

# Cold-start Bundle Recommendation via Popularity-based Coalescence and Curriculum Heating

Hyunsik Jeon\*

UC San Diego  
San Diego, CA, USA  
hyjeon@ucsd.edu

Jong-eun Lee

Seoul National University  
Seoul, Republic of Korea  
kjayjay40@snu.ac.kr

Jeongin Yun

Seoul National University  
Seoul, Republic of Korea  
yji00828@snu.ac.kr

U Kang

Seoul National University  
Seoul, Republic of Korea  
ukang@snu.ac.kr

## ABSTRACT

How can we recommend cold-start bundles to users? The cold-start problem in bundle recommendation is crucial because new bundles are continuously created on the Web for various marketing purposes. Despite its importance, existing methods for cold-start item recommendation are not readily applicable to bundles. They depend overly on historical information, even for less popular bundles, failing to address the primary challenge of the highly skewed distribution of bundle interactions. In this work, we propose CoHEAT (Popularity-based Coalescence and Curriculum Heating), an accurate approach for cold-start bundle recommendation. CoHEAT first represents users and bundles through graph-based views, capturing collaborative information effectively. To estimate the user-bundle relationship more accurately, CoHEAT addresses the highly skewed distribution of bundle interactions through a popularity-based coalescence approach, which incorporates historical and affiliation information based on the bundle’s popularity. Furthermore, it effectively learns latent representations by exploiting curriculum learning and contrastive learning. CoHEAT demonstrates superior performance in cold-start bundle recommendation, achieving up to 193% higher nDCG@20 compared to the best competitor.

## CCS CONCEPTS

• Information systems → Recommender systems.

## KEYWORDS

bundle recommendation, curriculum learning, contrastive learning

### ACM Reference Format:

Hyunsik Jeon, Jong-eun Lee, Jeongin Yun, and U Kang. 2024. Cold-start Bundle Recommendation via Popularity-based Coalescence and Curriculum Heating. In *Proceedings of the ACM Web Conference 2024 (WWW '24)*, May 13–17, 2024, Singapore, Singapore. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3589334.3645377>

## 1 INTRODUCTION

How can we recommend cold-start bundles to users? Recommender systems have been significant in online platforms providing users with a personalized list of relevant items [30–34, 37, 39]. In recent years, bundle recommendation has garnered significant attention in

\*This work was done when the author was at Seoul National University.

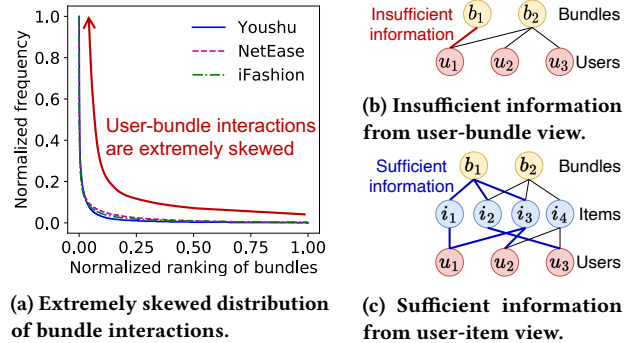
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

WWW '24, May 13–17, 2024, Singapore, Singapore

© 2024 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-0171-9/24/05

<https://doi.org/10.1145/3589334.3645377>



(a) Extremely skewed distribution of bundle interactions.

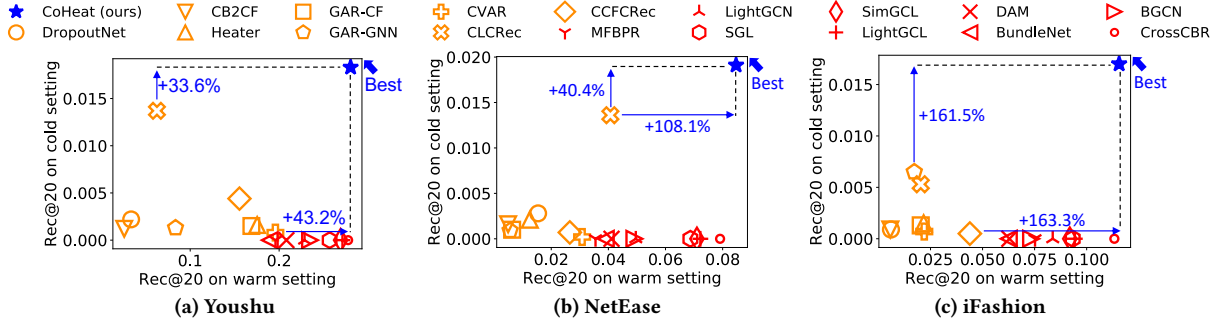
(c) Sufficient information from user-item view.

**Figure 1: (a) Extremely skewed distribution of bundle interactions in real-world datasets (data statistics are summarized in Table 1). (b-c) For an unpopular bundle, user-bundle view provides insufficient information while user-item view provides sufficient information.**

both academia and industry since it enables providers to offer items to users with one-stop convenience [35]. In particular, recommending new bundles to users (i.e. cold-start bundle recommendation) has become crucial with the Web’s evolution as new bundles are constantly created on the Web for various marketing purposes [12].

In recent years, bundle recommendation has seen advancements mainly through matrix factorization-based approaches [4, 6, 12, 24, 40] and graph learning-based approaches [8, 9, 17, 35, 41, 57, 64, 68, 69]. However, they have been developed for a warm-start setting, where all bundles already possess historical interactions with users. Consequently, their efficacy diminishes in cold-start scenarios, where certain bundles are devoid of historical interactions. This is because warm-start methods rely highly on historical information of user-bundle interactions to discern collaborative signals between users and bundles [47].

On the other hand, the cold-start problem [44] in item recommendation has been extensively studied, with a focus on aligning behavior representations with content representations. For instance, generative methods have aimed to model the generation of item behavior representations using mean squared error [48, 53], metric learning [67], or adversarial loss [11]. Dropout-based methods [51, 73] have aimed to bolster robustness to behavior information by randomly dropping the behavior embedding in the training phase. More recently, contrastive learning-based methods [58, 72] have shown superior performance by reducing the discrepancy between the distributions of behavior and content information of items. However, the existing methods for cold-start item recommendation fail to achieve high performance in bundle recommendation because they lack the ability to effectively leverage the user-item historical interactions when representing bundles. Furthermore,



**Figure 2: Performance comparison of CoHEAT with competitors on Youshu, NetEase, and iFashion datasets, evaluated by Recall@20. We mark cold-start methods as orange, and warm-start methods as red. The cold-start methods typically sacrifice warm setting performance to excel in cold settings. The warm-start methods show poor performance in cold settings. CoHEAT demonstrates superior performance over existing methods in both cold and warm settings.**

none of the existing works have explicitly considered the skewed distribution of user-bundle interactions which is a pivotal aspect in bundle recommendation as shown in Figure 1a. For unpopular bundles, aligning behavior representations from insufficient historical information with content representations amplifies inherent biases and makes it difficult to learn meaningful representations; this results in sacrificing the performance on a warm-start setting to improve the performance on a cold-start setting (see Figure 2).

In this paper, we propose CoHEAT (Popularity-based Coalescence and Curriculum Heating), to address the cold-start bundle recommendation. CoHEAT constructs representations of users and bundles using two distinct graph-based views: user-bundle view and user-item view. The user-bundle view is grounded in historical interactions between users and bundles, whereas the user-item view is rooted in bundle affiliations and historical interactions between users and items. To handle the extremely skewed distribution as shown in Figure 1a, CoHEAT strategically leverages both views in its predictions by our proposed popularity-based coalescence; this emphasizes user-item view for less popular bundles since they provide richer information than the sparse user-bundle view, as shown in Figures 1b and 1c. In addition, to effectively learn the user-item view representations which are fully used for cold-start bundles, CoHEAT exploits a curriculum learning approach that gradually shifts the training focus from the user-bundle view to the user-item view. CoHEAT further exploits a contrastive learning approach to align the representations of the two views effectively.

Our contributions are summarized as follows:

- **Problem.** To our knowledge, this is the first work that tackles the cold-start problem in bundle recommendation, a challenging problem of significant impact in real-world scenarios.
- **Method.** We propose CoHEAT, an accurate cold-start bundle recommendation method. CoHEAT learns user and bundle representations by addressing highly skewed interactions, enabling it to recommend cold bundles through their affiliations.
- **Experiments.** We experimentally show that CoHEAT provides the state-of-the-art performance achieving up to 193% higher nDCG@20 compared to the best competitor in cold-start scenarios while maintaining competitive performance in warm-start scenarios. (see Figure 2 and Table 2).

