Contrastive Learning for Cold-Start Recommendation

Yinwei Wei
National University of Singapore
weiyinwei@hotmail.com

Xiang Wang[§]
Sea-NExT Joint Lab, Singapore
National University of Singapore
xiangwang@u.nus.edu

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

Liqiang Nie[§]
Shandong University
nieliqiang@gmail.com

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

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

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

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 efficient 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.

CCS CONCEPTS

• **Information systems** → **Recommender systems**.

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 [12, 13, 29, 40]. We denote the representations encoding collaborative signals as collaborative embeddings, such as ID embeddings in MF [29], interaction history embeddings in SVD++ [19], and interaction graph embeddings in LightGCN [12]. 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

[§]Xiang Wang and Liqiang Nie are the corresponding authors.

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 [7, 31, 36] and constraint-based [3, 22, 30, 47]. 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 items whose interactions are missing. Technically, they augment the training data by randomly corrupting warm items' collaborative embeddings [7, 36] or manually crafting new items [21, 35], so as to enhance the generalization ability to unseen items. However, no function is designed to explicitly consider the relationships between item content and collaborative embeddings, thereby hardly refining useful collaborative signals for cold-start items.
- Constraint-based methods [2, 3, 47] explicitly model the relationships between item content and collaborative embeddings by applying a constraint loss. Specifically, for an item, its content is transferred as feature representation; thereafter, a constraint loss (*e.g.,* L_2 regularizer) enforces dimension-wise similarity between feature representation and collaborative embedding. Nonetheless, there are two limitations: (1) such dimension-wise regularization merely remains the consistency between these two representations, which possibly discards the heterogeneous and unique information; (2) it lacks sufficient theoretical support to ensure what and how much information on collaborative embeddings is preserved in feature representations.

In a nutshell, while being powered by the advance of neural network techniques (*e.g.,* dropout [33] and constraint [47]), 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, 26, 34] 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 $O = \{o_{u,i}\}$ be the set of user-item interactions, which involves the set of N users, $\mathcal{U} = \{u\}$ and the set of M items, $I = \{i\}$. With the assumption that users with similar behavior prefer similar items, modern collaborative filtering (CF) methods [16, 23, 24] 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 $\mathbf{z}_u \in \mathbb{R}^d$ ($\mathbf{z}_i \in \mathbb{R}^d$), where d is the embedding size. Optimizing collaborative embeddings is to maximize the posterior estimator given the observed interactions:

$$\mathbf{Z}^* = \arg\max_{\mathbf{Z}} p(\mathbf{Z}|O), \qquad (1)$$

where $\mathbf{Z} = [\mathbf{z}_{i_1}, \cdots, \mathbf{z}_{i_N}, \mathbf{z}_{u_1}, \cdots, \mathbf{z}_{u_M}]$ collects the collaborative embeddings of all users and items; $p(\mathbf{Z}|O)$ is to encode as much information on O into $\mathbf{Z}$ 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_θ is built upon the content feature vector $\mathbf{x}_i$ to generate the feature representation $\mathbf{f}_i$ of item i. Here we establish the collections of feature representations and content features as $\mathbf{F} = [\mathbf{f}_{i_1}, \cdots, \mathbf{f}_{i_N}]$ and $\mathbf{X} = [\mathbf{x}_{i_1}, \cdots, \mathbf{x}_{i_N}]$, respectively. Note that, $\mathbf{F}$ that are learnable, thus differs from $\mathbf{X}$ that are learned by pre-trained feature extractors [25, 38]. Previous works either treat the feature representation and collaborative embeddings individually [7, 31, 36] or apply a regularizer term to encourage their dimension-wise similarity [2, 22, 30]. 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:

$$\theta^* = \arg\max_\theta p(\theta|\mathbf{Z}, O, \mathbf{X}). \tag{2}$$

Having obtained the optimized feature encoder θ^*, 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 $\mathbf{X}$ are usually independent to the collaborative embeddings $\mathbf{Z}$, we can adopt Bayesian rules and rewrite Equation (2) as follows:

$$p(\theta|\mathbf{Z}, O, \mathbf{X}) \propto p(\mathbf{Z}, O|\mathbf{F}) \cdot p(\theta). \tag{3}$$

