BRIDGE
Bundle Recommendation via Instruction-Driven Generation

Bui, Tuan Nghia; Nguyen, Huy Son; Nguyen, Cam Van Thi; Le, Hoang Quynh; Le, Duc Trong

DOI
10.3233/FAIA251063

Licence
CC BY-NC

Publication date
2025

Document Version
Final published version

Published in
ECAI 2025 - 28th European Conference on Artificial Intelligence, including 14th Conference on Prestigious Applications of Intelligent Systems, PAIS 2025 - Proceedings

Citation (APA)
Bui, T. N., Nguyen, H. S., Nguyen, C. V. T., Le, H. Q., & Le, D. T. (2025). BRIDGE: Bundle Recommendation via Instruction-Driven Generation. In I. Lynce, N. Murano, M. Vallati, S. Villata, F. Chesani, M. Milano, A. Omicini, & M. Dastani (Eds.), *ECAI 2025 - 28th European Conference on Artificial Intelligence, including 14th Conference on Prestigious Applications of Intelligent Systems, PAIS 2025 - Proceedings* (pp. 2219-2226). (Frontiers in Artificial Intelligence and Applications; Vol. 413). IOS Press. https://doi.org/10.3233/FAIA251063

ECAI 2025
I. Lynce et al. (Eds.)

doi:10.3233/FAIA251063

BRIDGE: Bundle Recommendation via Instruction-Driven Generation

Tuan-Nghia Bui[a,1], Huy-Son Nguyen[b,1], Cam-Van Thi Nguyen[a], Hoang-Quynh Le[a] and Duc-Trong Le [a,*]

[a]VNU University of Engineering and Technology, Hanoi, Vietnam
[b]Delft University of Technology, Netherlands

Abstract. Bundle recommendation aims to suggest a set of interconnected items to users. However, diverse interaction types and sparse interaction matrices often pose challenges for previous approaches in accurately predicting user-bundle adoptions. Inspired by the distant supervision strategy and generative paradigm, we propose BRIDGE, a novel framework for bundle recommendation. It consists of two main components, namely the item-sensitive instruction generation and the pseudo bundle generation modules. Inspired by the distant supervision approach, the former is to generate more auxiliary information, e.g., sampled item-sensitive instruction, for training without using external data. This information is subsequently aggregated with collaborative signals from user historical interactions to create pseudo 'ideal' bundles. This capability allows BRIDGE to explore all aspects of bundles, rather than being limited to existing real-world bundles. It effectively bridging the gap between user imagination and predefined bundles, hence improving the bundle recommendation performance. Experimental results and analyses validate the superiority of BRIDGE over state-of-the-art methods across four benchmark datasets. Our implementation is available at `https://github.com/Rec4Fun/BRIDGE`.

1 Introduction

Beyond item-level recommendation, bundle recommendation captures a more nuanced understanding of user behavior in recommending a set of cohesive items for an exclusive intention. The understanding is often built upon leveraging historical user-item, user-bundle interactions, and bundle-item affiliations to learn user preferences [30, 16]. This task has gained significant attention in recent years due to its complexity. Exploring user preferences in recommendation systems, especially in the context of bundle recommendation, is a critical and complex challenge. It plays a vital role in improving user experiences across diverse domains [6]. Most effective graph-based models [3, 15, 32] employ Bayesian Personalized Ranking (BPR) [24] as the primary objective. These models differentiate between unseen user-bundle interactions (negative samples) and observed user-bundle interactions (positive ones) to rank recommendations by classifying true negatives. Regardless of the success of ranking-based methods across various recommendation domains, there exist several serious limitations, which call into question their ability to fully capture and respond to the complexities of user behavior. As highlighted by [31], one of the primary concerns is the

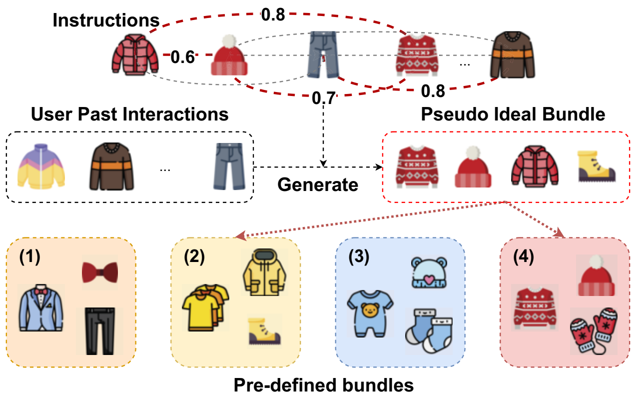


Figure 1. An example of item-sensitive instruction-driven bundle recommendation.

oversimplification of human behavior in adopting bundles of items. On the other hand, these methods often lack robustness in the face of data sparsity and noise. In diverse real-world scenarios, user-item interaction data is incomplete, noisy, and sparse, which poses urgent challenges for ranking models in deriving reliable inferences. This issue is even more pronounced in the context of bundle recommendations, where the user-bundle interaction data tends to be even sparser and less consistent. The complexity of bundles, which involve multiple items rather than single ones, adds another layer of difficulty in accurately modeling user preferences. When confronted with such limited data, ranking models are prone to over-fitting, relying too heavily on the small amount of available information.

Inspired by the distant supervision strategy [5, 17], we seek to leverage auxiliary resources to generate silver-standard labeled data for model training. However, the effectiveness of distant supervision in bundle recommendation remains largely unexplored due to the scarcity of relevant knowledge bases. Bundle recommendation datasets typically consist of user, bundle, and item IDs, but the lack of information to identify common objects across different datasets. Therefore, it is more challenging to leverage multiple external data sources to enhance recommendation performance.

