired with itself and other items refer to the positive and negative item-item pairs, respectively. (2) Having established the U-I and R-E contrastive pairs, we build contrastive embedding networks (CEN) upon them. The U-I CEN is to generate the collaborative embeddings for the user-item pairs, which can be built upon any CF encoder like MF and LightGCN. The R-E CEN contains a trainable encoder to learn feature representations of items based on their content features. (3) Afterwards, we perform contrastive optimization according to our designed objective function. Using the pre-defined density ratio functions, we score the constructed U-I and R-E contrastive pairs and then perform contrastive training – that is, identifying the positive pair from multiple negative pairs. As such, two CENs are trained to guide the learning of feature representations and collaborative embeddings, and encourage the information transfer between them. In the testing phase, for a cold-start item, we are able to generate high-quality feature representations that memorize informative signals from observed interactions, and predict how likely a user would adopt it. Through extensive experiments on four real-world datasets, we demonstrate that CLCRec achieves significant improvements over the state-of-the-art approaches in both warm- and cold-start scenarios. We summarize the contributions:

- We reformulate the cold-start representation learning as maximizing the sum of U-I and R-E mutual information.
- Towards mutual information maximum, we design a contrastive learning-based objective function and accordingly propose a general framework, CLCRec.
- We conduct extensive experiments on four public datasets to demonstrate the superiority of CLCRec. We have released the codes and datasets: https://github.com/weiyinwei/CLCRec to facilitate reproducibility.

2 METHODOLOGY

Generally, the cold-start problem explores how to predict the affinity of new items, which has no or few interaction records with users in the training set. In this work, we focus on recommending the complete cold-start items without any interactions, which is more challenging and promising. Before describing the detail of the proposed method, we reformulate the cold-start problem and accordingly derive the objective function.

2.1 Problem Formalization

Learning of Collaborative Embeddings. Let \( O = \{ o_{ui} \} \) be the set of user-item interactions, which involves the set of \( N \) users, \( \mathcal{U} = \{ u \} \) and the set of \( M \) items, \( \mathcal{I} = \{ i \} \). With the assumption that users with similar behavior prefer similar items, modern collaborative filtering (CF) methods [20, 27, 28] learn collaborative embeddings of users and items from the historical interactions. Without loss of generality, we describe an user \( u \) (a item \( i \)) with a collaborative embedding \( z_u \in \mathbb{R}^d \) (\( z_i \in \mathbb{R}^d \)), where \( d \) is the embedding size. Optimizing collaborative embeddings is to maximize the posterior estimator given the observed interactions:

\[
Z^* = \arg \max_Z p(Z|O),
\]  

where \( Z = \{ z_{u1}, \ldots , z_{uN}, z_{i1}, \ldots , z_{iM} \} \) collects the collaborative embeddings of all users and items; \( p(Z|O) \) is to encode as much information on \( O \) into \( Z \) as possible. However, for cold-start items \( i^c = \{ i^c \} \) without any historical interactions, CF is infeasible to optimize their collaborative embeddings, thus fails to predict user preference accurately.

Learning of Feature Representations. To alleviate this problem, extensive studies have been conducted to exploit the rich side information of items, especially content features (e.g., frames of video, soundtracks of music, and descriptions of product). In general, a feature encoder \( f_0 \) is built upon the content feature vector \( x_i \) to
generate the feature representation $f_i$ of item $i$. Here we establish the collections of feature representations and content features as $F = \{f_1, \ldots, f_N\}$ and $X = \{x_1, \ldots, x_N\}$, respectively. Note that, $F$ that are learnable, thus differs from $X$ that are learned by pre-trained feature extractors [29, 46]. Previous works either treat the feature representation and collaborative embeddings individually [8, 34, 38, 43] or apply a regularizer term to encourage their dimension-wise similarity [2, 3, 26, 37, 54].

Unlike these methods, we optimize the feature encoder by encouraging the feature representations to (1) encode collaborative signals from collaborative embeddings and (2) preserve affinities with users who have interacted with before. To this end, we reformulate the item representation learning as maximizing the posterior estimator, given the embeddings and interactions:

$$\theta^* = \arg \max_\theta p(\theta|Z, O, X).$$

Having obtained the optimized feature encoder $\theta^*$, we generate feature representations for cold-start items, which is enhanced with the collaborative knowledge from warm items. Finally, we could infer the interaction probability between a cold-start item and a user by measuring the similarities of their representations.

### 2.2 Objective Function

As the pre-existing content features $X$ are usually independent to the collaborative embeddings $Z$, we can adopt Bayesian rules and rewrite Equation (2) as follows:

$$p(\theta|Z, O, X) \propto p(Z, O|F) \cdot p(\theta).$$

Following the prior study [36], we assume that all items (and users) are independent with each other. As such, we can further obtain the following probability to associate the feature representation with the collaborative embeddings and interactions, respectively:

$$p(Z, O|F) \cdot p(\theta) = \sum_{i \in I} \ln p(z_i, f_i) + \sum_{(u,i) \in O} \ln p(z_u|z_i, f_i) + C \sum_{u \in U} \ln p(z_u) + \ln p(\theta).$$

Here $C$ is a constant term derived from the sum of conditional probabilities: $\sum_{(u,i) \in O} \ln p(z_u|z_i, f_i)$. It captures the intuition that an item is independent of the users who did not interact with before, whereas the presence of an interaction suggests their independence.