The code is available at <https://github.com/snudatalab/CoHeat>.

## 2 PRELIMINARIES

### 2.1 Problem Definition

The problem of cold-start bundle recommendation is defined as follows. Let  $\mathcal{U}$ ,  $\mathcal{B}$ , and  $\mathcal{I}$  be the sets of users, bundles, and items, respectively. Among the bundles,  $\mathcal{B}_w \subset \mathcal{B}$  refers to the warm-start bundles that have at least one historical interaction with users, while  $\mathcal{B}_c = \mathcal{B} \setminus \mathcal{B}_w$  represents the cold-start bundles that lack any historical interaction with users. The observed user-bundle interactions, user-item interactions, and bundle-item affiliations are defined respectively as  $\mathcal{X} = \{(u, b) | u \in \mathcal{U}, b \in \mathcal{B}_w\}$ ,  $\mathcal{Y} = \{(u, i) | u \in \mathcal{U}, i \in \mathcal{I}\}$ , and  $\mathcal{Z} = \{(b, i) | b \in \mathcal{B}, i \in \mathcal{I}\}$ . Given  $\{\mathcal{X}, \mathcal{Y}, \mathcal{Z}\}$ , our goal is to recommend  $k$  bundles from  $\mathcal{B}$  to each user  $u \in \mathcal{U}$ . Note that the given interactions are observed only for warm bundles but the objective includes recommending also cold bundles to users.

The central challenge in cold-start bundle recommendation, compared to traditional bundle recommendation, lies in accurately predicting the relationship between a user  $u \in \mathcal{U}$  and a cold-start bundle  $b \in \mathcal{B}_c$  in the absence of any historical interactions of  $b$ . Hence, the crux of addressing the problem is to effectively estimate the representations of cold-start bundles using their affiliation information.

Bundles are compositions of multiple items, each having distinct interactions with users. This contrasts sharply with traditional cold-start item recommendations where contents are often represented as independent entities like texts or images. Moreover, user-bundle interactions exhibit a pronounced skewness, far more than typical user-item interactions. These make the cold-start bundle recommendation uniquely challenging compared to the cold-start item recommendation.

### 2.2 Curriculum Learning

Curriculum learning, inspired by human learning, structures training from simpler to more complex tasks, unlike standard approaches that randomize task order [3, 55]. Its effectiveness has been proven in various domains, including computer vision [63, 66], natural language processing [21, 65], robotics [26, 36], and recommender systems [10, 15].

In this work, we harness curriculum learning to enhance the learning process of user-bundle relationships. We initiate with a

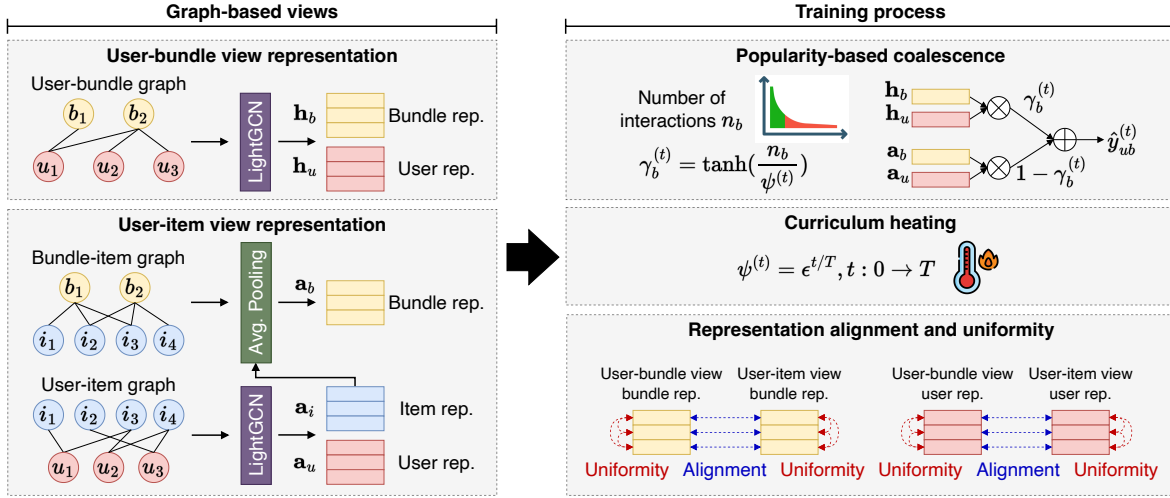


Figure 3: Overview of CoHEAT (see Section 3 for details).

focus on the more straightforward user-bundle view embeddings and then progressively shift attention to the intricate user-item view embeddings. This strategy stems from the ease of learning user-bundle view embeddings, which directly capture collaborative signals from historical interactions. In contrast, user-item view embeddings are more complicated due to their dependence on the representations of affiliated items.

### 2.3 Contrastive Learning

Contrastive learning aims to derive meaningful embeddings by distinguishing between similar and dissimilar data samples. This approach has consistently demonstrated superior performance across a range of research fields, including computer vision [13, 43, 54], natural language processing [19, 61], and recommender systems [5, 52]. In bundle recommendation, CrossCBR [35] has utilized InfoNCE [49] as a contrastive learning function to regularize embeddings of users and bundles between the user-bundle and user-item views. However, its approach of aligning the two views without differentiation in prediction can be limiting, especially in cold-start scenarios where the user-bundle view is sparse.

In this work, we enhance the application of contrastive learning in bundle recommendation. Instead of treating the two views equally, we dynamically adjust their weights based on bundle popularity. This facilitates the transfer of information from a more informative view to the counterpart, enabling effective recommendations for both cold and warm bundles. Furthermore, we leverage the alignment and uniformity loss [54], which has been demonstrated to outperform InfoNCE in various applications [52, 54, 60]. This loss directly optimizes the foundational principles of contrastive learning, ensuring more robust and meaningful embeddings.

## 3 PROPOSED METHOD

### 3.1 Overview

We address the following challenges to achieve high performance in cold-start bundle recommendation.

- C1. **Handling highly skewed interactions.** Previous works depend overly on user-bundle view representations, which

are unreliable if bundles have sparse interactions. How can we effectively learn the representations from highly skewed interactions?

- C2. **Effectively learning user-item view representations.** Despite the ample information provided by the user-item view, multiple items in a bundle complicate the learning of these representations. How can we effectively learn the user-item view representations?
- C3. **Bridging the gap between two view representations.** Aligning user-bundle and user-item views is crucial, as we estimate future interactions of cold bundles using only their affiliations. How can we effectively reconcile these two view representations?

To address these challenges, we propose CoHEAT (Popularity-based Coalescence and Curriculum Heating) with the following main ideas.

- I1. **Popularity-based coalescence.** For the score between users and bundles, we propose the coalescence of two view scores, with less popular bundles relying more on user-item view scores and less on user-bundle view scores.
- I2. **Curriculum heating.** We propose a curriculum learning approach that focuses initially on training representations using the user-bundle view, gradually shifting the focus to the user-item view.
- I3. **Representation alignment and uniformity.** We exploit a representation alignment and uniformity approach to effectively reconcile the user-bundle view and user-item view representations.

Figure 3 depicts the schematic illustration of CoHEAT. Given user-bundle interactions, user-item interactions, and bundle-item affiliations, CoHEAT forms two graph-based views. Then, it predicts user-bundle scores by coalescing scores from both views based on bundle popularity. During training, CoHEAT prioritizes user-bundle view initially, transitioning progressively to user-item view via curriculum heating. CoHEAT also exploits alignment and uniformity loss to regularize both views.

### 3.2 Two Graph-based Views

The objective of bundle recommendation is to estimate the relationship between users and bundles by learning their latent representations. We utilize graph-based representations of users and bundles to fully exploit the given user-bundle interactions, user-item interactions, and bundle-item affiliations. We construct user-bundle view and user-item view graphs and use LightGCN [23] to obtain embeddings of users and bundles [35].