To address these challenges, we propose a novel framework named **BRIDGE** for **B**undle **R**ecommendation using **I**nstruction-**D**riven **GE**neration, which is inspired by the strategy of distant supervision [5, 17] and generative retrieval [31, 23]. Unlike traditional distant supervision methods that rely on external data sources, BRIDGE generates silver-standard data through determining sampled rough

* Corresponding author: trongld@vnu.edu.vn
[1] Shared first-authors.

item set of high-similarity items from user-item associations using the *Item-Sensitive Instruction Generation* module. Leveraging historical interacted items of a given user, these sets are jointly exploited as ‘instructions’ to produce the pseudo ‘ideal’ bundle that aligns with the user preference via the *Pseudo Bundle Generation* module. The pseudo bundle helps retrieve relevant bundles from predefined options for personalized recommendation within the *Retrieval & Ranking* module. Figure 1 illustrates an example how BRIDGE works.

To summarize, our main contributions are as follows:

- We enhance the training process for the bundle recommendation task with auxiliary instructions, i.e, item-sensitive instruction, using a distant supervision-based approach without using any external data sources.
- We propose BRIDGE, which uses instructional guidance from historical user interactions and item sampled sets to generate pseudo 'ideal' bundles to discover relevant candidates from existing bundles for recommendation. To the best of our knowledge, we are the first to present an end-to-end generative approach for bundle recommendation.
- We conduct extensive experiments on four publicly-available datasets and achieve significant improvements over all baseline methods on various metrics.

The remainder of the paper is organized as follows: Related studies for the bundle recommendation task are literally discussed in the section 2, which is followed by the methodology section to thoroughly described the proposed model BRIDGE. We investigate its effectiveness in the section 4 before summarizing findings in the final section.

2 Related Work

Bundle Recommendation. Research on bundle recommendation typically focuses on three main approaches: *factorization methods* decompose interaction matrices into latent factors to predict and enhance bundle recommendations; *graph-based methods* utilize graphs to capture complex relationships between users, items, and bundles, refining recommendations through graph-based techniques; and *generative methods* employ generative models to create pseudo ideal bundles from historical interactions, addressing limitations of traditional ranking approaches by exploring new configurations aligned with user preferences. Unlike *next-basket recommendation* [27, 13], this approach often neglects temporal orders and exploits the whole history at once [26].

Factorization Methods. Early bundle recommendation methods, based on the BPR framework [24], use user-bundle interactions as positive pairs and sample negative pairs from unobserved interactions. DAM [4] is a multi-task framework that recommends both items, bundles using an attention mechanism and shared weights to capture user preferences at both levels.

Graph-based Recommendation. BGCN [3] integrates Graph Convolutional Networks with multi-view learning to exploit various interactions. Addressing the multi-view preference of users, CrossCBR [15], MultiCBR [16] adopt InfoNCE [8] to align multi-view user preferences, while MIDGN [32] models multiple intention hidden in users/bundles. BundleGT [30] explores the strategy-aware ability of user/bundle representations. BunCa [19] proposes a hypothesis of the asymmetric relationship between items to enhance the modeling process. CoHeat [11] not only improve the cold-start, but also general performance in bundle recommendation via popularity and curriculum heating.

Generative Recommendation. In traditional item recommendation, PURE [33] uses GANs to generate fake user and item embeddings, covering diverse feature space corners. LARA [25] applies multiple generators on item attributes to create pseudo user profiles, with a discriminator classifying real user-item pairs. DreamRec [31] employs a diffusion process to reconstruct item embeddings and retrieve recommendations based on similarity to these oracles. TIGER [23] uses Transformer-based architecture to generate item aspects matching user interests, enhancing item representations with auxiliary data. A recent diffusion approach (DisCo) strives to produce a new bundle in distribution space for each user to overcome the limitation of cold-start bundles, fostered through disentangled features of users [1]. Compared to these works, BRIDGE directly guide the generative process using real items in pseudo bundle, instead of using latent-feature representation.

Distant Supervision. Distant supervision is an efficient training strategy applied across various problems and domains, including relation extraction [17, 22], procedural activities recognition [14], and image captioning [21]. In the context of recommendation systems, distant supervision has been successfully applied for cross-domain recommendation [7, 2].

3 BRIDGE - Bundle Recommendation via Instruction-Driven Generation

The overall architecture of BRIDGE, shown in Fig. 2, consists of three main components: *item-sensitive instruction generation*, *pseudo bundle generation* and *retrieval & ranking modules*. Each component is thoroughly described in the following subsections.

3.1 Problem Formulation.

Given sets of users $\mathcal{U} = \{u_1, u_2, \ldots, u_{|\mathcal{U}|}\}$, bundles $\mathcal{B} = \{b_1, b_2, \ldots, b_{|\mathcal{B}|}\}$, and items $\mathcal{V} = \{v_1, v_2, \ldots, v_{|\mathcal{V}|}\}$, the observed user-bundle, bundle-item and user-item interactions are respectively represented as three binary matrices $X \in \{0,1\}^{|\mathcal{U}|\times|\mathcal{B}|}$, $Y \in \{0,1\}^{|\mathcal{B}|\times|\mathcal{V}|}$ and $Z \in \{0,1\}^{|\mathcal{U}|\times|\mathcal{V}|}$, where cells with value of 1 if there exists links between user-bundle, bundle-item or user-item pairs, and 0 otherwise. For a given user $u \in \mathcal{U}$ and a predefined bundle $b \in \mathcal{B}$, we seek to compute the probability score $y_{u,b}$ that the user u will adopt the bundle b as:

$$y_{u,b} = g(b, f_\theta(u, X, Y, Z)) \tag{1}$$

where g is a similarity function, and $f_\theta(u, X, Y, Z)$ is the pseudo bundle generation function with a learnable parameter θ, which manifests the preferential representation of u. Specifically, our goal is to maximize the similarity between the target bundle and the generated bundle. The top-K bundles with the highest similarity values will be recommended to user u.