Following the prior study [29], 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:

$$p(\mathbf{Z}, O|\mathbf{F}) \cdot p(\theta) \propto \sum_{i \in \mathcal{I}} \ln p(\mathbf{z}_i, \mathbf{z}_{u_1}, \mathbf{z}_{u_2}, \dots, \mathbf{z}_{u_N}|\mathbf{f}_i) + \ln p(\theta)$$

$$= \sum_{i \in \mathcal{I}} \ln p(\mathbf{z}_i, \mathbf{f}_i) + \sum_{(u,i) \in O} \ln p(\mathbf{z}_u|\mathbf{z}_i, \mathbf{f}_i) + C \sum_{u \in \mathcal{U}} \ln p(\mathbf{z}_u) + \ln p(\theta).$$

Here C is a constant term derived from the sum of conditional probabilities: $\sum_{(u,i) \notin O} \ln p(\mathbf{z}_u|\mathbf{z}_i, \mathbf{f}_i)$. 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 $\mathbf{z}_i$ and $\mathbf{f}_i$, such that $\mathbf{z}_i$ is encouraged to embed the information of $\mathbf{f}_i$. Thus, we could relieve the second term $\sum_{(u,i) \in O} \ln p(\mathbf{z}_u|\mathbf{z}_i, \mathbf{f}_i)$ from the complex relationship among three kinds of representations. Specifically, it is proportional to the posterior probabilities of user embedding, after observing the item embedding:

$$\sum_{i \in \mathcal{I}} \ln p(\mathbf{z}_i, \mathbf{f}_i) + \sum_{(u,i) \in O} \ln p(\mathbf{z}_u|\mathbf{z}_i, \mathbf{f}_i) + C \sum_{u \in \mathcal{U}} \ln p(\mathbf{z}_u) + \ln p(\theta)$$

$$\propto \sum_{i \in \mathcal{I}} \ln p(\mathbf{z}_i, \mathbf{f}_i) + \sum_{(u,i) \in O} \ln p(\mathbf{z}_u|\mathbf{z}_i) + C \sum_{u \in \mathcal{U}} \ln p(\mathbf{z}_u) + \ln p(\theta).$$

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:

$$\sum_{i \in \mathcal{I}} \ln p(\mathbf{z}_i, \mathbf{f}_i) + \sum_{(u,i) \in O} \ln p(\mathbf{z}_u|\mathbf{z}_i) + C \sum_{u \in \mathcal{U}} \ln p(\mathbf{z}_u) + \ln p(\theta)$$

$$\geq \underbrace{\sum_{i \in \mathcal{I}} p(\mathbf{f}_i, \mathbf{z}_i) \ln \frac{p(\mathbf{z}_i, \mathbf{f}_i)}{p(\mathbf{z}_i)p(\mathbf{f}_i)}}_{MI(\mathbf{F}, \mathbf{Z}_I)} + \underbrace{\sum_{(i,j) \in O} p(\mathbf{z}_u, \mathbf{z}_i) \ln \frac{p(\mathbf{z}_u|\mathbf{z}_i)}{p(\mathbf{z}_u)}}_{MI(\mathbf{Z}_I, \mathbf{Z}_U)}$$

$$+ \underbrace{\sum_{i \in \mathcal{I}} \ln p(\mathbf{f}_i) + \sum_{i \in \mathcal{I}} \ln p(\mathbf{z}_i) + (C+1) \sum_{u \in \mathcal{U}} \ln p(\mathbf{z}_u) + \ln p(\theta)}_{\ln p(\Theta)}$$

where $MI(\cdot, \cdot)$ represents the mutual information between two representations. In addition, we use Θ 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 [26], we define two density ratio functions:

$$h(\mathbf{z}_i, \mathbf{f}_i) \propto \frac{p(\mathbf{f}_i|\mathbf{z}_i)}{p(\mathbf{f}_i)}, \quad g(\mathbf{z}_u, \mathbf{z}_i) \propto \frac{p(\mathbf{z}_u|\mathbf{z}_i)}{p(\mathbf{z}_u)}. \tag{4}$$