**User-bundle view representation and score.** In user-bundle view, we aim to capture the behavior signal between users and bundles. Specifically, we construct a bipartite graph using user-bundle interactions, and propagate the historical information using a LightGCN. The  $k$ -th layer of the LightGCN is computed as follows:

$$\mathbf{h}_u^{(k)} = \sum_{b \in \mathcal{N}_u} \frac{1}{\sqrt{|\mathcal{N}_u|} \sqrt{|\mathcal{N}_b|}} \mathbf{h}_b^{(k-1)}, \mathbf{h}_b^{(k)} = \sum_{u \in \mathcal{N}_b} \frac{1}{\sqrt{|\mathcal{N}_b|} \sqrt{|\mathcal{N}_u|}} \mathbf{h}_u^{(k-1)}, \quad (1)$$

where  $\mathbf{h}_u^{(k)}, \mathbf{h}_b^{(k)} \in \mathbb{R}^d$  are the embeddings of user  $u$  and bundle  $b$  at  $k$ -th layer, respectively;  $\mathcal{N}_u$  and  $\mathcal{N}_b$  are the sets of user  $u$ 's neighbors and bundle  $b$ 's neighbors in the user-bundle graph, respectively.  $\mathbf{h}_u^{(0)}, \mathbf{h}_b^{(0)} \in \mathbb{R}^d$  are randomly initialized before the training of the model. We obtain the user-bundle view representations of user  $u$  and bundle  $b$  by aggregating the embeddings from all layers with a weighting approach that places greater emphasis on the lower layers as follows:

$$\mathbf{h}_u = \sum_{k=0}^K \frac{1}{k+1} \mathbf{h}_u^{(k)}, \mathbf{h}_b = \sum_{k=0}^K \frac{1}{k+1} \mathbf{h}_b^{(k)}, \quad (2)$$

where  $\mathbf{h}_u, \mathbf{h}_b \in \mathbb{R}^d$  are the user-bundle view embeddings of user  $u$  and bundle  $b$ , respectively;  $K$  denotes the last layer. Finally, the user-bundle view score between user  $u$  and bundle  $b$  is defined as  $h_{ub} = \mathbf{h}_u^\top \mathbf{h}_b$ .

**User-item view representation and score.** In user-item view, we aim to learn the relationship between users and bundles from the perspective of item affiliations. Specifically, we construct a bipartite graph using user-item interactions, and propagate the historical information using another LightGCN. Then, we obtain bundle representations by aggregating the affiliated items' representations. The  $k$ -th layer of the LightGCN is computed as follows:

$$\mathbf{a}_u^{(k)} = \sum_{i \in \mathcal{N}'_u} \frac{1}{\sqrt{|\mathcal{N}'_u|} \sqrt{|\mathcal{N}'_i|}} \mathbf{a}_i^{(k-1)}, \mathbf{a}_i^{(k)} = \sum_{u \in \mathcal{N}'_i} \frac{1}{\sqrt{|\mathcal{N}'_i|} \sqrt{|\mathcal{N}'_u|}} \mathbf{a}_u^{(k-1)}, \quad (3)$$

where  $\mathbf{a}_u^{(k)}, \mathbf{a}_i^{(k)} \in \mathbb{R}^d$  are the embeddings of user  $u$  and item  $i$  at  $k$ -th layer, respectively;  $\mathcal{N}'_u$  and  $\mathcal{N}'_i$  are the sets of user  $u$ 's neighbors and item  $i$ 's neighbors in the user-item graph, respectively.  $\mathbf{a}_u^{(0)}, \mathbf{a}_i^{(0)} \in \mathbb{R}^d$  are randomly initialized before the training. We obtain the user-item view representations of user  $u$  and item  $i$  by aggregating the embeddings from all layers with a weighting approach as follows:

$$\mathbf{a}_u = \sum_{k=0}^K \frac{1}{k+1} \mathbf{a}_u^{(k)}, \mathbf{a}_i = \sum_{k=0}^K \frac{1}{k+1} \mathbf{a}_i^{(k)}, \quad (4)$$

where  $\mathbf{a}_u, \mathbf{a}_i \in \mathbb{R}^d$  are the user-item view embeddings of user  $u$  and item  $i$ , respectively;  $K$  indicates the last layer. We then obtain the user-item view representations of bundle  $b$  by an average pooling as  $\mathbf{a}_b = \frac{1}{|\mathcal{N}'_b|} \sum_{i \in \mathcal{N}'_b} \mathbf{a}_i$ , where  $\mathcal{N}'_b$  is the set of bundle  $b$ 's affiliated

items. Finally, the user-item view score between user  $u$  and bundle  $b$  is defined as  $a_{ub} = \mathbf{a}_u^\top \mathbf{a}_b$ .

### 3.3 Popularity-based Coalescence

For recommending bundles to users, our objective is to estimate the final score  $\hat{y}_{ub} \in \mathbb{R}$  between user  $u$  and bundle  $b$  using scores  $h_{ub}$  and  $a_{ub}$ , derived from the two distinct views. However, real-world datasets present an inherent challenge of handling the extremely skewed distribution of interactions between users and bundles, as illustrated in Figure 1a. While both views are informative, many unpopular bundles are underrepresented in the user-bundle view due to the insufficient interactions as illustrated in Figure 1b. In contrast, they are often sufficiently represented in the user-item view, as depicted in Figure 1c. A uniform weighting strategy for both views, as in CrossCBR, risks amplifying biases inherent to the user-bundle view, especially for the unpopular bundles. This predicament is further exacerbated for cold-start bundles devoid of interactions in user-bundle view.

To deal with this challenge, we propose two desired properties for the user-bundle relationship score  $\hat{y}_{ub}$ .

**Property 1** (User-bundle view influence mitigation): The influence of user-bundle view score should be mitigated as a bundle's interaction number decreases, i.e.  $\frac{\partial \hat{y}_{ub}}{\partial h_{ub}} < \frac{\partial \hat{y}_{ub'}}{\partial h_{ub'}}$  if  $n_b < n_{b'}$ , where  $n_b$  is the number of user interactions of bundle  $b$ .

**Property 2** (User-item view influence amplification): The influence of user-item view score should be amplified as a bundle's interaction number decreases, i.e.  $\frac{\partial \hat{y}_{ub}}{\partial a_{ub}} > \frac{\partial \hat{y}_{ub'}}{\partial a_{ub'}}$  if  $n_b < n_{b'}$ , where  $n_b$  is the number of user interactions of bundle  $b$ .

Properties 1 and 2 are crucial in achieving a balanced interplay between the user-bundle view and user-item view scores based on bundle popularities. Specifically, they ensure a heightened emphasis on the user-item view over the user-bundle view for less popular bundles.

We propose the user-bundle relationship score  $\hat{y}_{ub}$  that satisfies the two desired properties by weighting the two scores  $h_{ub}$  and  $a_{ub}$  based on bundle popularities as follows:

$$\hat{y}_{ub} = \gamma_b h_{ub} + (1 - \gamma_b) a_{ub}, \quad (5)$$

where  $\gamma_b \in [0, 1]$ , which is defined in the next subsection, denotes a weighting coefficient such that  $\gamma_b > \gamma_{b'}$  if  $n_b > n_{b'}$ . A smaller value of  $\gamma_b$  (i.e. a smaller value of  $n_b$ ) ensures that the score  $\hat{y}_{ub}$  is predominantly influenced by the user-item view score  $a_{ub}$ . We show in Lemmas 3.1 and 3.2 that Equation (5) satisfies all the desired properties.

LEMMA 3.1. Equation (5) satisfies Property 1.

PROOF.  $\frac{\partial \hat{y}_{ub}}{\partial h_{ub}} = \gamma_b$ . Thus,  $\frac{\partial \hat{y}_{ub}}{\partial h_{ub}} < \frac{\partial \hat{y}_{ub'}}{\partial h_{ub'}}$  if  $n_b < n_{b'}$ , because  $\gamma_b < \gamma_{b'}$ .  $\square$

LEMMA 3.2. Equation (5) satisfies Property 2.

PROOF.  $\frac{\partial \hat{y}_{ub}}{\partial a_{ub}} = 1 - \gamma_b$ . Thus,  $\frac{\partial \hat{y}_{ub}}{\partial a_{ub}} > \frac{\partial \hat{y}_{ub'}}{\partial a_{ub'}}$  if  $n_b < n_{b'}$ , because  $1 - \gamma_b > 1 - \gamma_{b'}$ .  $\square$

### 3.4 Curriculum Heating

Despite the ample information provided by the user-item view, multiple items in a bundle complicate the learning of user-item

representations. This difficulty arises because accurate representation of a bundle necessitates well-represented embeddings of its all affiliated items, and each item further requires well-represented embeddings of the connected users. On the other side, the user-bundle view representation is relatively straightforward to learn. This simplicity arises because we encapsulate each bundle's historical characteristics into a single embedding rather than understanding the intricate composition of the bundle.