3.2 Item-Sensitive Instruction Generation

We assume that two items interacted by a common set of users via Z may indicate a potential combination within a pseudo bundle, suggesting that they are closely related in the representation space. This pseudo bundle can serve as a instructive signal to guide the bundle generator to create a meaningful bundle. Inherited from the inspection of item relations via co-purchase interaction [13, 18], we compute the item co-occurrence matrix $C = Z^T \cdot Z, C \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$. Next, we establish an item homogeneous graph $\mathcal{G} = \{\mathcal{V}, E\}$, where

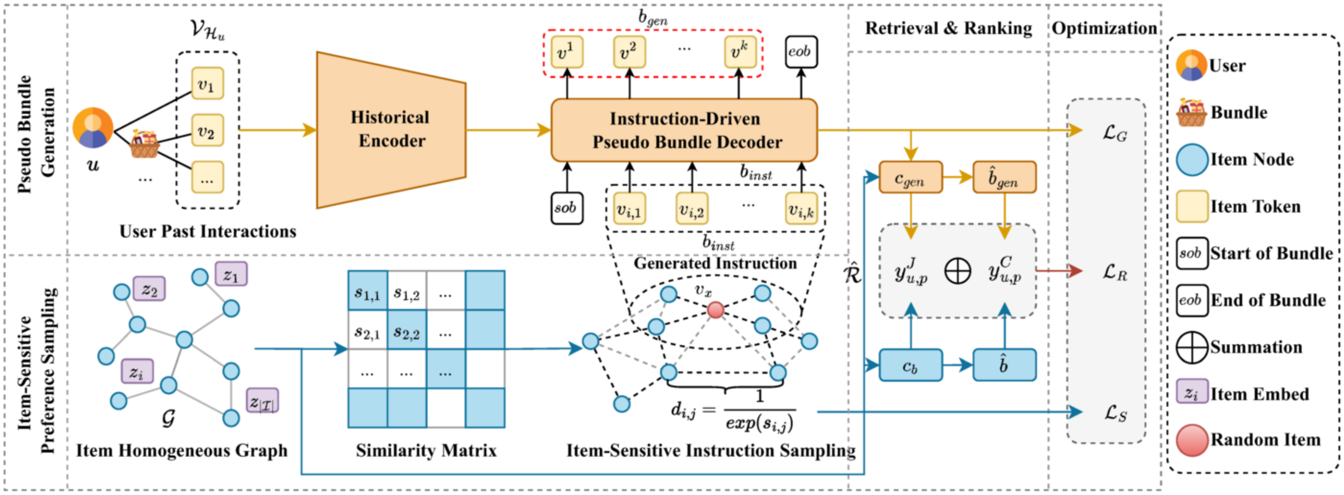


Figure 2. The end-to-end architecture of BRIDGE

$\mathcal{V}$ is the set of nodes and $E = \{e_{i,j} \mid v_i, v_j \in \mathcal{V}\}$ denotes the edge set, $e_{i,j} = 1$ if $C(i,j) > 0$, and 0 otherwise.

In order to learn the item latent embedding of a given item v_i using topological information, we employ a graph convolution network, i.e., LightGCN[10], on the item homogeneous graph $\mathcal{G}$. Let us denote $r_i^{(l)}$ is the item latent representation of item v_i at the $l-$th layer. It is derived as:

$$r_i^{(l)} = \sum_{j \in \mathcal{M}_i} \frac{1}{\sqrt{|\mathcal{M}_i|}\sqrt{|\mathcal{M}_j|}} r_j^{(l-1)}, \tag{2}$$

where $r_i^{(0)} \in \mathbb{R}^d$ is randomly initialized, $\mathcal{M}_i$ represent the neighbor set of item i in $\mathcal{G}$. The final item latent representation $\hat{r}_i$ of item i is inferred as follows:

$$r_i^* = \frac{1}{L+1} \sum_{l=0}^{L} r_i^{(l)}, \quad \hat{r}_i = \frac{r_i^*}{||r_i^*||_2^2} \tag{3}$$

where L is the number of propagation layers in the GCN, and the final representation of item i is obtained from r_i^* followed by a second order Euclidean normalization. Inspired by [13], we calculate the relevant score $s_{i,j}$ of an item pair $(i,j), i,j \in \mathcal{V}$ as:

$$s_{i,j} = \hat{r}_i^{\intercal} \cdot \hat{r}_j \tag{4}$$

Given an item v_i, the instructive item set $\mathcal{C}_i$ of k nearest neighbors is determined using a distance function for each pair of items (v_i, v_j) as follows:

$$d_{i,j} = \frac{1}{exp(s_{i,j})}, \tag{5}$$

To facilitate the bundle recommendation task, we aim to construct a pseudo bundle, which expresses the preferential intention of a given user u. Inspired by [31], it might be inferred via behavioral history $\mathcal{H}_u$, i.e., interacted items $\mathcal{V}_{\mathcal{H}_u} \in \mathcal{V}$. Using sets of those items built from the previous section, we suppose that they may create meaningful instructions to better construct the pseudo bundle. Given $\mathcal{B}_u \in \mathcal{B}$ as the bundle set associated with user u, $\mathcal{V}_u, \mathcal{V}_{\mathcal{B}_u} \in \mathcal{V}$ are the sets of adopted items of user u extracted from $\mathcal{Z}$, and $\mathcal{B}_u$, we have:

$$\mathcal{V}_{\mathcal{H}_u} := \mathcal{V}_u \cup \mathcal{V}_{\mathcal{B}u} \tag{6}$$

Item-Sensitive Instruction Sampling. For each training iteration of a user u, we randomly select an item v_i from his historical interactions $\mathcal{V}_{\mathcal{H}_u}$. Using sampled item set $\mathcal{C}_i$ of k nearest neighbors of v_i determined from the item-sensitive instruction generation, an instructive bundle b_{inst} is constructed via:

$$b_{inst} = \{\bar{v}_1, \bar{v}_2, \bar{v}_3, \ldots, \bar{v}_k\}, \tag{7}$$

where it is noted that $1 < k < n$, $n = |\mathcal{V}_{\mathcal{H}_u}|$, k is randomly selected during training to add random noises for improving model robustness. Depending on the number of historical interactions n, there may have a set of instructive bundles $\mathcal{B}_{u,inst} = \{b_{inst}\}$ generated from multiple training iterations for each user u.