In this equation, the first term focuses on the correlation between $z_i$ and $f_i$, such that $z_i$ is encouraged to embed the information of $f_i$. Thus, we could relieve the second term $\sum_{(u,i) \in O} \ln p(z_u|z_i, f_i)$ from the complex relationship among three kinds of representations. Specifically, it is proportional to the probabilities of user embedding, after observing the item embedding:

$$\ln p(z_i, f_i) + \sum_{(u,i) \in O} \ln p(z_u|z_i, f_i) + C \sum_{u \in U} \ln p(z_u) + \ln p(\theta)$$

$$= \sum_{i \in I} \ln p(z_i, f_i) + \sum_{(u,i) \in O} \ln p(z_u|z_i, f_i) + C \sum_{u \in U} \ln p(z_u) + \ln p(\theta).$$

Afterwards, we prove that the lower bound of this posterior probability is the sum of two components: (1) U-I mutual information between collaborative embeddings of users and items, and (2) R-E mutual information between collaborative embeddings and feature representations of items:

$$\ln p(\theta) = \sum_{i \in I} \ln p(z_i, f_i) + \sum_{(u,i) \in O} \ln p(z_u|z_i) + C \sum_{u \in U} \ln p(z_u) + \ln p(\theta)$$

$$\geq \sum_{i \in I} \ln p(f_i, z_i) \ln p(z_u|z_i) + \ln p(z_u) + \ln p(\theta)$$

$$+ \sum_{(u,i) \in O} \ln p(z_u|z_i) + \ln p(z_u) + \ln p(\theta)$$

where $MI(\cdot, \cdot)$ represents the mutual information between two representations. In addition, we use $O$ to represent all the parameters whose prior probabilities presumably follow Gaussian Distributions. Therefore, we cast the objective from the maximum posterior estimator to mutual information maximization.

To maximize the mutual information, we employ the technique of contrastive learning, which models the mutual dependencies of variable pairs by recognizing the positive pair from the negative ones. Specifically, following the prior work [33], we define two density ratio functions:

$$h(z_i, f_i) \propto \frac{p(f_i|z_i)}{p(f_i)} \quad g(z_u, z_i) \propto \frac{p(z_u|z_i)}{p(z_u)}$$

These functions (i.e., $h(\cdot, \cdot)$ and $g(\cdot, \cdot)$) could be used to measure the correlations between two variables. As a result, the contrastive learning-based objective function could be formulated as:

$$\mathcal{L} = \lambda \mathcal{L}_{RE} + (1 - \lambda) \mathcal{L}_{UI} + \eta \|\Theta\|_2^2$$

$$= - \lambda \mathbb{E}_{i \in I} \left[ \ln \frac{h(z_i, f_i)}{h(z_i, f_i) + \sum_{j \in U(i)} h(z_i, f_j)} \right]$$

$$- (1 - \lambda) \mathbb{E}_{(u,i) \in O} \left[ \ln \frac{g(z_u, z_i)}{g(z_u, z_i) + \sum_{(u,j) \in O} g(z_u, z_j)} \right] + \eta \|\Theta\|_2^2;$$
We use $L_{RE}$ and $L_{UI}$ to denote the contrastive loss for maximizing the R-E and U-I mutual information, respectively. As $L_{UI}$ has the same optimal objective with Personalized Ranking (BPR) loss [36], we could optimize the recommendation model in an end-to-end manner without any other loss function, and set a hyper-parameter $\lambda$ to balance the collaborative and feature representations learning.

2.3 CLCRec

According to the objective function, we develop a general framework, Contrastive Learning-based Cold-start Recommendation (CLCRec), as shown in Figure 1. This framework is composed of three components: contrastive pair organization, contrastive embedding network, and contrastive optimization. In what follows, we elaborate the detail of each component.

2.3.1 Contrastive Pair Organization. The core of contrastive learning is to identify the positive pair constructed by semantic similar instances from some negative ones paired by the dissimilar samples. As such, in this work, it is critical to organize the users and items as U-I and R-E contrastive pairs.

U-I Contrastive Pair. We view the user-item pair observed in the historical interactions as the positive sample, like $(u, i)$ shown in Figure 1. Meanwhile, we randomly sample $K$ items (e.g., $(j_1, j_2, \cdots, j_K)$), which have not been purchased by $u$, and pair the user to establish the negative pairs, as illustrated in Figure 1. Formally, the positive and negative U-I pairs can be defined as,

\[
\{(u, i), (u, j_1), (u, j_2), \cdots, (u, j_K)\}. \tag{6}
\]

Compared with the negative pairs, the positive one contains the similar collaborative signal. Hence, such a comparison facilitates the discovery of collaborative signal conveyed by the interactions.

R-E Contrastive Pair. Different from U-I pairs, R-E pairs adopt the self-discrimination task to maximize the mutual information of two different representations of items. To construct the pairs, taking $i$ in Figure 1 as an example, we set it as the anchor item and join the anchor with itself as the positive pair, which reveals the semantic similarity between two representations of the same item. By contrast, the negative pairs organized by the anchor with other items are semantically dissimilar. As shown in Figure 1, we pair the items together to obtain the R-E pairs:

\[
\{(i, i), (i, j_1), (i, j_2), \cdots, (i, j_K)\}, \tag{7}
\]

where $(i, i)$ is the positive pair and the others are negative ones.

2.3.2 Contrastive Embedding Network. Upon contrastive pairs, we devise the U-I and R-E contrastive embedding networks (CENs) to represent the user and item and calculate the correlation of each pair using the pre-defined density ratio functions.