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:

$$\mathcal{L} = \lambda \mathcal{L}_{RE} + (1 - \lambda)\mathcal{L}_{UI} + \eta||\Theta||_2^2 \tag{5}$$

$$= -\lambda \mathop{\mathbb{E}}_{i \in \mathbf{I}} \left[\ln \frac{h(\mathbf{z}_i, \mathbf{f}_i)}{h(\mathbf{z}_i, \mathbf{f}_i) + \sum_{j \in \mathbf{I}/\{i\}} h(\mathbf{z}_i, \mathbf{f}_j)} \right]$$

$$- (1 - \lambda) \mathop{\mathbb{E}}_{(u,i) \in O} \left[\ln \frac{g(\mathbf{z}_u, \mathbf{z}_i)}{g(\mathbf{z}_u, \mathbf{z}_i) + \sum_{(u,j) \notin O} g(\mathbf{z}_u, \mathbf{z}_j)} \right] + \eta||\Theta||_2^2.$$

We use $\mathcal{L}_{RE}$ and $\mathcal{L}_{UI}$ to denote the contrastive loss for maximizing the R-E and U-I mutual information, respectively. As $\mathcal{L}_{UI}$ has the same optimal objective with Personalized Ranking (BPR) loss [29], we could optimize the recommendation model in an end-to-end manner without any other loss function, and set a hyper-parameter λ 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(a). 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.
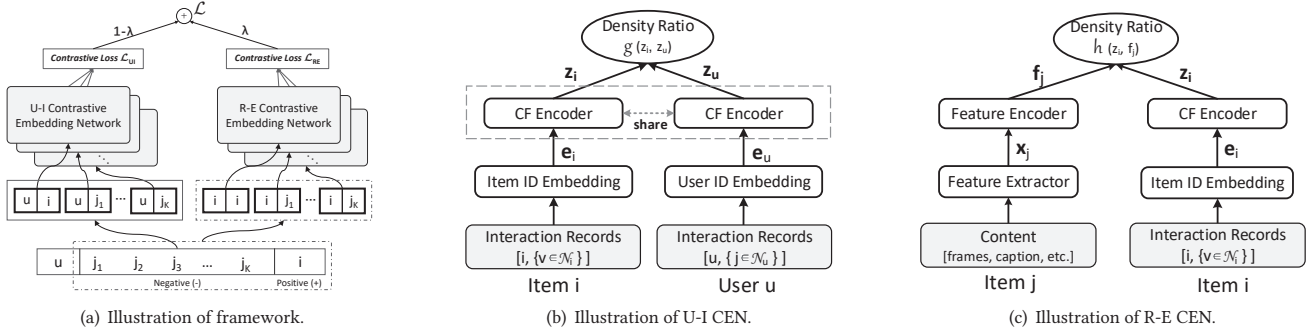
(a) Illustration of framework.   (b) Illustration of U-I CEN.   (c) Illustration of R-E CEN.

Figure 1: Schematic illustration of our proposed framework and contrastive embedding networks.

*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(a). 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(a). 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(a) 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(a), 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.,* $\mathbf{e}_u$ and $\mathbf{e}_i$) from an look-up table, which is defined by a parameter matrix:

$$\mathbf{E} = [\mathbf{e}_{i_1}, \ldots, \mathbf{e}_{i_N}, \mathbf{e}_{u_1}, \ldots, \mathbf{e}_{u_M}]. \tag{8}$$

Then, a shared CF encoder is utilized to learn the collaborative embeddings for the users and items, as shown in Figure 1(b). It can be implemented by various models, such as MF-based [20, 29], neural network-based [13], and graph neural network-based [12, 37, 39] models. In this work, we provide two simple yet efficient

implementations, CEN_{MF} and CEN_{GCN}, which are based on MF [20] and LightGCN [12], respectively. For conciseness, we neglect their details and formulate them as,

$$\begin{cases} \mathbf{z}_i = \mathcal{E}(\mathbf{e}_i, \{\mathbf{e}_v | v \in \mathcal{N}_i\}) \\ \mathbf{z}_u = \mathcal{E}(\mathbf{e}_u, \{\mathbf{e}_j | j \in \mathcal{N}_u\}), \end{cases} \tag{9}$$

where, $\mathcal{E}$ is the CF encoder taking id embeddings as input; $\mathcal{N}_i$ is the set of users who have interacted with item i, while $\mathcal{N}_u$ represents the set of items which have been purchased by user u. Having obtained the collaborative embeddings of user-item pair (*i.e.,* $\mathbf{z}_i$ and $\mathbf{z}_u$), we define the following function to measure their correlation:

$$g(\mathbf{z}_i, \mathbf{z}_u) = \exp(\mathbf{z}_i^\mathsf{T} \mathbf{z}_u / \tau), \tag{10}$$