3.3 Pseudo Bundle Generation

Historical Encoder. To personalize the bundle generation process, the user's historical interactions $\mathcal{V}_{\mathcal{H}_u}$ are encoded to convey the personalized preferences of user u as:

$$q_u = \Psi_\theta(\mathcal{V}_{\mathcal{H}_u}), \tag{8}$$

q_u represents the encoded information of user u. $\Psi_\theta(\cdot)$ denotes the *Historical Encoder*, whose architecture follows Transformer [29].

Instruction-Driven Pseudo Bundle Decoder. In order to form a pseudo bundle for relevant bundle retrieval, the instructive set $\mathcal{B}_{u,inst}$ is fed into a sequence-to-sequence architecture, e.g., Transformer, to aggregate potential items that highly correlate and align with user preferences. Motivated by [31], we employ the reconstruction distribution process to reconstruct the temporal distribution of potential item probabilities. Using the collaborative signals from the user's previous interactions $\mathcal{V}_{\mathcal{H}_u}$ and a given b_{inst}, the probability of pseudo bundle b_{gen} after the $T-$th step generation is derived as:

$$p_\theta(\tilde{v}^{(1:T)}) := \prod_{t=1}^{T} p_\theta(\tilde{v}^{(t)}|\mathcal{V}_{\mathcal{H}_u}, \bar{v}_{0:t-1}), \tag{9}$$
$$p_\theta(\tilde{v}^{(t)}|\mathcal{V}_{\mathcal{H}_u}, \bar{v}_{0:t-1}) := \mathcal{N}(\Phi_\theta(q_u), \bar{v}_{0:t-1}), \beta\mathbf{I}),$$

where $\tilde{v}^{(t)}$ is the $t-$th candidate item, and $\tilde{v}^{(1:T)}$ is the candidate item set after T generation steps for b_{gen}. Likewise, $\bar{v}_{0:t-1}$ is the item set including a start-of-bundle pseudo item $[sob]$ at the first index and the first $t-1$ items of b_{ins}. $\mathcal{N}(\epsilon, \sigma^2)$ denotes the Gaussian distribution. $\mathbf{I} \in \mathbb{R}^{|\mathcal{V}|}$ is the identity tensor. $\Phi_\theta(\cdot)$ represents *Instruction-Driven Pseudo Bundle Decoder*, following the Transformer decoder architecture [29], and β is a hyper-parameter control the variance of the probability distribution. The final pseudo bundle $b_{gen} = \{\tilde{v}_1, \tilde{v}_2, \ldots, \tilde{v}_t\}$ is the t candidate items extracted from the $\tilde{v}^{(1:T)}$ set, where v_{t+1} is the end-of-bundle pseudo item $[eob]$.

During inference process, we do not utilize the instructive bundle set to neglect biases in constructing the pseudo bundle. It is merely generated based on the historical interaction of users $\mathcal{V}_{\mathcal{H}_u}$ and previously-selected candidate items via the following procedure:

$$p_\theta(\tilde{v}^{(1:T)}) := \prod_{t=1}^{T} p_\theta(v^{(t)} | \mathcal{V}_{\mathcal{H}_u}, \tilde{v}^{(0:t-1)}), \quad (10)$$
$$p_\theta(\tilde{v}^{(t)} | v_{1:n}, \tilde{v}^{(0:t-1)}) := \mathcal{N}(\Phi_\theta(q_u), \tilde{v}^{(0:t-1)}), \beta_I \mathbf{I}),$$

where the final pseudo bundle b_{gen} is built in the similar way as the training phase.

3.4 Retrieval & Ranking

With the objective of recommending existing bundles, we employ a retrieval and ranking workflow. The main idea is to discover for the top$-K$ bundles in $\mathcal{B}$ that are most similar to the pseudo bundle b_{gen}. It raises a need to calculate similarity score between the pseudo bundle b_{gen} and each bundle $b \in \mathcal{B}$ of a given user u. Generally, there are two typical similar scores used in BRIDGE including Jaccard matching score $y^J_{u,b}$ and Cosine recommendation score $y^C_{u,b}$. The former favors the exact matching among bundles while the latter emphasizes the relative one using latent preferential features.

For Jaccard similarity, let us denote $c_{gen}, c_b \in \{0,1\}^{|\mathcal{V}|}$ are binary vectors that represent the occurrence of items within the pseudo bundle b_{gen} and a bundle $b \in \mathcal{B}$, we have:

$$y^J_{u,b} = \frac{c_{gen}^\intercal \cdot c_b}{c_{gen}^\intercal \cdot \mathbf{I} + c_b^\intercal \cdot \mathbf{I} - c_{gen}^\intercal \cdot c_b}, \quad (11)$$

Equally important, the recommendation score $y^C_{u,b}$ is computed as the following procedure:

$$\hat{b} = \frac{1}{c_b^\intercal \cdot \mathbf{I}} c_b^\intercal \cdot \hat{\mathcal{R}}, \quad (12)$$
$$\hat{b}_{gen} = \frac{1}{c_{gen}^\intercal \cdot \mathbf{I}} c_{gen}^\intercal \cdot \hat{\mathcal{R}}, \quad (13)$$
$$y^C_{u,b} = \frac{\hat{b}^\intercal \cdot \hat{b}_{gen}}{||\hat{b}||_2^2 \cdot ||\hat{b}_{gen}||_2^2}, \quad (14)$$

where $\hat{\mathcal{R}} \in \mathbb{R}^{|\mathcal{V}| \times d}$ is the latent item representation obtained from the item-sensitive instruction generation module via Eq (3). Finally, we combine the two metrics to leverage all relevant bundles considering both two matching strategies:

$$y_{u,b} = \alpha y^J_{u,b} + (1-\alpha) y^C_{u,b} \quad (15)$$

where $\alpha \in [0,1]$ is a trade-off hyperparameter to control the balance between two terms. The top-K candidate bundles with the highest similarity score $y_{u,b}$ are recommended to the user u.