U-I Contrastive Embedding Network. To model collaborative signal from user-item interactions, we first fetch their id embeddings (e.g., $e_u$ and $e_i$) from an look-up table, which is defined by a parameter matrix:

\[
E = [e_{i_1}, \cdots, e_{i_{|N_i|}}, e_{u_1}, \cdots, e_{u_{|N_u|}}]. \tag{8}
\]

Then, a shared CF encoder is utilized to learn the collaborative embeddings for the users and items, as shown in Figure 2. It can be implemented by various models, such as MF-based [24, 32, 36], neural network-based [15], and graph neural network-based [14, 44, 47] models. In this work, we provide two simple yet efficient implementations, CENMF and CENGCN, which are based on MF [32] and LightGCN [14], respectively.

For conciseness, we neglect their details and formulate them as,

\[
\begin{align*}
    z_i &= E(e_i, \{e_{j} | j \in N_i\}) \\
    z_u &= E(e_u, \{e_{j} | j \in N_u\}). \tag{9}
\end{align*}
\]

where, $E$ is the CF encoder taking id embeddings as input; $N_i$ is the set of users who have interacted with item $i$, while $N_u$ represents the set of items which have been purchased by user $u$. Having obtained the collaborative embeddings of user-item pair (i.e., $z_i$ and $z_u$), we define the following function to measure their correlation:

\[
g(x_i, z_u) = \exp(z_i^\top z_u / \tau), \tag{10}
\]

where $\tau$ is a temperature parameter [4]. Here we employ the inner product to approach the correlation, leaving the exploration of other non-negative functions [33] in future work.

R-E Contrastive Embedding Network. This network is used to score the correlation between two views of items — collaborative signals and content information. As shown in Figure 3, it can be regarded as two parallel pipelines to model the item’s feature representation and collaborative embedding, respectively. The right one for the collaborative embedding is the same with the corresponding operations in the U-I network.

Towards representing the content information, we utilize the feature extractor, which could be a pre-trained deep learning model, followed by the feature encoder [19]. More specifically, for item $j$ in the R-E contrastive pair, we extract its feature vector, denoted as $x_j$, from the content information. We then design the feature
encoder as a multiple layer perceptron (MLP) with one hidden layer, formally
\[ f_j = W^{(2)} \phi(W^{(1)} x_j + b^{(1)}) + b^{(2)}, \]  
(11)
where \( W^{(1)} \) and \( b^{(1)} \) denote the trainable matrix and bias vector of the encoder, respectively. Moreover, \( f_j \) is the desired feature representation of item \( j \), which aims to distill the content feature and maximally preserve the information associating to the collaborative signal. In the inference phase, powered by such feature representations, we could measure the affinity between the user and cold-start item.

After obtaining the outputs of two pipelines, i.e., the item’s collaborative embedding and feature representation, we score their density ratios with the following function:
\[ h(z_u, f_j) = \exp\left(\frac{z_u^T f_j}{||z_u|| \cdot ||f_j||} \cdot \frac{1}{\tau}\right). \]  
(12)
We empirically normalize the vectors to measure the ratio in an unit space instead of projecting them into a common space, which helps to maintain the characters of content information.

### 2.3.3 Contrastive Optimization
In order to maximize the mutual information, we perform the contrastive training strategy to optimize the parameters of the model. Incorporating the defined density ratio functions, we implement the objective function as,
\[ L = \lambda L_{RE} + (1 - \lambda) L_{UI} + \eta ||\Theta||_2^2 \]  
(13)
\[ -\lambda \mathbb{E}_{i \in U} \ln \frac{\exp\left(\frac{z_i^T f_i}{||z_i|| \cdot ||f_i||} \cdot \frac{1}{\tau}\right)}{\sum_{k=1}^K \exp\left(\frac{z_i^T f_k}{||z_i|| \cdot ||f_k||} \cdot \frac{1}{\tau}\right)} + (1 - \lambda) \mathbb{E}_{(u,i) \in O} \left(\ln \frac{\exp\left(\frac{z_u^T}{\tau}\right)}{\exp\left(\frac{z_u^T}{\tau}\right) + \sum_{k=1}^K \exp\left(\frac{z_u^T}{\tau}\right)} + \eta ||\Theta||_2^2 \right). \]

### Relation with BPR Loss
Within the function, we find that BPR loss can be regarded as a special case of \( L_{UI} \), when setting \( \tau = 1 \) and sampling one negative pair for training. However, we suggest that \( L_{UI} \) is superior to BPR loss. On one hand, it is intuitively closer to the real-world scenario that we tend to face multiple options instead of the either-or situation. On the other hand, we indicate the function intrinsically facilitates the model optimization. Specially, we calculate the gradient of \( L_{UI} \) with respect to \( z_u \) and \( f_i \):
\[ \frac{\partial L_{UI}}{\partial z_u} = \left( \sum_{(u,i) \in O} \exp(z_u^T z_u/\tau) \right) \left( \frac{\partial z_u^T}{\partial z_u} \right) z_u/\tau, \]
\[ \frac{\partial L_{UI}}{\partial f_i} = \left( \sum_{(u,i) \in O} \exp(z_i^T z_u/\tau) \right) \left( \frac{\partial z_i^T}{\partial f_i} \right) z_u/\tau. \]

### Hybrid Contrastive Training
As mentioned in Section 2.2, maximizing the R-E mutual information improves the correlation between \( z_u \) and \( f_i \), and then approximates \( \sum_{(u,i) \in O} \ln p(z_u|z_i; f_i) \) to \( \sum_{(u,i) \in O} \ln p(z_u|z_i) \). However, there is still a gap between them at the beginning of the training phase. To alleviate this drawback, we empirically utilize a hybrid training strategy in the U-I CEN, as shown in Figure 4. In particular, we randomly choose the learned feature representation or item collaborative embedding with a pre-defined probability \( \rho \) and feed it into the density ratio function. It is able to maximize the correlation between learned features and user preference as well as model the collaborative embeddings in the training phase. Taking the advantage of this design, we could predict the affinities between users and cold-start items by measuring the similarities of their representations in the inference phase.