where τ is a temperature parameter [4]. Here we employ the inner product to approach the correlation, leaving the exploration of other non-negative functions [26] 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 1(c), 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. More specifically, for item j in the R-E contrastive pair, we extract its feature vector, denoted as $\mathbf{x}_j$, from the content information. We then design the feature encoder as a multiple layer perceptron (MLP) with one hidden layer, formally

$$\mathbf{f}_j = \mathbf{W}^{(2)} \phi(\mathbf{W}^{(1)} \mathbf{x}_j + \mathbf{b}^{(1)}) + \mathbf{b}^{(2)}, \tag{11}$$

where $\mathbf{W}^{(\cdot)}$ and $\mathbf{b}^{(\cdot)}$ denote the trainable matrix and bias vector of the encoder, respectively. Moreover, $\mathbf{f}_j$ is the desired feature representation of item j, which aims to distill the content feature and maximumly 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:

$$h(\mathbf{z}_i, \mathbf{f}_j) = \exp\left(\frac{\mathbf{z}_i^\mathsf{T} \mathbf{f}_j}{||\mathbf{z}_i|| \cdot ||\mathbf{f}_j||} \cdot \frac{1}{\tau}\right). \tag{12}$$
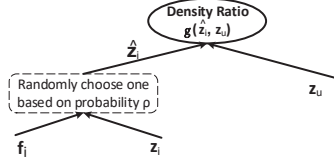
Figure 2: Illustration of the hybrid contrastive training.

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 **Contrasitve 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,

$$\mathcal{L} = \lambda \mathcal{L}_{RE} + (1-\lambda)\mathcal{L}_{UI} + \eta||\Theta||_2^2 \tag{13}$$

$$= -\lambda \mathbb{E}_{i \in \mathbb{I}}\left[\ln \frac{\exp(\frac{\mathbf{z}_i^\mathsf{T} \mathbf{f}_i}{||\mathbf{z}_i||\cdot||\mathbf{f}_i||} \cdot \frac{1}{\tau})}{\exp(\frac{\mathbf{z}_i^\mathsf{T} \mathbf{f}_i}{||\mathbf{z}_i||\cdot||\mathbf{f}_i||} \cdot \frac{1}{\tau}) + \sum_{k=1}^K \exp(\frac{\mathbf{z}_i^\mathsf{T} \mathbf{f}_{j_k}}{||\mathbf{z}_i||\cdot||\mathbf{f}_{j_k}||} \cdot \frac{1}{\tau})} \right]$$

$$- (1-\lambda)\mathbb{E}_{(u,i)\in O}\left[\ln \frac{\exp(\frac{\mathbf{z}_i^\mathsf{T} \mathbf{z}_u}{\tau})}{\exp(\frac{\mathbf{z}_i^\mathsf{T} \mathbf{z}_u}{\tau}) + \sum_{k=1}^K \exp(\frac{\mathbf{z}_{j_k}^\mathsf{T} \mathbf{z}_u}{\tau})} \right] + \eta||\Theta||_2^2.$$

Relation with BPR Loss. Within the function, we find that BPR loss can be regarded as a special case of $\mathcal{L}_{UI}$, when setting τ as 1 and sampling one negative pair for training. However, we suggest that $\mathcal{L}_{UI}$ 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 that the function intrinsically facilitates the model optimization. Specially, we calculate the gradient of $\mathcal{L}_{UI}$ with respect to $\mathbf{z}_i$:

$$\frac{\partial \mathcal{L}_{UI}}{\partial \mathbf{z}_i} = \left(\frac{\sum_{k=1}^K \exp(\mathbf{z}_{j_k}^\mathsf{T} \mathbf{z}_u/\tau)}{\exp(\mathbf{z}_i^\mathsf{T} \mathbf{z}_u/\tau) + \sum_{k=1}^K \exp(\mathbf{z}_{j_k}^\mathsf{T} \mathbf{z}_u/\tau)} \right)\left(-\frac{\mathbf{z}_i}{\tau} \right)\frac{\partial \mathbf{z}_u^\mathsf{T}}{\partial \mathbf{z}_i}. \tag{14}$$

During the training process, the negative items are increasingly pushed away from the users who have no interactions. Thus, the scores of the negative pairs are close to zero and the contribution of the above gradients comes mainly from hard negative items. For BPR loss, it is negatively influenced by the randomly negative sampling, which gains the hard negative in a low probability. $\mathcal{L}_{UI}$ is benefited by its multiple negative samples, which obviously increases the probability of hard negative detection, thus offers meaningful and larger gradients to the optimization [28].