3.5 Optimization

Our model BRIDGE is trained with triplet losses namely item-sensitive instruction sampling loss $\mathcal{L}_S$, the pseudo bundle generation loss $\mathcal{L}_G$ and the recommendation loss $\mathcal{L}_R$. Specifically, the sampling loss is to maximize the sensitive score between potential items within bundle instructions. It is calculated as:

Dataset	*Clothing*	*Electronic*	*Food*	*Steam*
#User $\|\mathcal{U}\|$	965	888	879	$29,634$
#Item $\|\mathcal{I}\|$	$4,487$	$3,499$	$3,767$	$2,819$
#Bundle $\|\mathcal{B}\|$	$1,910$	$1,750$	$1,784$	615
$\mathcal{X}$ Density	0.10%	0.11%	0.11%	0.48%
$\mathcal{Z}$ Density	0.15%	0.20%	0.19%	1.08%
Avg #I/B	3.31	3.52	3.58	5.73
Avg #B/I	1.40	1.76	1.69	1.25
Avg $\|\mathcal{V}_{\mathcal{H}_u}\|$	10.72	11.25	11.80	37.60

Table 1. Statistics of four benchmark datasets.

$$\mathcal{L}_S = \sum_{(v_i, v_j, v_{j'}) \in P} -ln\ \sigma \left(ln(d_{i,j}{}^{-1}) - ln(d_{i,j'}{}^{-1}) \right), \quad (16)$$

where $\sigma(\cdot)$ denotes the Sigmoid function; $d_{i,j}$ represents the distance between the pair of items (v_i, v_j) in the latent space as Equation (5). In addition, $P = \{(v_i, v_j, v_{j'}) \mid v_i, v_j, v_{j'} \in \mathcal{V},\ C_{i,j} = 1,\ v_i \neq v_j,\ C_{i,j'} = 0,\ v_i \neq v_{j'}\}$.

To distill the knowledge from the instruction-driven bundles to the pseudo bundle, we apply the cross-entropy loss over T timesteps as:

$$\mathcal{L}_G = -\frac{1}{T} \sum_{t=1}^{T} \ln \left(p_\theta(r^t \mid v_{1:n}, r_{0:t-1})^\intercal \right) \cdot b^{(t)}_{inst}, \quad (17)$$

where $b^{(t)}_{inst}$ is the target distribution at t given by b_{inst}.

Inspired by [24], the bundle recommendation loss is computed using Bayesian Personalized Ranking loss as:

$$\mathcal{L}_R = \sum_{(u,b,b') \in Q} -ln\sigma \left(y^C_{u,b} - y^C_{u,b'} \right), \quad (18)$$

where $Q = \{(u,b,b') | u \in \mathcal{U}; b, b' \in \mathcal{B}; Z_{u,b} = 1, Z_{u,b'} = 0\}$. Finally, the combined loss function of BRIDGE is achieved as follows:

$$\mathcal{L} = \mathcal{L}_G + \mathcal{L}_S + \mathcal{L}_R + \lambda ||\theta||_2^2, \quad (19)$$

where $||\theta||_2^2$ denotes the $L2$ regularization, and λ indicates a hyper-parameter to control regularization term.

4 Experiments

4.1 Experimental Setup

Datasets. We conduct experiments on four datasets in diverse domains namely Clothing, Electronics, Food and Steam. The data statistics are illustrated in Table 1. Clothing, Electronic, Food [28] are constructed from Amazon with high quality bundles of products using crowd-sourcing resources. Steam[2] [20] includes bundles of games purchased together on the Australian game platform.

Comparative Models. To demonstrate the effectiveness, we compare BRIDGE with several state-of-the-art models for bundle recommendation. *Factorization Models:* BPRMF [24] and DAM [4]; *Graph-based Models:* LightGCN [10], BGCN [3], MIDGN [32], CrossCBR [15], BundleGT [30], MultiCBR [16], CoHeat [11] and BunCa [19]. The overall performances are illustrated in Table 2.

Evaluation Strategy & Metrics. To ensure a fair comparison, we divide the data into training, validation, and test sets using a 7:1:2 ratio, consistent with the comparative models. For performance evaluation, we utilize two typical metrics for the Top-K recommendation