Hence, we propose to focus initially on learning easier view representations and gradually shift the focus to learning harder view representations. Thus, we modify Equation (5) by exploiting a curriculum learning approach that focuses initially on training user-bundle view representations, and gradually shifts the focus to the user-item view representations as follows:

$$\hat{y}_{ub}^{(t)} = \gamma_b^{(t)} h_{ub} + (1 - \gamma_b^{(t)}) a_{ub}, \quad (6)$$

where  $\hat{y}_{ub}^{(t)} \in \mathbb{R}$  is the estimated relationship score between user  $u$  and bundle  $b$  at epoch  $t$ .  $\gamma_b^{(t)} \in \mathbb{R}$  is defined as  $\gamma_b^{(t)} = \tanh\left(\frac{n_b}{\psi^{(t)}}\right)$ , where  $n_b$  is the number of interactions of bundle  $b$ , and  $\psi^{(t)} > 0$  is the temperature at epoch  $t$ . Note that  $\gamma_b^{(t)}$  lies within the interval  $[0, 1]$  because  $\frac{n_b}{\psi^{(t)}} \geq 0$ . Then, we incrementally raise the temperature  $\psi^{(t)}$  up to the maximum temperature as follows:

$$\psi^{(t)} = e^{t/T}, t: 0 \rightarrow T, \quad (7)$$

where  $t, T \in \mathbb{R}$  are the current and the maximum epochs of the training process, and  $e > 1$  is the hyperparameter of the maximum temperature. In the initial epochs of training,  $\gamma_b^{(t)}$  is large since  $t$  is small. As a result, the score  $\hat{y}_{ub}^{(t)}$  relies more heavily on  $h_{ub}$  than  $a_{ub}$ . However, as the training progresses and  $t$  increases,  $\gamma_b^{(t)}$  diminishes, shifting the emphasis from  $h_{ub}$  to  $a_{ub}$ . This heating mechanism is applied to all bundles regardless of their popularity. Furthermore, we show in Lemmas 3.3 and 3.4 that Equation (6) still satisfies the two desired properties.

LEMMA 3.3. Equation (6) satisfies Property 1.

PROOF.  $\frac{\partial \hat{y}_{ub}^{(t)}}{\partial h_{ub}} = \tanh\left(\frac{n_b}{\psi^{(t)}}\right)$ . Thus,  $\frac{\partial \hat{y}_{ub}^{(t)}}{\partial h_{ub}} < \frac{\partial \hat{y}_{ub'}^{(t)}}{\partial h_{ub'}}$  if  $n_b < n_{b'}$  because  $\psi^{(t)}$  is the same for all bundles at epoch  $t$  and  $\tanh(\cdot)$  is an increasing function.  $\square$

LEMMA 3.4. Equation (6) satisfies Property 2.

PROOF.  $\frac{\partial \hat{y}_{ub}^{(t)}}{\partial a_{ub}} = 1 - \tanh\left(\frac{n_b}{\psi^{(t)}}\right)$ . Thus,  $\frac{\partial \hat{y}_{ub}^{(t)}}{\partial a_{ub}} > \frac{\partial \hat{y}_{ub'}^{(t)}}{\partial a_{ub'}}$  if  $n_b < n_{b'}$  because  $\psi^{(t)}$  is the same for all bundles at epoch  $t$  and  $1 - \tanh(\cdot)$  is a decreasing function.  $\square$

### 3.5 Representation Alignment and Uniformity

While the user-bundle view and user-item view are crafted to capture distinct representations, aligning the two views is essential, especially when predicting future interactions of cold bundles solely based on user-item view representations. Moreover, aligning two views facilitates knowledge transfer between the two views. This is essential because we gradually change the learning focus of views by curriculum heating, and the alignment helps the success of curriculum heating by effectively transferring knowledge from a

view with richer knowledge to the opposite view. To achieve this, we exploit a contrastive learning-based approach that reconciles the two views. Specifically, we use the alignment and uniformity loss [54] as a regularization for the representations of the two views. We firstly  $l_2$ -normalize the embeddings of the two views as follows:

$$\tilde{\mathbf{h}}_u = \frac{\mathbf{h}_u}{\|\mathbf{h}_u\|_2}, \tilde{\mathbf{a}}_u = \frac{\mathbf{a}_u}{\|\mathbf{a}_u\|_2}, \tilde{\mathbf{h}}_b = \frac{\mathbf{h}_b}{\|\mathbf{h}_b\|_2}, \tilde{\mathbf{a}}_b = \frac{\mathbf{a}_b}{\|\mathbf{a}_b\|_2}, \quad (8)$$

where  $\mathbf{h}_u, \mathbf{h}_b \in \mathbb{R}^d$  are user-bundle view representations of user  $u$  and bundle  $b$ , respectively;  $\mathbf{a}_u, \mathbf{a}_b \in \mathbb{R}^d$  are user-item view representations of user  $u$  and bundle  $b$ , respectively. Then, we define an alignment loss as follows:

$$l_{align} = \mathbb{E}_{u \sim p_{user}} \|\tilde{\mathbf{h}}_u - \tilde{\mathbf{a}}_u\|_2^2 + \mathbb{E}_{b \sim p_{bundle}} \|\tilde{\mathbf{h}}_b - \tilde{\mathbf{a}}_b\|_2^2, \quad (9)$$

where  $p_{user}$  and  $p_{bundle}$  are the distributions of users and bundles, respectively. The alignment loss makes the embeddings of the two views close to each other for each user and bundle. We also define a uniformity loss as follows:

$$l_{uniform} = \log \mathbb{E}_{u, u' \sim p_{user}} e^{-2\|\tilde{\mathbf{h}}_u - \tilde{\mathbf{h}}_{u'}\|_2^2} + \log \mathbb{E}_{u, u' \sim p_{user}} e^{-2\|\tilde{\mathbf{a}}_u - \tilde{\mathbf{a}}_{u'}\|_2^2} \\ + \log \mathbb{E}_{b, b' \sim p_{bundle}} e^{-2\|\tilde{\mathbf{h}}_b - \tilde{\mathbf{h}}_{b'}\|_2^2} + \log \mathbb{E}_{b, b' \sim p_{bundle}} e^{-2\|\tilde{\mathbf{a}}_b - \tilde{\mathbf{a}}_{b'}\|_2^2}, \quad (10)$$

where  $u'$  and  $b'$  denote a user and a bundle distinct from  $u$  and  $b$ , respectively. The uniformity loss ensures distinct representations for different users (or bundles) by scattering them across the space. Finally, we define the contrastive loss for the two views as follows:

$$\mathcal{L}_{AU} = l_{align} + l_{uniform}. \quad (11)$$

### 3.6 Objective Function and Training

To effectively learn the user-bundle relationship, we utilize Bayesian Personalize Ranking (BPR) loss [42], which is the most widely used loss owing to its powerfulness, as follows:

$$\mathcal{L}_{BPR}^{(t)} = \mathbb{E}_{(u, b^+, b^-) \sim p_{data}} -\ln \sigma(\hat{y}_{ub^+}^{(t)} - \hat{y}_{ub^-}^{(t)}), \quad (12)$$

where  $p_{data}$  is the data distribution of user-bundle interactions, with  $u$  denoting a user,  $b^+$  indicating a positive bundle, and  $b^-$  representing a negative bundle. We define the final objective function as follows:

$$\mathcal{L}^{(t)} = \mathcal{L}_{BPR}^{(t)} + \lambda_1 \mathcal{L}_{AU} + \lambda_2 \|\Theta\|_2, \quad (13)$$

where  $\lambda_1, \lambda_2 \in \mathbb{R}$  are balancing hyperparameters for the terms, and  $\Theta$  denotes trainable parameters of CoHEAT. For the distributions  $p_{user}$  and  $p_{bundle}$ , we use in-batch sampling which selects samples from the training batch of  $p_{data}$  rather than the entire dataset. This approach has empirically demonstrated to mitigate the training bias in prior studies [52, 71]. All the parameters are optimized in an end-to-end manner through the optimization. We also adopt an edge dropout [35, 59] while training to enhance the performance robustness.

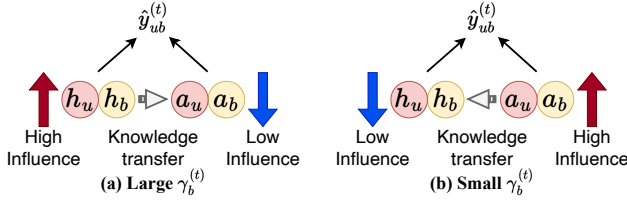
### 3.7 Discussion of CoHEAT

The core of CoHEAT lies in its ability to dynamically adjust the weights of two distinct views, setting it apart from previous methods such as CrossCBR [35]. This dynamic adjustment is pivotal for achieving superior performance in the cold-start bundle recommendation.