## 3 EXPERIMENTS
In this section, we first introduce the datasets for evaluation and describe the experiment setting. Then, to evaluate the effectiveness of our proposed method, we compare our proposed method with the baselines in the warm-start, cold-start, and all item scenarios. Moreover, we analyze the designs of our proposed framework through an ablation study.

### 3.1 Experimental Settings

#### 3.1.1 Dataset
To evaluate the effectiveness of CLRec, extensive experiments are conducted on four real-world datasets, including Movielens\(^1\), Tiktok\(^2\), Kwai\(^3\), and Amazon\(^4\). Beyond the collected user-item interaction records, these datasets provide rich content information of items. For fairness, we discard the feature extractor in our framework and use the feature vectors captured by some pre-trained deep learning models, including ResNet [13], VGGish [16], and Sentence2Vector [1], which are summarised in Table 1. For each dataset, we randomly sample some items as the cold-start items and split them into the cold validation and testing sets with the

![Figure 4: Illustration of the hybrid contrastive training.](image-url)

### Table 1: Summary of the datasets. V, A, and T denote the dimensions of feature vectors in visual, acoustic, and textual modalities, respectively.

| Dataset   | Inter. # | User. # | Warm. # | Cold. # | V  | A  | T  |
|-----------|----------|---------|---------|---------|----|----|----|
| Movielens | 922,007  | 55,485  | 5,119   | 567     | -  | -  | -  |
| Tiktok    | 389,406  | 32,309  | 7,696   | 2,048   | -  | -  | -  |
| Kwai      | 236,130  | 7,010   | 100     | 236     | -  | -  | -  |
| Amazon    | 142,539  | 27,044  | 64      | 128     | -  | -  | -  |

\(^1\)https://movielens.org/.
\(^2\)https://www.tiktok.com/.
\(^3\)https://www.kwai.com/.
\(^4\)http://jmcauley.ucsd.edu/data/amazon/.
We emphasize the effectiveness of the proposed objective function by comparing two different implementations (i.e., \( \text{DUIF} \) and \( \text{DUIF} \)) to highlight the performance of our proposed model for cold-start item recommendation. Furthermore, \( \text{CLCRec} \) demonstrates the robustness of the model.

### Table 2: Comparison with Cold-Start Recommendation Models over Four Datasets.

| Metric       | Model   | Movielens | Tiktok | Kwai | Amazon |
|--------------|---------|-----------|--------|------|--------|
|              |         | Warm Cold All | Warm Cold All | Warm Cold All | Warm Cold All |
| recall@10    | \( \text{DUIF} \) | 0.2417 0.0691 0.1907 | 0.0736 0.0057 0.0567 | 0.0317 0.0068 0.0273 | 0.0047 0.0139 0.0043 |
|              | DropoutNET | 0.2437 0.0473 0.1939 | 0.0792 0.0034 0.0618 | 0.0406 0.0022 0.0349 | 0.0102 0.0010 0.0070 |
|              | MTPR     | 0.2607 0.0593 0.2059 | 0.0852 0.0045 0.0661 | 0.0464 0.0027 0.0401 | 0.0113 0.0028 0.0077 |
|              | CB2CF    | 0.2708 0.0317 0.2134 | 0.0870 0.0016 0.0678 | 0.0465 0.0023 0.0403 | 0.0167 0.0139 0.0110 |
|              | Heater   | 0.2523 0.0586 0.1993 | 0.0828 0.0032 0.0639 | 0.0464 0.0029 0.0400 | 0.0104 0.0012 0.0071 |
|              | CLCRec   | 0.3178 0.0730 0.2485 | 0.1408 0.0113 0.1033 | 0.0648 0.0098 0.0522 | 0.0238 0.0313 0.0190 |
| %Improv.     |         | 17.35% 3.64% 16.45% | 61.86% 103.95% 52.48% | 39.35% 44.71% 29.33% | 42.51% 96.24% 73.04% |

| Data         | MF-BPR | LightGCN | \( CEN_{MF} \) | \( CEN_{GCN} \) |
|--------------|--------|----------|----------------|----------------|
| Movielens    |        |          |                |                |
| Warm         | 0.2591 | 0.2136   | 0.3178         | 0.3110         |
| Cold         | 0.0190 | 0.0131   | 0.0730         | 0.0637         |
| All          | 0.2051 | 0.1942   | 0.2485         | 0.2441         |
| Tiktok       |        |          |                |                |
| Warm         | 0.0853 | 0.0941   | 0.1408         | 0.1532         |
| Cold         | 0.0004 | 0.0022   | 0.0113         | 0.0111         |
| All          | 0.0662 | 0.0729   | 0.1033         | 0.1133         |
| Kwai         |        |          |                |                |
| Warm         | 0.0474 | 0.0425   | 0.0648         | 0.0659         |
| Cold         | 0.0005 | 0.0006   | 0.0098         | 0.0107         |
| All          | 0.0406 | 0.0370   | 0.0522         | 0.0549         |
| Amazon       |        |          |                |                |
| Warm         | 0.0112 | 0.0155   | 0.0258         | 0.0251         |
| Cold         | 0.0008 | 0.0008   | 0.0313         | 0.0276         |
| All          | 0.0076 | 0.0105   | 0.0190         | 0.0196         |

- **DUIF** [9] Different from the CF-based model, this method learns the user reference to the item’s features without modeling the CF signal, which intrinsically avoids the cold-start problem.
- **DropoutNet** [43] This method performs the dropout operation that randomly discards the partial collaborative embeddings to build the cold-start condition in the training phase. It alleviates the cold-start problem by improving the robustness of the model.