[2] http://cseweb.ucsd.edu/~jmcauley/

Dataset	Metric	BPRMF [24]	DAM [4]	LightGCN [10]	BGCN [3]	MIDGN [32]	CrossCBR [15]	MultiCBR [16]	CoHeat [11]	BundleGT [30]	BunCa [19]	**BRIDGE**	Imp (↑ %)
Steam	R@1	0.0033	0.0016	0.0017	0.0014	0.0022	0.0619	0.0664	0.2355	0.0014	0.0489	**0.3472**†	43.4%
	R@2	0.0046	0.0025	0.0013	0.0218	0.0267	0.1221	0.1139	0.3435	0.0124	0.1253	**0.4298**†	20.6%
	N@1	0.0032	0.0023	0.0017	0.0044	0.0058	0.1074	0.1112	0.2913	0.0013	0.0897	**0.4504**†	54.6%
	N@2	0.0032	0.0043	0.0012	0.0195	0.0212	0.1327	0.1344	0.3336	0.0075	0.1350	**0.4532**†	32.7%
Electronic	R@1	0.0214	0.0135	0.0337	0.0533	0.0781	0.5528	0.4632	0.7514	0.2779	0.6566	**0.8451**†	12.4%
	R@2	0.0275	0.0567	0.0645	0.0672	0.1229	0.7348	0.5917	0.8810	0.3555	0.8044	**0.9673**†	9.7%
	N@1	0.0217	0.0135	0.0345	0.0557	0.1036	0.5907	0.4968	0.8052	0.2981	0.7582	**0.9055**†	12.4%
	N@2	0.0247	0.0324	0.0535	0.0646	0.1621	0.6842	0.5554	0.8544	0.3351	0.7693	**0.9464**†	10.7%
Clothing	R@1	0.0126	0.0174	0.0213	0.0616	0.1057	0.6806	0.5775	0.7197	0.3496	0.7043	**0.8511**†	18.2%
	R@2	0.0282	0.0354	0.0426	0.0937	0.1635	0.8334	0.6995	0.8509	0.3942	0.8594	**0.9871**†	14.9%
	N@1	0.0142	0.0216	0.0328	0.0669	0.1324	0.7356	0.6211	0.7778	0.3667	0.7560	**0.9248**†	18.8%
	N@2	0.0266	0.0305	0.0574	0.0836	0.1781	0.7987	0.6704	0.8245	0.3823	0.8212	**0.9653**†	17.0%
Food	R@1	0.0105	0.0124	0.0193	0.0822	0.0961	0.5665	0.4986	0.7428	0.3177	0.6401	**0.8335**†	12.2%
	R@2	0.0242	0.0210	0.0372	0.1052	0.1986	0.7184	0.6305	0.8885	0.4314	0.7655	**0.9465**†	6.5%
	N@1	0.0133	0.0141	0.0215	0.0887	0.1238	0.6216	0.5399	0.8125	0.3453	0.7019	**0.9041**†	11.2%
	N@2	0.0253	0.0154	0.0307	0.0988	0.3034	0.6821	0.5966	0.8631	0.4028	0.7392	**0.9466**†	9.6%

Table 2. Overall performances on four benchmark datasets. The best results are in **bold**, and the second best results are underlined. The symbol † indicates statistically significant improvements over the second-best models with ($p < 0.01$)

Dataset Metric	*Steam* CoHeat	*Steam* **BRIDGE**	Imp (↑ %)	*Electronic* CoHeat	*Electronic* **BRIDGE**	Imp (↑ %)
R@1	0.2355	**0.3422**	45.3%	0.7514	**0.8451**	12.4%
R@2	0.3434	**0.4228**	23.1%	0.8810	**0.9673**	9.7%
R@5	0.5008	**0.7437**	48.5%	0.9334	**0.9650**	3.3%
R@10	0.6153	**0.7679**	24.8%	0.9481	**0.9691**	2.2%
N@1	0.2914	**0.4559**	56.4%	0.8052	**0.9055**	12.4%
N@2	0.3337	**0.4495**	34.7%	0.8544	**0.9464**	10.7%
N@5	0.4031	**0.5772**	43.1%	0.8797	**0.9580**	8.9%
N@10	0.4460	**0.5856**	31.3%	0.8838	**0.9582**	8.4%
Dataset Metric	*Clothing* CoHeat	*Clothing* **BRIDGE**	Imp (↑ %)	*Food* CoHeat	*Food* **BRIDGE**	Imp (↑ %)
R@1	0.7197	**0.8511**	18.2%	0.7428	**0.8335**	12.2%
R@2	0.8509	**0.9871**	16.0%	0.8885	**0.9465**	6.5%
R@5	0.9099	**0.9969**	9.5%	0.9448	**0.9870**	4.4%
R@10	0.9369	**0.9999**	6.8%	0.9671	**0.9875**	2.1%
N@1	0.7778	**0.9248**	18.8%	0.8125	**0.9041**	11.2%
N@2	0.8245	**0.9653**	17.0%	0.8631	**0.9466**	9.6%
N@5	0.8531	**0.9656**	13.1%	0.8873	**0.9538**	7.4%
N@10	0.8611	**0.9666**	12.2%	0.8957	**0.9537**	6.4%

Table 3. Performance of BRIDGE compared to most comparative baseline over size of retrieval list.

task: Recall ($R@K$) and Normalized Discounted Cumulative Gain ($N@K$). Specifically, $R@K$ reflects the ratio of true recommended bundles to all ground truth bundles (exact matched), calculated as:

$$R@K = \frac{\sum_{i=1}^{k} \mathrm{rel}_i}{\#\mathrm{rel}}$$

where $\mathrm{rel}_i \in \{0, 1\}$ indicates if the i-th item in the ranking list is relevant or not and $\#\mathrm{rel}$ indicates the number of all relevant items. While $N@K$ is higher when the true recommended bundles appear at the top of the retrieval ranking list and is formulated as:

$$DCG@K = \sum_{i=1}^{k} \frac{\mathrm{rel}_i}{\log_2(i+1)}, \;\; iDCG@K = \sum_{i=1}^{k} \frac{1}{\log_2(i+1)},$$
$$N@K = \frac{DCG@K}{iDCG@K},$$

For both metrics, we report the average performance of all models on the test bundles using 5 runs with different random initializations. Significant differences are validated using a two-tailed paired-sample Student's t-test at a 0.01 significance level.

Implementation Details. All comparative models are reproduced using their official source code while BRIDGE is implemented using Flax[3]. We utilize Adam optimizer [12] with the learning rate tuned in the set of $\{1\mathrm{e}-2, 1\mathrm{e}-3, 1\mathrm{e}-4\}$. The number of encoder/decoder blocks, attention heads, and graph encoder layers are empirically selected in the range of values $\{1, 2, 3, 4\}$. For top-K recommendation, we select $K \in \{1, 2\}$ for all datasets to favor ideal recommendation. All experiments are conducted on a single NVIDIA P100 GPU.

4.2 Comparisons with Comparative Models

Table 2 presents the performance comparison between BRIDGE and other comparative models. Conventional recommendation techniques, such as BPRMF and LightGCN, are less effective for bundle recommendation because they focus on optimizing representations of users and bundles but fail to capture item-level information adequately. In contrast, state-of-the-art methods like CrossCBR, MultiCBR, and CoHeat perform well across all four datasets by modeling user preferences at the item level and minimizing inconsistencies between different levels of preferences through multi-view learning. Our model, BRIDGE, consistently outperforms all comparative methods across both evaluation metrics on the four benchmark datasets, demonstrating its effectiveness for bundle recommendation.