**Table 1: Summary of three real-world datasets where “dens.” denotes the density of a matrix.**

Dataset	Users	Bundles	Items	User-bundle (dens.)	User-item (dens.)	Bundle-item (dens.)	Avg. size of bundle
Youshu <sup>1</sup>	8,039	4,771	32,770	51,377 (0.13%)	138,515 (0.05%)	176,667 (0.11%)	37.03
NetEase <sup>1</sup>	18,528	22,864	123,628	302,303 (0.07%)	1,128,065 (0.05%)	1,778,838 (0.06%)	77.80
iFashion <sup>1</sup>	53,897	27,694	42,563	1,679,708 (0.11%)	2,290,645 (0.10%)	106,916 (0.01%)	3.86

<sup>1</sup> <https://github.com/mysbupt/CrossCBR>**Figure 4: Learning mechanism of CoHEAT (see Section 3.7 for details).**

Through the popularity-based coalescence, CoHEAT dynamically adjusts the weight  $\gamma_b^{(t)}$  in Equation (6) to effectively harness the more informative view. For instance, when a bundle  $b$  is popular, the influence of user-bundle view is bolstered with a large  $\gamma_b^{(t)}$  because the bundle is rich of knowledge in this view. The knowledge then gets transferred to the user-item view by the alignment and uniformity loss, as depicted in Figure 4 (a). Conversely, for a less popular bundle, the influence of user-item view is amplified with a small  $\gamma_b^{(t)}$ , transferring the learned knowledge to the user-bundle view, as shown in Figure 4 (b). This strategy contrasts with CrossCBR, which may inadvertently amplify the underrepresented knowledge of unpopular bundles due to its uniform weighting strategy.

Additionally, the curriculum heating of CoHEAT further adjusts the weight  $\gamma_b^{(t)}$  throughout the learning process. As the epochs progress,  $\gamma_b^{(t)}$  diminishes (transitioning from Figure 4 (a) to Figure 4 (b)), ensuring both views are thoroughly utilized during training. This dynamic exchange of knowledge between two views results in CoHEAT’s superior performance in both cold and warm settings, owing to the wealth of knowledge each view offers. This strategy is distinct from CrossCBR since the uniform weights for both views may lead to suboptimal results, especially in cold-start scenarios where the user-bundle view is sparse. Moreover, the curriculum heating strategically focuses on the easier view first, gradually shifting its attention to the more challenging view as the learning progresses. This helps a smoother and more effective knowledge transfer between the views.

We analyze the space and time complexities of CoHEAT in Appendix A.

## 4 EXPERIMENTS

We perform experiments to answer the following questions.

- Q1. **Comparison with cold-start methods.** Does CoHEAT show superior performance in comparison to other cold-start methods in bundle recommendation?
- Q2. **Comparison with warm-start methods.** Does CoHEAT show similar performance in warm-start bundle recommendation compared with baselines, although CoHEAT is a cold-start bundle recommendation method?

- Q3. **Comparison by cold bundle ratio.** Does the performance difference between CoHEAT and baseline increase as the cold bundle ratio increases?
- Q4. **Ablation study.** How do the main ideas of CoHEAT affect the performance?
- Q5. **Effect of the maximum temperature.** How does the maximum temperature  $\epsilon$ , the critical hyperparameter, affect the performance of CoHEAT?

### 4.1 Experimental Setup

**Datasets.** We use three real-world bundle recommendation datasets as summarized in Table 1. Youshu [12] comprises bundles of books sourced from a book review site; NetEase [6] features bundles of music tracks from a cloud music service; iFashion [14] consists of bundles of fashion items from an outfit sales platform.

**Baseline cold-start methods.** We compare CoHEAT with existing cold-start item recommendation methods because they can be easily adapted for bundle recommendation by considering bundle-item affiliations as content information. DropoutNet [51] is a robustness-based method with a dropout operation. CB2CF [2] and Heater [73] are constraint-based methods that regularize the alignment. GAR [11] is a generative method with two variants GAR-CF and GAR-GNN. CVAR [70] is another generative method with a conditional decoder. CLCRec [58] and CCFCRec [72] are contrastive learning-based methods. We omit other competitors such as DUIF [20], MTPR [18], and NFM [22] because CLCRec and CCFCRec outperform them by a large margin on their extensive experiments. We also omit other generative competitors such as DeepMusic [48] and MetaEmb [38] because GAR [11] greatly outperforms them on its experiments. We use bundle-item multi-hot vectors as their content information.

**Baseline warm-start methods.** We also compare CoHEAT with previous warm-start recommendation methods. MFBR [42] and LightGCN [23] are item recommendation methods with the modelings of matrix factorization and graph learning, respectively. SGL [59], SimGCL [62], and LightGCL [5] are the improved methods of item recommendation with contrastive learning approaches. DAM [12] is a bundle recommendation method with the modeling of matrix factorization. BundleNet [17], BGCN [8, 9], and CrossCBR [35] are other bundle recommendation methods with the modeling of graph learning.

**Evaluation metrics.** We use Recall@ $k$  and nDCG@ $k$  metrics as in previous works [35, 58]. Recall@ $k$  measures the proportion of relevant items in the top- $k$  list, while nDCG@ $k$  weighs items by their rank. We set  $k$  to 20. In tables, bold and underlined values indicate the best and second-best results, respectively.

**Experimental process.** We conduct experiments in warm-start, cold-start, and all-bundle scenarios as in previous works [58]. For the warm-start scenario, interactions are split into 7:1:2 subsets for training, validation, and testing. In the cold-start scenario, bundles are split in 7:1:2 ratio. In the all-bundle scenario, interactions are

**Table 2: Performance comparison of CoHEAT and baseline cold-start methods on three real-world datasets.**

Model	Youshu						NetEase						iFashion					
	Recall@20			nDCG@20			Recall@20			nDCG@20			Recall@20			nDCG@20		
	Cold	Warm	All	Cold	Warm	All	Cold	Warm	All	Cold	Warm	All	Cold	Warm	All	Cold	Warm	All
DropoutNet [51]	.0022	.0336	.0148	.0007	.0153	.0055	.0028	.0154	.0046	.0015	.0078	.0024	.0009	.0060	.0039	.0008	.0045	.0027
CB2CF [2]	.0012	.0258	.0028	.0007	.0208	.0021	.0016	.0049	.0027	.0006	.0027	.0014	.0009	.0057	.0066	.0006	.0043	.0048
Heater [73]	.0016	.1753	.0541	.0007	.0826	.0286	.0021	.0125	.0102	.0010	.0064	.0054	.0015	.0217	.0123	.0010	.0151	.0083
GAR-CF [11]	.0015	.1688	.0529	.0011	.0726	.0317	.0010	.0063	.0014	.0005	.0035	.0008	.0013	.0203	.0090	.0013	.0143	.0055
GAR-GNN [11]	.0013	.0835	.0358	.0006	.0569	.0178	.0009	.0056	.0027	.0003	.0030	.0012	<u>.0065</u>	.0172	.0126	.0030	.0107	.0087
CVAR [70]	.0008	<u>.1958</u>	<u>.0829</u>	.0002	<u>.1112</u>	<u>.0533</u>	.0002	.0308	.0156	.0001	.0154	.0084	.0007	.0220	.0125	.0004	.0152	.0084
CLCRec [58]	<u>.0137</u>	.0626	.0367	<u>.0087</u>	.0317	.0194	<u>.0136</u>	<u>.0407</u>	<u>.0259</u>	<u>.0075</u>	<u>.0215</u>	<u>.0138</u>	.0053	.0203	.0126	<u>.0043</u>	.0135	.0085
CCFCRec [72]	.0044	.1554	.0702	.0022	.0798	.0425	.0007	.0265	.0130	.0004	.0128	.0068	.0005	<u>.0439</u>	<u>.0252</u>	.0003	<u>.0304</u>	<u>.0172</u>
<b>CoHEAT (ours)</b>	<b>.0183</b>	<b>.2804</b>	<b>.1247</b>	<b>.0105</b>	<b>.1646</b>	<b>.0833</b>	<b>.0191</b>	<b>.0847</b>	<b>.0453</b>	<b>.0093</b>	<b>.0455</b>	<b>.0264</b>	<b>.0170</b>	<b>.1156</b>	<b>.0658</b>	<b>.0096</b>	<b>.0876</b>	<b>.0504</b>

**Table 3: Performance comparison of CoHEAT and baseline warm-start methods on three real-world datasets.**

Model	Youshu		NetEase		iFashion	
	Recall@20	nDCG@20	Recall@20	nDCG@20	Recall@20	nDCG@20
MFBPR [42]	.1959	.1117	.0355	.0181	.0752	.0542
LightGCN [23]	.2286	.1344	.0496	.0254	.0837	.0612
SGL [59]	.2568	.1527	.0687	.0368	.0933	.0690
SimGCL [62]	.2691	.1593	.0710	.0377	.0919	.0677
LightGCL [5]	.2712	.1607	.0722	.0388	.0943	.0686
DAM [12]	.2082	.1198	.0411	.0210	.0629	.0450
BundleNet [17]	.1895	.1125	.0391	.0201	.0626	.0447
BGCN [8, 9]	.2347	.1345	.0491	.0258	.0733	.0531
CrossCBR [35]	<u>.2776</u>	<u>.1641</u>	<u>.0791</u>	<u>.0433</u>	<u>.1133</u>	<u>.0875</u>
<b>CoHEAT (ours)</b>	<b>.2804</b>	<b>.1646</b>	<b>.0847</b>	<b>.0455</b>	<b>.1156</b>	<b>.0876</b>

split in 7:1:2 ratio with a half for warm-start and the other half for cold-start bundles. We report the best Recall@20 and nDCG@20 within 100 epochs, averaged over three runs.