### Table 3: Comparison with Cold-Start Recommendation Models over Four Datasets.

| Data         | MF-BPR | LightGCN | \( CEN_{MF} \) | \( CEN_{GCN} \) |
|--------------|--------|----------|----------------|----------------|
| Movielens    |        |          |                |                |
| Warm         | 0.2591 | 0.2136   | 0.3178         | 0.3110         |
| Cold         | 0.0190 | 0.0131   | 0.0730         | 0.0637         |
| All          | 0.2051 | 0.1942   | 0.2485         | 0.2441         |
| Tiktok       |        |          |                |                |
| Warm         | 0.0853 | 0.0941   | 0.1408         | 0.1532         |
| Cold         | 0.0004 | 0.0022   | 0.0113         | 0.0111         |
| All          | 0.0662 | 0.0729   | 0.1033         | 0.1133         |
| Kwai         |        |          |                |                |
| Warm         | 0.0474 | 0.0425   | 0.0648         | 0.0659         |
| Cold         | 0.0005 | 0.0006   | 0.0098         | 0.0107         |
| All          | 0.0406 | 0.0370   | 0.0522         | 0.0549         |
| Amazon       |        |          |                |                |
| Warm         | 0.0112 | 0.0155   | 0.0258         | 0.0251         |
| Cold         | 0.0008 | 0.0008   | 0.0313         | 0.0276         |
| All          | 0.0076 | 0.0105   | 0.0190         | 0.0196         |

### 3.3 Evaluation Metrics

We adopt the full ranking evaluation for the warm-start items, cold-start items, and all items, respectively. Whereinto, all items indicate the case that the warm- and cold-start item are combined to perform the experiments. In all cases, we adopt recall@K and Normalized Discounted Cumulative Gain (NDCG@K) as the metrics, which are widely used in personalized recommender systems. By default, we set K=10 and report the average values of the above metrics for all users during the testing phase.

### 3.4 Parameter Settings

We implement baselines and our proposed model with the help of Pytorch\(^5\) and torch-geometric package.\(^6\) Xavier [10] and Adam [22] algorithms are utilized in the experiments to initialize and optimize the parameters of the models. For fairness, we set the dimension of the collaborative embedding as 64 for all models. In terms of the hyper-parameters, we use the grid search [20, 51]: the learning rate is tuned in \([0.0001, 0.001, 0.01, 0.1]\) and regularization weight is searched in \([0.0001, 0.001, 0.01, 0.1]\). Besides, we employ the early stopping strategy [18], which stops the training if recall@10 on the validation data does not increase for

---

\(^5\)https://pytorch.org/  
\(^6\)https://pytorch-geometric.readthedocs.io/
3.2 Performance Comparison

3.2.1 Empirical Results w.r.t. Cold-start Recommendation.

Beyond the empirical results of all methods, we report the improvements in Table 2, which are calculated between our proposed method and the strongest baselines highlighted with underline. Analyzing the table from top to bottom, we observe that:

- In terms of recall@10 and NDCG@10, the performance of DUJF surpasses that of other baselines by a large margin for the items which are not observed in the training set. It is reasonable since the user in the content-based model is represented by refining the features of items she/he interacted, which associates with the unobserved item via the item features. Whereas, there is a gap between the user’s collaborative embedding and item’s feature representation in CF-based models.

- By jointly analyzing the results in Table 3, it can be found that the methods (Dropout, MTPR, CB2CF, and Heater) designed for cold-start problem outperforms MF-BPR and LightGCN in the cold-start scenario. It indicates that these methods capture the collaborative signal from items’ features, which is helpful for predicting the interactions between users and cold-start items.

- Undoubtedly, CLCRec outperforms the baselines in all cases (i.e., warm, cold, all) across the datasets and in both evaluation metrics of recall@10 and NDCG@10. In particular, for cold-start items, CLCRec achieves the improvements over the strongest baseline w.r.t. recall@10 by 5.64%, 103.95%, 44.71%, and 96.24% on the four datasets, respectively. In addition, compared with the other CF-based baselines, the improvements of our proposed method are more significant. It can be attributed to the following aspects: (1) CLCRec equipped with the U-I contrastive learning facilitates the collaborative embedding modeling, and (2) maximizing the R-E mutual information makes the feature representation preserve much more information related to the collaborative signal.

10 successive epochs. For the baselines, we do the same options and follow the designs in their articles to achieve the best performance.

3.3 Ablation Study

In this section, we perform ablation studies to obtain deep insights on our proposed framework. We start by exploring U-I CEN to justify its influence towards the CF embedding and feature representation. We then investigate R-E CEN to figure out how the R-E mutual information affects the performance. In what follows, we analyze an important hyper-parameter $\lambda$, which implements the

**Figure 5: Effect of U-I Contrastive Embedding Network.** Negative\# denotes the number of negative pairs.
trade-off between the warm- and cold-start performance. Due to the space limitation, we omit some results which have the similar trend with the ones exhibited in the following.