For Steam, BRIDGE shows significant improvements from 20.6% to 54.6% compared to the second-best approach. It could be explained by the statistics of Steam dataset, in which the average number of items in a bundle is 5.73, and each item appears in an average of 1.25 bundles. This implies that any two randomly selected bundles are likely to have relatively low similarity, as they will contain different combinations of the available items. If the pseudo bundle has just one item matching the unseen truth bundles, it increases the probability to retrieve the truth bundle. For three Amazon datasets the improvement is still considerable on Electronic from 9.7% to 12.4%, Clothing from 14.9% to 18.8% and Food from 6.5% to 12.2% compared to the second best baseline method.

When comparing on larger retrieval recommendation size, BRIDGE still consistently outperform the second best baseline CoHeat as shown in Table 3, where on the Steam dataset, the improvement is still significant fluctuates between 24.8% to 48.5%.

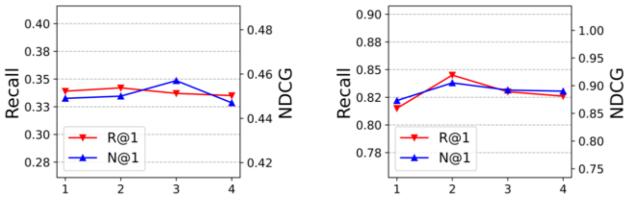
4.3 Model Component Contribution

We conduct various ablation studies to investigate the importance of BRIDGE main components. We mainly present the analysis for the

[3] https://github.com/google/flax

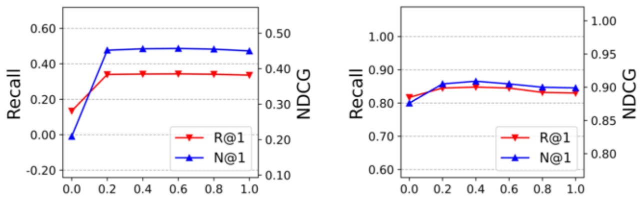
Dataset / Metric	*Steam* R@1	R@2	N@1	N@2	*Electronic* R@1	R@2	N@1	N@2
BRIDGE	**0.3472**	**0.4298**	**0.4504**	**0.4532**	**0.8451**	**0.9673**	**0.9055**	**0.9464**
w/o Inst$_{(\downarrow\%)}$	$0.2758_{\downarrow 19.5\%}$	$0.3267_{\downarrow 24.0\%}$	$0.3657_{\downarrow 18.8\%}$	$0.3574_{\downarrow 21.1\%}$	$0.7020_{\downarrow 16.9\%}$	$0.8295_{\downarrow 14.2\%}$	$0.7621_{\downarrow 15.8\%}$	$0.8045_{\downarrow 15.0\%}$
w/o Gen$_{(\downarrow\%)}$	$0.0711_{\downarrow 79.2\%}$	$0.1283_{\downarrow 70.1\%}$	$0.1078_{\downarrow 76.2\%}$	$0.1322_{\downarrow 70.8\%}$	$0.3944_{\downarrow 53.3\%}$	$0.4392_{\downarrow 54.6\%}$	$0.4280_{\downarrow 52.7\%}$	$0.4346_{\downarrow 54.1\%}$
Dataset / Metric	*Clothing* R@1	R@2	N@1	N@2	*Food* R@1	R@2	N@1	N@2
BRIDGE	**0.8511**	**0.9871**	**0.9248**	**0.9653**	**0.8335**	**0.9465**	**0.9041**	**0.9466**
w/o Inst$_{(\downarrow\%)}$	$0.773_{\downarrow 9.1\%}$	$0.913_{\downarrow 7.4\%}$	$0.843_{\downarrow 8.7\%}$	$0.887_{\downarrow 8.0\%}$	$0.690_{\downarrow 17.1\%}$	$0.822_{\downarrow 13.1\%}$	$0.759_{\downarrow 16.0\%}$	$0.800_{\downarrow 15.4\%}$
w/o Gen$_{(\downarrow\%)}$	$0.361_{\downarrow 57.5\%}$	$0.399_{\downarrow 59.5\%}$	$0.399_{\downarrow 56.8\%}$	$0.397_{\downarrow 58.8\%}$	$0.453_{\downarrow 45.6\%}$	$0.497_{\downarrow 47.4\%}$	$0.493_{\downarrow 45.4\%}$	$0.495_{\downarrow 47.6\%}$

Table 4. Impact of key components on the performance of BRIDGE.



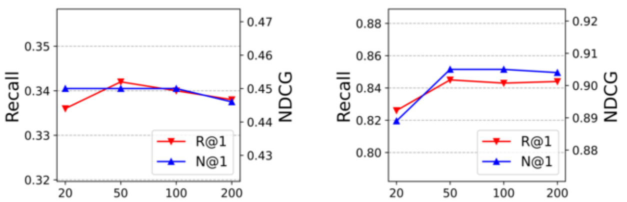
(a) Steam (b) Electronic

Figure 3. Impact of the number of Encoder-Decoder block on Steam and Electronic.



(a) Steam (b) Electronic

Figure 5. Impact of trade-off coefficient α on Steam and Electronic.



(a) Steam (b) Electronic

Figure 4. Impact of max-context length T on Steam and Electronic.

Steam and Electronic datasets, with the remaining results provided in the Appendix.