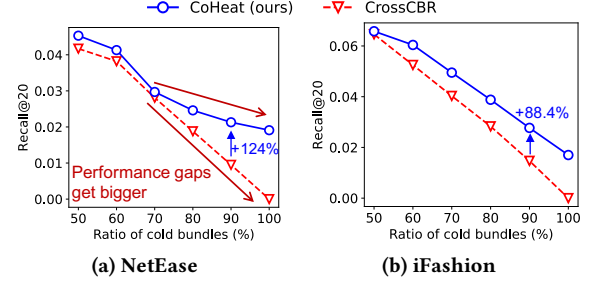
**Hyperparameters.** The hyperparameter setting is described in Appendix B.

#### 4.2 Comparison with Cold-start Methods (Q1)

In Table 2, we compare CoHEAT with baseline cold-start methods. The results show that CoHEAT consistently surpasses the baselines across all datasets and settings, verifying its superiority. Notably, CoHEAT achieves 193% higher nDCG@20 compared to CCFCRec, the best competitor, on the iFashion dataset in the all-bundle scenario. The superiority of CoHEAT over other cold-start methods stems from the following two key aspects. First, CoHEAT adeptly harnesses collaborative information of each affiliated item in a bundle through the user-item view. This approach diverges from existing cold-start methods, which fall short in utilizing user-item interactions for bundle affiliations. Second, CoHEAT explicitly addresses the pronounced skewness in user-bundle interactions through the proposed popularity-based coalescence. The results reveal the importance of tackling the inherent biases in distributions with extreme skewness, such as user-bundle interactions.

#### 4.3 Comparison with Warm-start Methods (Q2)

Table 3 compares CoHEAT with baseline warm-start methods in the warm-start scenario. Even though CoHEAT is primarily designed for cold-start bundle recommendation, it surpasses all the baselines in the warm-start scenario as well. This indicates CoHEAT

**Figure 5: Performance comparison by cold bundle ratio.**

effectively learns representations from both user-bundle and user-item views by dynamically adjusting the weights of two views in training. For the baselines, the performance improves when contrastive learning is used as exemplified in SGL, SimGCL, LightGCL, and CrossCBR. Additionally, graph-based models such as LightGCN, SGL, SimGCL, LightGCL, BGCN, and CrossCBR typically excel over other non-graph-based models. In light of these observations, CoHEAT strategically exploits a graph-based modeling approach and harnesses the power of contrastive learning. This makes CoHEAT robustly achieve the highest performance across diverse scenarios.

#### 4.4 Comparison with Cold Bundle Ratio (Q3)

In Figure 5, we compare the performance between CoHEAT and CrossCBR on NetEase and iFashion datasets under varying cold bundle ratios in test datasets. We focus on investigating the performance disparity as conditions become increasingly colder, despite their analogous performance in warm settings, as shown in Table 3. The figure reveals a pronounced performance disparity between CoHEAT and CrossCBR, intensifying as the cold bundle ratios increase. Remarkably, CrossCBR’s performance plummets to zero in entirely cold conditions while CoHEAT maintains a more stable trajectory. This divergence is particularly accentuated in NetEase due to its sparser interactions and larger bundle size. The superiority of CoHEAT over CrossCBR is rooted in strategically reducing biases inherent in sparse interactions by adopting the popularity-based coalescence. Furthermore, CoHEAT enhances the learning of user-item view by exploiting curriculum heating, thereby utilizing bundle affiliation information more effectively. Thus, this approach is more beneficial for larger bundle sizes.

#### 4.5 Ablation Study (Q4)

Table 4 provides an ablation study that compares CoHEAT with its four variants CoHEAT-PC, CoHEAT-CH-Ant, CoHEAT-CH-Fix,

**Table 4: Ablation study of CoHEAT in cold-start scenario which is our main target.**

Model	Youshu		NetEase		iFashion	
	Recall @20	nDCG @20	Recall @20	nDCG @20	Recall @20	nDCG @20
CoHEAT-PC	.0000	.0000	.0000	.0000	.0000	.0000
CoHEAT-CH-Ant	.0177	.0087	.0176	.0087	.0164	.0093
CoHEAT-CH-Fix	.0180	.0092	.0182	.0090	.0164	.0092
CoHEAT-AU	.0069	.0031	.0029	.0013	.0013	.0005
<b>CoHEAT (ours)</b>	<b>.0183</b>	<b>.0105</b>	<b>.0191</b>	<b>.0093</b>	<b>.0170</b>	<b>.0096</b>

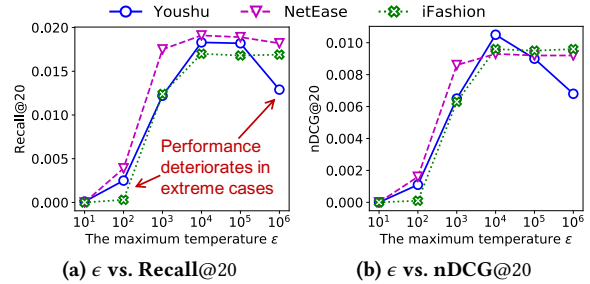
and CoHEAT-AU. This study is conducted in the cold-start scenario, which is the primary focus of our work. In CoHEAT-PC, we remove the influence of popularity-based coalescence by setting the value of  $\gamma_b^{(t)}$  in Equation (6) to a constant 0.5. For CoHEAT-CH-Ant, we exploit an anti-curriculum learning strategy. The temperature in Equation (7) is defined as  $t : T \rightarrow 0$ , initiating the learning process with the user-item view and gradually shifting the focus to the user-bundle view. For CoHEAT-CH-Fix, we remove the effect of curriculum learning by setting the value of the  $\psi^{(t)}$  in Equation (7) to the fixed maximum temperature  $\epsilon$  regardless of epochs. For CoHEAT-AU, we omit  $\mathcal{L}_{AU}$  from Equation (13), thereby excluding the contrastive learning between the two views. As shown in the table, CoHEAT consistently outperforms all the variants, which verifies all the main ideas help improve the performance. In particular, CoHEAT-PC shows a severe performance drop, justifying the importance of satisfying Properties 1 and 2 when addressing the extreme skewness inherent in cold-start bundle recommendation.

#### 4.6 Effect of the Maximum Temperature (Q5)

The maximum temperature  $\epsilon$  in Equation (7) is the most influential hyperparameter of CoHEAT since it directly affects both popularity-based coalescence and curriculum heating. Accordingly, we analyze the influence of  $\epsilon$  in cold-start scenario on real-world datasets, as depicted in Figure 6. As shown in the figure, CoHEAT shows low performance for the extreme low temperature because the representations of user-item view are not sufficiently learned. For the extreme high temperature, the performance degrades because the speed of the curriculum is too fast to fully learn the representation of the two views. As a result, we set  $\epsilon$  to  $10^4$  for all datasets since it shows the best performance.

## 5 RELATED WORKS

**Bundle recommendation.** In the field of bundle recommendations, extensive research has covered various tasks including accurately recommending existing bundles [6, 40], creating personalized bundles [1, 16, 25, 28, 56], facilitating bundle recommendations through conversation [27], and enhancing the diversity of recommended bundles [29]. Our work focuses on the cold-start problem in bundle recommendation. Previous works can be categorized based on their modeling structures: matrix factorization-based models [4, 6, 12, 24, 40] and graph learning-based models [8, 9, 17, 35, 41, 57, 64, 68, 69]. Such methods operate under the assumption that all bundles have historical interactions, which makes them ill-suited for tackling the cold-start problem. However, in real-world scenarios, new bundles are introduced daily, leading

**Figure 6: Effect of the maximum temperature  $\epsilon$ .**

to an inherent cold-start challenge. Our work addresses this significant yet overlooked issue, recognizing its potential impact on the field.