Hybrid Contrastive Training. As mentioned in Section 2.2, maximizing the R-E mutual information improves the correlation between $\mathbf{z}_i$ and $\mathbf{f}_i$, and then approximates $\sum_{(u,i)\in O} \ln p(\mathbf{z}_u|\mathbf{z}_i, \mathbf{f}_i)$ to $\sum_{(u,i)\in O} \ln p(\mathbf{z}_u|\mathbf{z}_i)$. 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 2. In particular, we randomly choose the learned feature representation or item collaborative embedding with a pre-defined probability ρ 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

Table 1: Summary of the datasets.

Dataset	Inter. #	User #	Warm #	Cold#	V	A	T
Movielens	922,007	55,485	5,119	867	2,048	128	100
Tiktok	389,406	32,309	57,832	8,624	256	128	128
Kwai	236,130	7,010	74,470	12,013	2,048	-	-
Amazon	142,539	27,044	68,810	17,696	64	-	-

the affinities between users and cold-start items by measuring the similarities of their representations in the inference phase.

3 EXPERIMENTS
3.1 Experimental Settings

*3.1.1 **Dataset**.* We conduct extensive experiments on four real-world datasets, including Movielens, Tiktok, Kwai, and Amazon. Beyond the collected user-item interaction records, these datasets provide rich content information of items. For fairness, we discard the feature extractor and extract the features via some pre-trained deep learning models, including ResNet [11], VGGish [14], and Sentance2Vector [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 ratio 1 : 1. The other items are grouped into the training set, warm-validation set, and warm-testing set with the ratio 8 : 1 : 1. And, we build the all validation (testing) set by combing the warm- with cold- validation (testing) sets. The validation and testing sets are used to tune the hyper-parameters and evaluate the performance in the experiments, respectively.

*3.1.2 **Baselines**.* We compare CLCRec with several state-of-the-art models designed for the cold-start recommender system. In addition to the robustness-based (*i.e.* DropoutNET, MTPR) and constraint-based methods (*i.e.* CB2CF, Heater), we introduce a content-based model (*i.e.,* DUIF) to highlight the performance of our proposed model for cold-start item recommendation. Furthermore, we emphasize the effectiveness of the proposed objective function by comparing two different implementations (*i.e.,* CEN_{MF} and CEN_{GCN}) with the BPR loss-based model: MF-BPR and LightGCN.

- **DUIF** [8] 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** [36] 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.
- **MTPR** [7] Inspired by the counterfactual thinking, this method defines counterfactual representation to replace the collaborative embedding with the all-zero vector.
- **CB2CF** [3] It performs a deep neural multiview model to represent the rich content information. And, a mean squared error (MSE) serves as a bridge from items' feature representations into their collaborative embeddings.
- **Heater** [47] This method uses the sum squared error (SSE) loss to model the collaborative embedding from the content information. Meanwhile, it harnesses a randomized training method to promote the effectiveness.
- **MF-BPR** [29] The method learns latent vectors (*i.e.* collaborative embedding) to represent the user and item according to past

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	DUIF	0.2417	0.0691	0.1907	0.0736	0.0057	0.0567	0.0317	0.0068	0.0273	0.0047	0.0159	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%	5.64%	16.45%	61.86%	103.95%	52.48%	39.35%	44.71%	29.53%	42.51%	96.24%	73.04%
NDCG@10	DUIF	0.1577	0.0432	0.1308	0.0426	0.0028	0.0338	0.0273	0.0036	0.0246	0.0021	0.0082	0.0020
	DropoutNET	0.1427	0.0217	0.1191	0.0459	0.0020	0.0368	0.0303	0.0012	0.0272	0.0049	0.0004	0.0034
	MTPR	0.1720	0.0317	0.1427	0.0473	0.0025	0.0378	0.0406	0.0017	0.0365	0.0061	0.0014	0.0041
	CB2CF	0.1785	0.0168	0.1474	0.0503	0.0006	0.0404	0.0401	0.0014	0.0358	0.0088	0.0065	0.0058
	Heater	0.1627	0.0368	0.1349	0.0466	0.0020	0.0370	0.0397	0.0015	0.0357	0.0055	0.0006	0.0038
	CLCRec	**0.2392**	**0.0444**	**0.1969**	**0.0853**	**0.0062**	**0.0650**	**0.0528**	**0.0059**	**0.0451**	**0.0126**	**0.0175**	**0.0103**
	%Improv.	34.01%	2.78%	33.85%	69.54%	119.01%	61.07%	30.05%	64.61%	23.56%	42.76%	111.89%	77.66%

interactions, and predict their affinities by measuring the similarities between the learned representations.