3.3.1 Impact of U-I CEN. To investigate whether the proposed method can benefit from U-I CEN and the hybrid contrastive training, we discard R-E CEN and varied the hybrid sampling probability $\rho$ in the range of $\{0.0, 0.2, 0.4, 0.6, 0.8, 1.0\}$ under single and multiple negative sampling cases, respectively. Figure 5 shows the results w.r.t. recall@10 on Movielens and Amazon datasets in warm- and cold-start scenarios. We observe that:

- Compared with the single negative pair sampling, U-I CEN with 128 negative pairs is capable of boosting the performance substantially in most cases. Focusing on the performance of warm-start items, we find that the improvements on Movielens are more significant than those on Amazon. This might be caused by the sparsity of interactions on Amazon, making the hard negative sampling easy, even for the single negative sampling.
- The hybrid contrastive training serves as vitamin to the cold-start recommendation. That is, even a small $\rho$, like 0.2, is beneficial to the feature representation learning, which verifies the effectiveness of this training strategy.
- From the results, we observe that the performance on Movielens dataset slightly decreases when sampling multiple negative pairs and increasing the sampling probability $\rho$. It is probably caused by the densely interacted items on Movielens, which makes the information unrelated to collaborative signal accumulate. With the increasing negative samples and $\rho$, much more noisy information is encoded into the collaborative embedding.

3.3.2 Impact of R-E CEN. To demonstrate the effectiveness of R-E CEN, we do ablation study by adjusting the number of R-E negative pairs. In particular, we sample one negative user-item pair for U-I CEN to emphasize the influence of R-E CEN for the performance of cold-start items. The results w.r.t. recall@10 and NDCG@10 on Movielens and Amazon datasets are plotted in Figure 6, from which we derive two key observations:

- Jointly analyzing the results in Table 2, our proposed method shows significant improvements over all competing methods even when we only sample several negative pairs for contrastive training. Moreover, increasing the negative pair number from 32 to 256 significantly enhances the performance. Taking Movielens dataset as an example, setting the number as 256 leads to the highest recall@10 of 0.0810. It illustrates the substantial influence of R-E CEN on item recommendation.
- However, the performance drops when the number of negative pairs goes beyond 256, especially for the Amazon dataset. One possible reason is that the overload negative sampling causes the overfitting of R-E mutual information, which corrupts the expressiveness of the collaborative embedding. This suggests that, although our proposed method benefits from the multiple negative pairs, it also could be suffered from the overuse.

3.3.3 Trade-off between Warm- and Cold-Start. We vary the quantity of the trade-off hyper-parameter ($i.e., \lambda$) to investigate the effectiveness of U-I and R-E CENs towards the warm- and cold-start item recommendation. Observing the performance shown in Figure 7, we have the following findings:

- It is not unexpected that the performance w.r.t. recall@10 consistently improves in the cold-start case, when increasing the value of $\lambda$ in range of $\{0.0, 0.1, \ldots, 0.8\}$. Despite the visually smooth curve on Movielens dataset, the absolute value of recall@10 increases from 0.0463 at the beginning to the highest value 0.0730. It demonstrates that $L_{RE}$ is beneficial to capture the collaborative information from the content for the cold-start recommendation.
- For the warm-start items, the performance is slightly affected by varying $\lambda$ in most cases. It again verifies that the improvements of the cold-start recommendation are mainly obtained by maximizing the mutual information between items’ collaborative embeddings and feature representations. The contribution of U-I mutual information is to enrich the collaborative embedding.
- Although the content information could be supplementary for collaborative signal, the performance of the warm-start items decreases with the increase of contribution of R-E mutual information, especially for the Amazon dataset. The main reason is that more information unrelated to the collaborative signal is injected into the collaborative embedding by the back-propagation during the training. From the results for the all-item scenario, it demonstrates that an appropriate value for $\lambda$ is essential to balance these two kinds of mutual information for pursuing the best performance on all of scenarios.

3.4 In-depth Analysis
To further explore how our proposed method optimizes the recommendation, we conduct the additional experiments to
investigate the training process w.r.t. gradient magnitude. Following the prior work [45], we test the proposed method in different numbers of contrastive negative pairs and compute the gradient magnitude in Eq. 14 as the evaluation metric. We record the status of gradient magnitude at each epoch and exhibit the learning curves associated with the performance w.r.t. recall@10 on Movielens and Amazon datasets in Figure 8.

Through analyzing the average gradient magnitude and the performance, we find that our proposed method achieves the larger gradient magnitudes as compared to MF-BPR; moreover, the performance is boosting with increasing the number of negative pairs. Furthermore, the contrastive learning not only improves the recommendation performance but promotes the convergence on Movielens dataset. We attribute its efficiency to the intrinsic hard negative detection. This demonstrates that the optimization is benefited from the contrastive training strategy, which prevents the gradient vanishing problem during the training process.

4 RELATED WORK

4.1 Cold-start Recommendation

Towards solving the cold-start problem, it is crucial to represent the item to alleviate the difference between the warm- and cold-start items. Thus, the side information, especially the content features, tends to be incorporated into the CF-based recommendation models [6, 7], as the bridge to model the collaborative signal for cold-start items. For instance, DropoutNET [43], MTPR [8], and CC-CC [38] randomly forgo some collaborative embeddings to improve the robustness of CF-based model, which implicitly discovers the information related to collaborative signal from the items’ content features. Different from these robustness-based models, some efforts [3, 26, 39, 54] are dedicated to explicitly modeling the correlation between the content information and collaborative embeddings. By devising the various constraint functions, such as local geometric similarity and mean square error, they enforce the representation learned from content information to approximate the collaborative embeddings. By contrast, we explicitly model the correlation between the content information and collaborative filtering signal by maximizing the mutual information instead of imposing strict constraint loss.