**Cold-start recommendation.** The cold-start problem, a long-standing challenge in recommender systems, focuses on recommending cold-start items with which users have not yet interacted. Existing works are primarily categorized into content-based methods [53, 67], generative methods [7, 11, 46, 48, 70], dropout-based methods [18, 45, 51], meta-learning methods [38, 50], and constraint-based methods [2, 58, 72, 73]. However, these methods are designed for item recommendation where contents are often represented as independent entities such as bag-of-words vectors, texts, or images. Moreover, such prior works have not explicitly addressed the highly skewed distribution of interactions, a critical aspect in bundle recommendation. Thus, our work excels over these methods in cold-start bundle recommendation by effectively harnessing intricate bundle affiliations from the user-item view and addressing the skewed distribution during training.

## 6 CONCLUSION

We propose CoHEAT, an accurate method for cold-start bundle recommendation. CoHEAT strategically leverages user-bundle and user-item views to handle the extremely skewed distribution of bundle interactions. By emphasizing the user-item view for less popular bundles, CoHEAT effectively captures richer information than the often sparse user-bundle view. The incorporation of curriculum learning further enhances the learning process, starting with the simpler user-bundle view embeddings and gradually transitioning to the more intricate user-item view embeddings. In addition, the contrastive learning of CoHEAT bolsters the learning of representations from the two views facilitating effective knowledge transfer from the richer to the sparser view. Extensive experiments show that CoHEAT provides the state-of-the-art performance in cold-start bundle recommendation, achieving up to 193% higher nDCG@20 compared to the best competitor.

## ACKNOWLEDGMENTS

This work was supported by Jung-Hun Foundation. This work was also supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) [NO.2021-0-01343, Artificial Intelligence Graduate School Program (Seoul National University)] and [NO.2021-0-02068, Artificial Intelligence Innovation Hub (Artificial Intelligence Institute, Seoul National University)]. The Institute of Engineering Research and ICT at Seoul National University provided research facilities for this work. U Kang is the corresponding author.



## REFERENCES

- [1] Jinze Bai, Chang Zhou, Junshuai Song, Xiaoru Qu, Weiting An, Zhao Li, and Jun Gao. 2019. Personalized Bundle List Recommendation. In *WWW*.
- [2] 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*.
- [3] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *ICML*.
- [4] Tzoof Avny Brosh, Amit Livne, Oren Sar Shalom, Bracha Shapira, and Mark Last. 2022. BRUCE: Bundle Recommendation Using Contextualized item Embeddings. In *RecSys*.
- [5] Xuheng Cai, Chao Huang, Lianghao Xia, and Xubin Ren. 2023. LightGCL: Simple Yet Effective Graph Contrastive Learning for Recommendation. In *ICLR*.
- [6] Da Cao, Liqiang Nie, Xiangnan He, Xiaochi Wei, Shunzhi Zhu, and Tat-Seng Chua. 2017. Embedding Factorization Models for Jointly Recommending Items and User Generated Lists. In *SIGIR*.
- [7] Dong-Kyu Chae, Jin-Soo Kang, Sang-Wook Kim, and Jaeho Choi. 2019. Rating Augmentation with Generative Adversarial Networks towards Accurate Collaborative Filtering. In *WWW*.
- [8] Jianxin Chang, Chen Gao, Xiangnan He, Depeng Jin, and Yong Li. 2020. Bundle Recommendation with Graph Convolutional Networks. In *SIGIR*.
- [9] Jianxin Chang, Chen Gao, Xiangnan He, Depeng Jin, and Yong Li. 2023. Bundle Recommendation and Generation With Graph Neural Networks. *IEEE Trans. Knowl. Data Eng.* (2023).
- [10] Hong Chen, Yudong Chen, Xin Wang, Ruobing Xie, Rui Wang, Feng Xia, and Wenwu Zhu. 2021. Curriculum Disentangled Recommendation with Noisy Multi-feedback. In *NeurIPS*.
- [11] Hao Chen, Zefan Wang, Feiran Huang, Xiao Huang, Yue Xu, Yishi Lin, Peng He, and Zhoujun Li. 2022. Generative Adversarial Framework for Cold-Start Item Recommendation. In *SIGIR*.
- [12] Liang Chen, Yang Liu, Xiangnan He, Lianli Gao, and Zibin Zheng. 2019. Matching User with Item Set: Collaborative Bundle Recommendation with Deep Attention Network. In *IJCAI*.
- [13] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A Simple Framework for Contrastive Learning of Visual Representations. In *ICML*.
- [14] Wen Chen, Pipei Huang, Jiaming Xu, Xin Guo, Cheng Guo, Fei Sun, Chao Li, Andreas Pfadler, Huan Zhao, and Binqiang Zhao. 2019. POG: Personalized Outfit Generation for Fashion Recommendation at Alibaba iFashion. In *KDD*.
- [15] Yudong Chen, Xin Wang, Miao Fan, Jizhou Huang, Shengwen Yang, and Wenwu Zhu. 2021. Curriculum Meta-Learning for Next POI Recommendation. In *KDD*.
- [16] Qilin Deng, Kai Wang, Minghao Zhao, Runze Wu, Yu Ding, Zhene Zou, Yue Shang, Jianrong Tao, and Changjie Fan. 2021. Build Your Own Bundle - A Neural Combinatorial Optimization Method. In *MM*.
- [17] Qilin Deng, Kai Wang, Minghao Zhao, Zhene Zou, Runze Wu, Jianrong Tao, Changjie Fan, and Liang Chen. 2020. Personalized Bundle Recommendation in Online Games. In *CIKM*.
- [18] 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 *MM*.
- [19] Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *EMNLP*.
- [20] Xue Geng, Hanwang Zhang, Jingwen Bian, and Tat-Seng Chua. 2015. Learning Image and User Features for Recommendation in Social Networks. In *ICCV*.
- [21] Yantao Gong, Cao Liu, Jiazhen Yuan, Fan Yang, Xunliang Cai, Guanglu Wan, Jiansong Chen, Ruiyao Niu, and Houfeng Wang. 2021. Density-Based Dynamic Curriculum Learning for Intent Detection. In *CIKM*.
- [22] Xiangnan He and Tat-Seng Chua. 2017. Neural Factorization Machines for Sparse Predictive Analytics. In *SIGIR*.
- [23] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yong-Dong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. In *SIGIR*.
- [24] Yun He, Jianling Wang, Wei Niu, and James Caverlee. 2019. A Hierarchical Self-Attentive Model for Recommending User-Generated Item Lists. In *CIKM*.
- [25] Yun He, Yin Zhang, Weiwei Liu, and James Caverlee. 2020. Consistency-Aware Recommendation for User-Generated Item List Continuation. In *WSDM*.
- [26] Zhenghua He, Chaochen Gu, Rui Xu, and Kaijie Wu. 2020. Automatic Curriculum Generation by Hierarchical Reinforcement Learning. In *ICONIP*.
- [27] Zhankui He, Handong Zhao, Tong Yu, Sungchul Kim, Fan Du, and Julian J. McAuley. 2022. Bundle MCR: Towards Conversational Bundle Recommendation. In *RecSys*.
- [28] Hyunsik Jeon, Jun-Gi Jang, Taehun Kim, and U Kang. 2023. Accurate bundle matching and generation via multitask learning with partially shared parameters. *Plos one* (2023).
- [29] Hyunsik Jeon, Jongjin Kim, Jaeri Lee, Jong-eun Lee, and U Kang. 2023. Aggregately Diversified Bundle Recommendation via Popularity Debiasing and Configuration-Aware Reranking. In *PAKDD*.
- [30] Hyunsik Jeon, Jongjin Kim, Hoyoung Yoon, Jaeri Lee, and U Kang. 2022. Accurate Action Recommendation for Smart Home via Two-Level Encoders and Commonsense Knowledge. In *CIKM*.
- [31] Hyunsik Jeon, Bonhun Koo, and U Kang. 2019. Data Context Adaptation for Accurate Recommendation with Additional Information. In *BigData*.
- [32] Jongjin Kim, Hyunsik Jeon, Jaeri Lee, and U Kang. 2023. Diversely Regularized Matrix Factorization for Accurate and Aggregately Diversified Recommendation. In *PAKDD*.
- [33] Bonhun Koo, Hyunsik Jeon, and U Kang. 2020. Accurate News Recommendation Coalescing Personal and Global Temporal Preferences. In *PAKDD*.
- [34] Bonhun Koo, Hyunsik Jeon, and U Kang. 2021. PGT: news recommendation coalescing personal and global temporal preferences. *Knowl. Inf. Syst.* (2021).
- [35] Yunshan Ma, Yingzhi He, An Zhang, Xiang Wang, and Tat-Seng Chua. 2022. CrossCBR: Cross-view Contrastive Learning for Bundle Recommendation. In *KDD*.
- [36] Binyamin Manela and Armin Biess. 2022. Curriculum learning with Hindsight Experience Replay for sequential object manipulation tasks. *Neural Networks* (2022).
- [37] Ruihui Mu. 2018. A Survey of Recommender Systems Based on Deep Learning. *IEEE Access* (2018).
- [38] Feiyang Pan, Shuokai Li, Xiang Ao, Pingzhong Tang, and Qing He. 2019. Warm Up Cold-start Advertisements: Improving CTR Predictions via Learning to Learn ID Embeddings. In *SIGIR*.
- [39] Haekyu Park, Jinhong Jung, and U Kang. 2017. A comparative study of matrix factorization and random walk with restart in recommender systems. In *BigData*.
- [40] Apurva Pathak, Kshitiz Gupta, and Julian J. McAuley. 2017. Generating and Personalizing Bundle Recommendations on Steam. In *SIGIR*.
- [41] Yuyang Ren, Haonan Zhang, Luoyi Fu, Xinbing Wang, and Chenghu Zhou. 2023. Distillation-Enhanced Graph Masked Autoencoders for Bundle Recommendation. In *SIGIR*.
- [42] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *UAI*.
- [43] Joshua David Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. 2021. Contrastive Learning with Hard Negative Samples. In *ICLR*.
- [44] Andrew I. Schein, Alexandrin Popescul, Lyle H. Ungar, and David M. Pennock. 2002. Methods and metrics for cold-start recommendations. In *SIGIR*.
- [45] 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*.
- [46] Changfeng Sun, Han Liu, Meng Liu, Zhaochun Ren, Tian Gan, and Liqiang Nie. 2020. LARA: Attribute-to-feature Adversarial Learning for New-item Recommendation. In *WSDM*.
- [47] Zhu Sun, Jie Yang, Kaidong Feng, Hui Fang, Xinghua Qu, and Yew Soon Ong. 2022. Revisiting Bundle Recommendation: Datasets, Tasks, Challenges and Opportunities for Intent-aware Product Bundling. In *SIGIR*.
- [48] Aäron van den Oord, Sander Dieleman, and Benjamin Schrauwen. 2013. Deep content-based music recommendation. In *NIPS*.
- [49] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation Learning with Contrastive Predictive Coding. *CoRR* (2018).
- [50] Manasi Vartak, Arvind Thiagarajan, Conrado Miranda, Jeshua Bratman, and Hugo Larochelle. 2017. A Meta-Learning Perspective on Cold-Start Recommendations for Items. In *NIPS*.
- [51] Maksims Volkovs, Guang Wei Yu, and Tomi Poutanen. 2017. DropoutNet: Addressing Cold Start in Recommender Systems. In *NIPS*.
- [52] Chenyang Wang, Yuanqing Yu, Weizhi Ma, Min Zhang, Chong Chen, Yiqun Liu, and Shaoping Ma. 2022. Towards Representation Alignment and Uniformity in Collaborative Filtering. In *KDD*.
- [53] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. 2015. Collaborative Deep Learning for Recommender Systems. In *KDD*.
- [54] Tongzhou Wang and Phillip Isola. 2020. Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere. In *ICML*.
- [55] Xin Wang, Yudong Chen, and Wenwu Zhu. 2022. A Survey on Curriculum Learning. *IEEE Trans. Pattern Anal. Mach. Intell.* (2022).
- [56] Penghui Wei, Shaoguo Liu, Xuanhua Yang, Liang Wang, and Bo Zheng. 2022. Towards Personalized Bundle Creative Generation with Contrastive Non-Autoregressive Decoding. In *SIGIR*.
- [57] Yinwei Wei, Xiaohao Liu, Yunshan Ma, Xiang Wang, Liqiang Nie, and Tat-Seng Chua. 2023. Strategy-aware Bundle Recommender System. In *SIGIR*.
- [58] Yinwei Wei, Xiang Wang, Qi Li, Liqiang Nie, Yan Li, Xuanping Li, and Tat-Seng Chua. 2021. Contrastive Learning for Cold-Start Recommendation. In *MM*.
- [59] Jiancan Wu, Xiang Wang, Fuli Feng, Xiangnan He, Liang Chen, Jianxun Lian, and Xing Xie. 2021. Self-supervised Graph Learning for Recommendation. In *SIGIR*.
- [60] Jun Xia, Lirong Wu, Jintao Chen, Bozhen Hu, and Stan Z. Li. 2022. SimGRACE: A Simple Framework for Graph Contrastive Learning without Data Augmentation. In *WWW*.
- [61] Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. 2021. ConSERT: A Contrastive Framework for Self-Supervised Sentence Representation Transfer. In *ACL*.