- **LightGCN** [12] Based on graph convolutional networks, it learns the high-order CF signal and injects it into the collaborative embedding to optimize the recommendation.

3.1.3 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.

3.1.4 Parameter Settings. For all models, we implement them with the help of Pytorch[1] and torch-geometric package[2]. Xavier [9] and Adam [18] algorithms are utilized in the experiments to initialize and optimize the parameters. In terms of the hyper-parameters, we use the grid search [16, 44]: 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 set the embedding dimension as 64 and stop the training if recall@10 on the validation data does not increase for 10 successive epochs.

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 DUIF 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.

---
[1]https://pytorch.org/.
[2]https://pytorch-geometric.readthedocs.io/.

Table 3: Comparison with BPR loss-based 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	0.0238	**0.0251**
	Cold	0.0008	0.0008	**0.0313**	0.0276
	All	0.0076	0.0105	0.0190	**0.0196**

- 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.
- Analyzing the results across different datasets, we find that the improvements of our proposed method on Tiktok and Amazon are much more significant than that of Movienlens and Kwai datasets. It signifies that CLCRec can distill the informative features from the content information, while the baselines are limited by the sparse interactions on Tiktok and Amazon datasets.

3.2.2 Performance Comparison w.r.t. Objective Functions. In this work, we propose an objective function founded on the

(a) Warm-start Item on Movielens

(b) Cold-start Item on Movielens

(c) Warm-start Item on Amazon
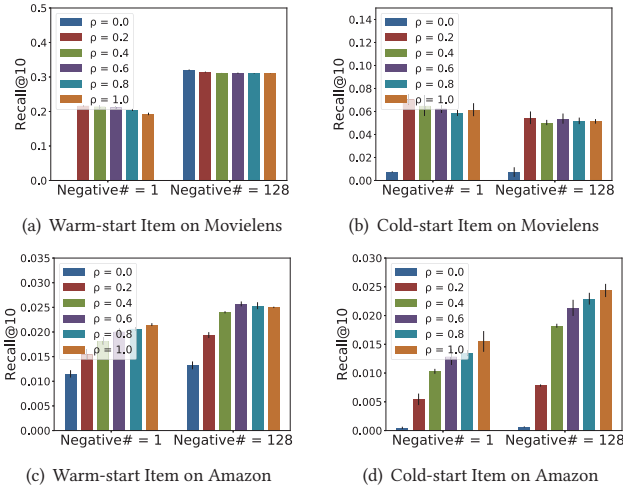
(d) Cold-start Item on Amazon

Figure 3: Effect of U-I Contrastive Embedding Network. Negative# denotes the number of negative pairs.

contrastive learning and develop a general framework with the flexible encoders. To justify the objective function, we compare two different implementations equipped with MF-based and GCN-based CF encoders (CEN_{MF} and CEN_{GCN}) with MF-BPR and LightGCN, which are optimized with BPR loss function. We analyze their experimental results *w.r.t.* recall@10 listed in Table 3:

- Clearly, CEN_{MF} and CEN_{GCN} significantly outperform MF-BPR and LightGCN not only on the cold-start but on the warm-start conditions. It indicates that our proposed objective function is able to effectively model collaborative embedding as well as distill the features associating with the collaborative signal.

- Analyzing the results on the cold-start scenario across the four datasets, we can see that CEN_{MF} performs slightly better than CEN_{GCN} in most cases. One possible reason is that the GCN-based model is enhanced by injecting the items' local structure information into their collaborative embeddings, while such information is hard to learn purely from their own content information by the feature encoder.

- Although LightGCN could encode the local structure information into collaborative embeddings, we find the results of MF-based methods are superior on Movielens dataset. We attribute it to the dense interaction per user on this dataset, which probably results in the smoothness of collaborative embeddings and negatively influences the GCN-based recommender system.