4.2 Contrastive Learning

The contrastive learning, as a representative algorithm of the self-supervised learning, is widely used in the field of computer vision [4, 11, 12, 21] and natural language processing [5, 33]. By identifying the positive pair from some negative ones, it maximizes the mutual information between different representations, which discovers the semantic information shared by different views. Towards this end, Oord et al. [33] proposed a probabilistic contrastive loss, termed InfoNCE, in a way that maximally preserves the mutual information between the observation and context signals. Proposed independently of CPC, Hjelm et al. [17] compared the global and local features of each image and maximized their mutual information to discover the invariant semantic signal. With their success on representation learning, several approaches are proposed for various applications. For the recommender system, several models [53] employ the contrastive learning to optimize the representations of users or items. However, the methods focus on the representation from either the collaborative or content space. In contrast, we treat the collaborative embedding and feature representation as two different views and maximize their mutual information, in order to capture the collaborative signal from the content information.

5 CONCLUSION AND FUTURE WORK

In this work, we focus on the cold-start problem, especially for the complete cold-start items. By exploring the disadvantages of prior works, we propose to reformulate the cold-start item representation learning and prove it to be lower-bounded by the integration of U-I and R-E mutual information. And, we develop a novel objective function founded on the contrastive learning to maximize these two kinds of mutual information. Accordingly, we further develop a general cold-start recommendation framework, named Contrastive Learning-based Cold-start Recommendation, consisting of the contrastive pair organization, contrastive embedding network, and contrastive optimization. We conduct extensive experiments over four datasets, and verify the effectiveness and efficiency of our proposed method, which outperforms the state-of-the-art baselines by a large margin both in the warm- and cold-start scenarios.

To the best of our knowledge, this is the first attempt to solve the cold-start problem by maximizing the mutual information between feature representations and collaborative embeddings of items. In future, we plan to explore the related tasks in mutual information in recommendation, such as the explainable recommender system through disentangling the mixture information of users’ and items’ representations. Besides, we expect to further investigate the effectiveness of mutual information maximization in various cross-modal applications, like the video localization [29] and fashion matching [49, 50, 52].

REFERENCES

[1] Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2016. A simple but tough-to-beat baseline for sentence embeddings. In Proceedings of International Conference on Learning Representations. 1–16.
[2] Iman Barjastej, Rana Forsati, Dennis Ross, Abdol-Hossein Esfahanian, and Hayder Radha. 2016. Cold-start recommendation with provable Guarantees: A decoupled approach. IEEE Transactions on Knowledge and Data Engineering 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 Proceedings of ACM Conference on Recommender Systems. 228–236.
[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. Proceedings of International Conference on Machine Learning.
[5] Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2020. InfoXLM: An information-theoretic framework for cross-lingual language model pre-training. Proceedings of AAAI Conference on Artificial Intelligence.
[6] Yashar Deldooj, Maurizio Ferrari Dacremet, Mihai Gabriel Constantin, Hamid Eghbal-Zadeh, Stefano Cereda, Markus Schefli, Bogdan Ionescu, and Paolo Cremonesi. 2019. Movie genome: alleviating new item cold start in movie recommendation. User Modeling and User-Adapted Interaction 29, 2 (2019), 291–343.
[7] Yashar Deldooj, Markus Schefli, Paolo Cremonesi, and Gabriella Pasi. 2020. Recommender systems leveraging multimedia content. ACM Computing Surveys (CSUR) 53, 5 (2020), 1–38.
[8] Xiaoyu Du, Xiang Wang, Xiangnan He, Zechao Li, Jinlui Tang, and Tat-Seng Chua. 2020. How to Learn Item Representation for Cold-Start Multimedia Recommendation?. In Proceedings of ACM International Conference on Multimedia. 3469–3477.
[9] Xue Geng, Hanwang Zhang, Jingwen Bian, and Tat-Seng Chua. 2015. Learning image and user features for recommendation in social networks. In Proceedings of the IEEE International Conference on Computer Vision. 4274–4282.

[10] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In Proceedings of the International Conference on Artificial Intelligence and Statistics. 249–256.

[11] Jean-Bastien Grill, Florian Strub, Florent Altchén, Corentin Tallec, Pierre H Richemond, Elena Buschukatya, Carl Doersch, Bernardo Avila Pires, Zhaozhong Daniel Gao, Mohammad Ghasemlou Azar, et al. 2020. Bootstrap your own latent: A new approach to self-supervised learning. Advances in Neural Information Processing Systems.

[12] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. 770–778.

[13] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. Proceedings of International ACM SIGIR Conference on Research and Development in Information Retrieval. 639–648.

[14] Xiangnan He, Lizi Xie, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In Proceedings of International Conference on World Wide Web. 173–182.

[15] Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Arie Jiang, R Chang-Beeg Moore, Manoj Plakal, Devin Platt, Rj A Saurous, Bryan Seymour, et al. 2017. CNN architectures for large-scale audio classification. In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing. 131–135.

[16] B. Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karon Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. 2019. Learning deep representations by mutual information estimation and maximization. Proceedings of International Conference on Learning Representations.

[17] Wei Ji, Xi Li, Lina Wei, Fei Wu, and Yueting Zhuang. 2019. Human-centric clothing segmentation via deformable semantic locality-preserving network. IEEE Transactions on Circuits and Systems for Video Technology. 30 (12), 2019, 4837–4848.