- [62] Junliang Yu, Hongzhi Yin, Xin Xia, Tong Chen, Lizhen Cui, and Quoc Viet Hung Nguyen. 2022. Are Graph Augmentations Necessary?: Simple Graph Contrastive Learning for Recommendation. In *SIGIR*.
- [63] Qing Yu, Daiki Ikami, Go Irie, and Kiyoharu Aizawa. 2020. Multi-task Curriculum Framework for Open-Set Semi-supervised Learning. In *ECCV*.
- [64] Zhouxin Yu, Jintang Li, Liang Chen, and Zibin Zheng. 2022. Unifying multi-associations through hypergraph for bundle recommendation. *Knowl. Based Syst.* (2022).
- [65] Runzhe Zhan, Xuebo Liu, Derek F. Wong, and Lidia S. Chao. 2021. Meta-Curriculum Learning for Domain Adaptation in Neural Machine Translation. In *AAAI*.
- [66] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. 2021. FlexMatch: Boosting Semi-Supervised Learning with Curriculum Pseudo Labeling. In *NeurIPS*.
- [67] Fuzheng Zhang, Nicholas Jing Yuan, Defu Lian, Xing Xie, and Wei-Ying Ma. 2016. Collaborative Knowledge Base Embedding for Recommender Systems. In *KDD*.
- [68] Zhenning Zhang, Boxin Du, and Hanghang Tong. 2022. SuGeR: A Subgraph-based Graph Convolutional Network Method for Bundle Recommendation. In *CIKM*.
- [69] Sen Zhao, Wei Wei, Ding Zou, and Xianling Mao. 2022. Multi-view intent disentangle graph networks for bundle recommendation. In *AAAI*.
- [70] Xu Zhao, Yi Ren, Ying Du, Shenzheng Zhang, and Nian Wang. 2022. Improving Item Cold-start Recommendation via Model-agnostic Conditional Variational Autoencoder. In *SIGIR*.
- [71] Chang Zhou, Jianxin Ma, Jianwei Zhang, Jingren Zhou, and Hongxia Yang. 2021. Contrastive Learning for Debaised Candidate Generation in Large-Scale Recommender Systems. In *KDD*.
- [72] Zhihui Zhou, Lilin Zhang, and Ning Yang. 2023. Contrastive Collaborative Filtering for Cold-Start Item Recommendation. In *The Web Conference*.
- [73] 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*.

## A COMPLEXITY ANALYSIS

The space and time complexity of CoHEAT is as follows.

**Space complexity.** The space complexity of CoHEAT is  $O((U + B + I)d)$  where  $U$ ,  $B$ , and  $I$  are respectively the numbers of users, bundles, and items, and  $d$  is the dimensionality of embeddings.

**Time complexity.** The time complexity is evaluated twofold: graph learning and contrastive learning. We notice that the time complexities of popularity-based coalescence and curriculum heating are neglectable. The time complexity of the graph learning is  $O(|E|Kds\frac{|E|}{T})$ , where  $|E|$  is the number of edges,  $K$  is the number of layers,  $d$  is the dimensionality of embeddings,  $s$  is the number of epochs, and  $T$  is the batch size. The time complexity of the contrastive learning is  $O(2d(T + 2T^2)s\frac{|E|}{T})$ .

## B HYPERPARAMETER SETTINGS

We utilize the baselines with their official implementations and use their reported best hyperparameters. We implement CoHEAT with PyTorch. We set the dimensionality  $d$  of node embeddings as 64. The other hyperparameters are grid-searched: the learning rate in  $\{0.001, 0.0001, 0.00001\}$ ,  $\lambda_1$  in  $\{0.1, 0.2, 0.5, 1.0\}$ ,  $\lambda_2$  in  $\{0.00004, 0.0001, 0.0004, 0.001\}$ ,  $K$  in  $\{1, 2\}$ , and the maximum temperature  $\epsilon$  in  $\{10^1, 10^2, 10^3, 10^4, 10^5, 10^6\}$ .