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 λ, which implements the 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
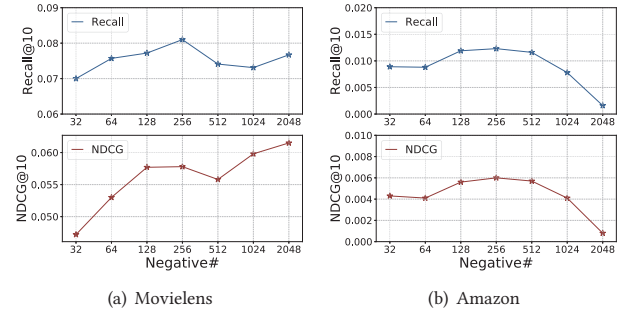

(a) Movielens

(b) Amazon

Figure 4: Effect of R-E Contrastive Embedding Network *w.r.t.* recall@10 and NDCG@10.

training, we discard R-E CEN and varied the hybrid sampling probability ρ 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 3 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 ρ, 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 ρ. 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 ρ, 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 4, 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
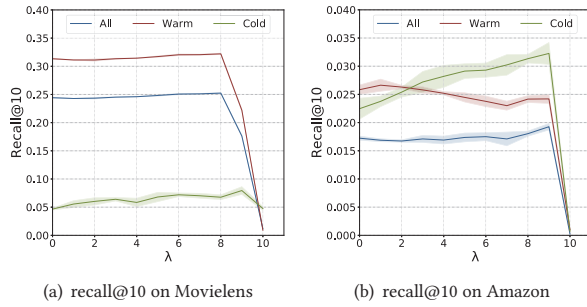
(a) recall@10 on Movielens   (b) recall@10 on Amazon

Figure 5: Performance comparison *w.r.t.* different λ.

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.*, λ) 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 5, 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 λ 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 $\mathcal{L}_{RE}$ 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 λ 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 λ is essential to balance these two kinds of mutual information for pursuing the best performance on all of scenarios.

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, as the bridge to model the collaborative signal for cold-start items. For instance, DropoutNET [36] and CC-CC [31] 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, 22, 32, 47]

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 is widely used in the field of computer vision [4, 6, 17] and natural language processing [5, 26, 27]. 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.* [26] proposed a probabilistic contrastive loss, termed InfoNCE, in a way that maximally preserves the mutual information between the observation and context signals. And, Hjelm *et al.* [15] 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 recommendation, several models [10, 41, 46] employ the contrastive learning to optimize the representation learning. 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 to learn the collaborative signal from the content information.

5 CONCLUSION AND FUTURE WORK

In this work, we 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 and disentangled user and item representation learning. Besides, we expect to investigate the contrastive learning in various applications, like video localization [25] and fashion recommendation [42, 43, 45].

6 ACKNOWLEDGEMENT

REFERENCES

[1] Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2016. A simple but tough-to-beat baseline for sentence embeddings. In *ICLR*. 1–16.

[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] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. *ICML*.

[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. *AAAI*.

[6] Ching-Yao Chuang, Joshua Robinson, Lin Yen-Chen, Antonio Torralba, and Stefanie Jegelka. 2020. Debiased contrastive learning. *NIPS*.

[7] Xiaoyu Du, Xiang Wang, Xiangnan He, Zechao Li, Jinhui Tang, and Tat-Seng Chua. 2020. How to Learn Item Representation for Cold-Start Multimedia Recommendation?. In *ACM MM*. 3469–3477.

[8] 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.

[9] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *IJCAI*. 249–256.

[10] Chaoyang He, Tian Xie, Yu Rong, Wenbing Huang, Yanfang Li, Junzhou Huang, Xiang Ren, and Cyrus Shahabi. 2019. Bipartite Graph Neural Networks for Efficient Node Representation Learning. *arXiv preprint arXiv:1906.11994* (2019).

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*. 770–778.

[12] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. *SIGIR*, 639–648.

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

[14] Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, et al. 2017. CNN architectures for large-scale audio classification. In *ICASSP*. 131–135.

[15] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. 2019. Learning deep representations by mutual information estimation and maximization. *ICLR*.

[16] Hao Jiang, Wenjie Wang, Yinwei Wei, Zan Gao, Yinglong Wang, and Liqiang Nie. 2020. What Aspect Do You Like: Multi-scale Time-aware User Interest Modeling for Micro-video Recommendation. In *ACM MM*. 3487–3495.

[17] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *NIPS*.

[18] Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*. 1–16.

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

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

[21] 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.