[18] Hao Jiang, Wenjie Wang, Yinwei Wei, Zao Gao, Yinglong Wang, and Liqiang Nie. 2020. What Aspect Do You Like: Multi-scale Time-aware User Interest Modeling for Micro-video Recommendation. In Proceedings of the 28th ACM International Conference on Multimedia. 3487–3495.

[19] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Arun Misra, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. Advances in Neural Information Processing Systems.

[20] Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In Proceedings of International Conference on Learning Representations. 1–16.

[21] Yehuda Koren. 2008. Factorization meets the neighborhood: a multifaceted technique for recommender systems. ACM, 2008, 30–37.

[22] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. Computer, Vol. 42. 8 (2009), 30–37.

[23] Hoyeop Lee, Jinbae Im, Seongwon Jang, Hyunsouk Cho, and Sehee Chung. 2019. Contrastive Multiview Coding. arXiv preprint arXiv:1906.05849 (2019).

[24] Xiang Wang, Xiang, Xi Li, Fei Wu, and Yueting Zhuang. 2021. Hierarchical User Intent Graph Network for Multimedia Recommendation. IEEE Transactions on Multimedia.

[25] Xiang Wang, Xiangnan He, Meng Wang, FuI Feng, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In Proceedings of the International ACM SIGIR conference on Research and Development in Information Retrieval. 165–174.

[26] Xiang Wang, Xiaokun Xu, Xiangnan He, Yixin Cao, Meng Wang, and Tat-Seng Chua. 2020. Reinforced negative sampling over knowledge graph for recommendation. In Proceedings of The Web Conference 2020. 99–109.

[27] Xuyi Wei, Yuanfang Li, Liang Nie, Zhouchen Lin, and Baoquan Chen. 2019. Learning image and user features for recommendation in social networks. In Proceedings of the 28th ACM International Conference on Information Discovery & Data Mining. 1536–1537.

[28] Andrey Mnih and Rus R Salakhutdinov. 2008. Probabilistic matrix factorization. In Advances in neural information processing systems. 1257–1264.

[29] Andriy Mnih, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Arun Misra, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. 695–704.

[30] Zhaohan Daniel Guo, Mohammad Ghasemlou Azar, et al. 2020. Bootstrap your own latent: A new approach to self-supervised learning. arXiv preprint cs.IR/2005.12964 (2020).

[31] Yaohui Fan, 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 Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. 695–704.

[32] Xiang Wang, Xiangnan He, Meng Wang, FuI Feng, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In Proceedings of the International ACM SIGIR conference on Research and Development in Information Retrieval. 165–174.

[33] Xiang Wang, Xiangnan He, Meng Wang, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In Proceedings of the International ACM SIGIR conference on Research and Development in Information Retrieval. 165–174.

[34] Xiyu Wei, Yuanfang Li, Liang Nie, Zhouchen Lin, and Baoquan Chen. 2019. Learning image and user features for recommendation in social networks. In Proceedings of the 28th ACM International Conference on Information Discovery & Data Mining. 1536–1537.

[35] Yuanfu Lu, Yuanfang Li, and Liqiang Nie. 2020. Smart Contract Vulnerability Detection using Graph Neural Network. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. 695–704.

[36] Xun Yang, Xiangnan He, Xiang Wang, Yunshan Ma, Fuli Feng, Meng Wang, and Tat-Seng Chua. 2019. MeLU: meta-learned user preference estimator for cold-start recommendation. Advances in Neural Information Processing Systems.

[37] Xun Yang, Xiangnan He, Meng Wang, and Tat-Seng Chua. 2020. Reinforced negative sampling over knowledge graph for recommendation. In Proceedings of The Web Conference 2020. 99–109.

[38] Xuyi Wei, Yuanfang Li, Liang Nie, Zhouchen Lin, and Baoquan Chen. 2019. Learning image and user features for recommendation in social networks. In Proceedings of the 28th ACM International Conference on Information Discovery & Data Mining. 1536–1537.

[39] Zhaohan Daniel Guo, Mohammad Ghasemlou Azar, et al. 2020. Bootstrap your own latent: A new approach to self-supervised learning. arXiv preprint cs.IR/2005.12964 (2020).

[40] Personalized Item Recommendation for New Users and New Items via Randomized Training and Reinforcement Learning. In Proceedings of ACM International Conference on Information and Knowledge Management. 1451–1460.

[41] Bojan Kostic, Jovan Stefanovic, and فلاورا بات. 2019. Combining content and collaboration in text filtering. In Proceedings of International Joint Conferences on Artificial Intelligence. Vol. 99. 86–93.

[42] Martin Saveski and Amin Mantrach. 2014. Item cold-start recommendations: learning local collective embeddings. In Proceedings of ACM Conference on Recommender systems. 89–96.

[43] Zhe Xue, Hanwang Zhang, Jingwen Bian, and Tat-Seng Chua. 2015. Learning image and user features for recommendation in social networks. In Proceedings of the IEEE International Conference on Computer Vision. 2474–2482.

[44] Xiang Wang, Xiangnan He, Meng Wang, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In Proceedings of the International ACM SIGIR conference on Research and Development in Information Retrieval. 165–174.

[45] Xiang Wang, Xiangnan He, Meng Wang, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In Proceedings of the International ACM SIGIR conference on Research and Development in Information Retrieval. 165–174.

[46] Xiang Wang, Xiangnan He, Meng Wang, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In Proceedings of the International ACM SIGIR conference on Research and Development in Information Retrieval. 165–174.