[22] Jingjing Li, Mengmeng Jing, Ke Lu, Lei Zhu, Yang Yang, and Zi Huang. 2019. From zero-shot learning to cold-start recommendation. In *AAAI*, Vol. 33. 4189–4196.

[23] Fan Liu, Zhiyong Cheng, Changchang Sun, Yinglong Wang, Liqiang Nie, and Mohan Kankanhalli. 2019. User Diverse Preference Modeling by Multimodal

[24] Fan Liu, Zhiyong Cheng, Lei Zhu, Zan Gao, and Liqiang Nie. 2021. Interest-Aware Message-Passing GCN for Recommendation. In *WWW*. ACM, 1296–1305.

[25] Meng Liu, Xiang Wang, Liqiang Nie, Qi Tian, Baoquan Chen, and Tat-Seng Chua. 2018. Cross-Modal Moment Localization in Videos. In *ACM MM*. 843–851.

[26] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748* (2018).

[27] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese bert-networks. *EMNLP* (2019).

[28] Steffen Rendle and Christoph Freudenthaler. 2014. Improving pairwise learning for item recommendation from implicit feedback. In *WSDM*. 273–282.

[29] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. *CUAI*, 452–461.

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

[31] 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 *CIKM*. 1451–1460.

[32] Ian Soboroff and Charles Nicholas. 1999. Combining content and collaboration in text filtering. In *IJCAI*, Vol. 99. 86–91.

[33] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *JMLR* 15, 1 (2014), 1929–1958.

[34] Yonglong Tian, Dilip Krishnan, and Phillip Isola. 2019. Contrastive Multiview Coding. *arXiv preprint arXiv:1906.05849* (2019).

[35] Manasi Vartak, Arvind Thiagarajan, Conrado Miranda, Jeshua Bratman, and Hugo Larochelle. 2017. A meta-learning perspective on cold-start recommendations for items. *NIPS* (2017).

[36] Maksims Volkovs, Guangwei Yu, and Tomi Poutanen. 2017. Dropoutnet: Addressing cold start in recommender systems. In *NIPS*. 4957–4966.

[37] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural Graph Collaborative Filtering. In *SIGIR*. 165–174.

[38] Yinwei Wei, Xiang Wang, Weili Guan, Liqiang Nie, Zhouchen Lin, and Baoquan Chen. 2019. Neural multimodal cooperative learning toward micro-video understanding. *TIP* 29 (2019), 1–14.

[39] Yinwei Wei, Xiang Wang, Xiangnan He, Liqiang Nie, Yong Rui, and Tat-Seng Chua. 2021. Hierarchical User Intent Graph Network for Multimedia Recommendation. *TMM* (2021).

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

[41] Xu Xie, Fei Sun, Zhaoyang Liu, Jinyang Gao, Bolin Ding, and Bin Cui. 2020. Contrastive Pre-training for Sequential Recommendation. *arXiv preprint arXiv:2010.14395* (2020).

[42] Xun Yang, Xiangnan He, Xiang Wang, Yunshan Ma, Fuli Feng, Meng Wang, and Tat-Seng Chua. 2019. Interpretable fashion matching with rich attributes. In *SIGIR*. 775–784.

[43] Xun Yang, Yunshan Ma, Lizi Liao, Meng Wang, and Tat-Seng Chua. 2019. Transnfcm: Translation-based neural fashion compatibility modeling. In *AAAI*, Vol. 33. 403–410.

[44] Xuzheng Yu, Tian Gan, Yinwei Wei, Zhiyong Cheng, and Liqiang Nie. 2020. Personalized Item Recommendation for Second-hand Trading Platform. In *ACM MM*. 3478–3486.

[45] Na Zheng, Xuemeng Song, Zhaozheng Chen, Linmei Hu, Da Cao, and Liqiang Nie. 2019. Virtually Trying on New Clothing with Arbitrary Poses. In *ACM MM*. ACM, 266–274.

[46] Chang Zhou, Jianxin Ma, Jianwei Zhang, Jingren Zhou, and Hongxia Yang. 2020. Contrastive learning for debiased candidate generation in large-scale recommender systems. *arXiv preprint cs.IR/2005.12964* (2020).

[47] Ziwei Zhu, Shahin Sefati, Parsa Saadatpanah, and James Caverlee. 2020. Recommendation for New Users and New Items via Randomized Training and Mixture-of-Experts Transformation. In *SIGIR*. 1121–1130.