The ablation results are shown in Table 4. Specifically, for the w/o Inst scenario, instructions are removed, meaning the generation module is trained without any guidance from the instructions. In the w/o Gen scenario, instead of generating pseudo ideal bundles, we aggregate all user instructions as user preferences. **Without guidance from instructions** (w/o Inst), there is a significant drop on both the Steam and Electronic datasets: $R@1$ decreases by 19.5%, $R@2$ by 24%, $N@1$ by 18.8%, and $N@2$ by 21.1% on Steam. On Electronic, the reductions range from 14% to 17%. This shows the importance of training the bundle generation within the relevant guidance signals. Without it, not only does it affect the quality of the generated bundles, but it also drags down the similarity between pseudo bundle and predefined ones due to introducing more noisy items to the 'ideal' bundles. The effectiveness of the generation component is also highlighted. **Without pseudo bundle generation** (w/o Gen), a substantial drop in performance is consistently observed, with reductions exceeding 70% on Steam and over 50% on Electronic.

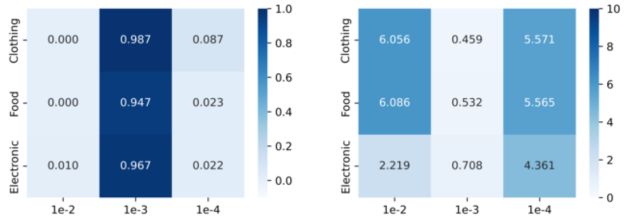
We also conduct experiments to further analyze the impact of the number of encoder-decoder layers and the maximum context length T on *the Pseudo Bundle Generation module*. As shown in Figure 3, the **number of encoder-decoder layers** noticeably impacts model performance. Increasing the number of encoder/decoder blocks from 1 to 2 leads to a clear improvement on Electronic, while performance on Steam remains stable across different numbers of layers. The **maximum context length** T has varying effects on results across different datasets. Figure 3(a) shows that BRIDGE maintains stable performance on Steam across different values of T, while $\mathrm{T} \geq 50$ performs better than $\mathrm{T} = 20$ on Electronic (see Figure 4(b)). This is because a longer context length captures more aspects of user historical interactions. On Steam, where the average number of past interactions is 37.60, $\mathrm{T} = 20$ is insufficient for capturing user collaborative signals. On Electronic, with an average of 11.25 interactions and 8.8% of users having more than 20 past interactions, $\mathrm{T} = 20$ results in performance degradation. With $\mathrm{T} \geq 50$, BRIDGE effectively captures all user preferences, leading to more stable performance.

In the *Retrieval & Ranking module*, we investigate the impact of the **trade-off coefficient** α between two similarities on the model performance as described in Figure 5. Without the combination of two similarity metrics, i.e., $\alpha = 0$ or 1, the BRIDGE's performance drops considerably. On Steam dataset, $\alpha = 0$ makes a sharp drop in performance, while with $\alpha = 1$, $R@1$ decreases from 0.347 to 0.336. The results on the Electronic dataset also show a noticeable decrease in performance for both $\alpha = 0$, and $\alpha = 1$, highlighting the importance of using a combined similarity metric for retrieving bundles. Our model generates a single pseudo 'ideal' bundle for each user but struggles to recommend multiple bundles with varied interests using Jaccard similarity. The model is also experimented with using cosine similarity when retrieving the recommendation list. This allows the generated bundles to cover a broader range of aspects that may be relevant to the user. But this comes at the potential cost of losing the 'ideal' nature of the bundle, as the focus shifts more towards diversity rather than optimization for a single aspect. The trade-off between accuracy and aspect diversity is shown in Table 5. With small values of top-K, BRIDGE performs better with Jaccard (BRIDGE-J), whereas with larger values of top-K, it has lower performance than the variant with Cosine similarity (BRIDGE-C) due to the noisy signal of large bundles. To take advantages of the two variants, BRIDGE combine two retrieval strategies as Eq (15), which obtain the best result over all the retrieval recommendation size.

We also study the sensitivity of BRIDGE to training hyperparameters. Specifically, using a learning rate that is too high or too low (i.e., 1e−2 or 1e−4) results in convergence difficulties for BRIDGE, as shown in Figure 6(b). Compared to a learning rate of 1e−3, the loss values for the other two settings remain significantly

Recall top $-\{K\}$	1	2	5	10	20	50
BRIDGE-J (α = 1)	0.336	0.415	0.741	0.752	0.752	0.780
BRIDGE-C (α = 0)	0.134	0.169	0.227	0.360	0.646	0.961
BRIDGE (full)	**0.342**	**0.423**	**0.744**	**0.768**	**0.800**	**0.979**

Table 5. Performance of BRIDGE variants on Steam over size of retrieval recommendation list.



(a) Recall@2 (b) Total Loss

Figure 6. Accuracy and loss of three Amazon datasets on different learning rates.

higher, even after 100 training epochs. A learning rate that is too high causes overly large gradient steps, while it is too low, even when using a warm-up scheduler, fails to sufficiently optimize BRIDGE within 100 epochs, leading to a noticeable performance drop.

4.4 *Qualitative Analysis*

The quality of a generated pseudo ideal bundle is hard to evaluate. Pathak et al. [20], Han et al. [9] compute the compatibility score between items within the generated bundles as a way to measure their quality. Inspired by these approaches, we evaluate generated bundles through a downstream bundle recommendation task. By such mean, a generated bundle is considered as a meaningful bundle if it is highly similar to an unseen bundle of a user.

Item-Sensitive Instruction. Before demonstrating that the generated bundles are meaningful, we show some examples of the instructions given to the generator are consistent wherein the items have same underlying representation. Figure 7 shows the latent representations for items where items with high relevant are close to each other, some of the meaningful sets are shown: the "*Desktop*" item-sensitive set contains "*Logitech Speakers*", "*Audio Headphones*", "*Wireless Keyboard*", and "*Wireless Mouse*". While the "*Camera*" item-sensitive set is a combination of "*Lens for Digital Cameras*", "*Light Stand*" and "*Backdrop Background*".

Pseudo 'Ideal' Bundle. For user in the testing set, the output bundles generated by our model are highly similar with the ground truth bundles at item level. This leads to very significant improvement over all baseline methods, on Steam the generated bundle tends to be a jointly subset of ground truth bundles that make BRIDGE can retrieve multiple bundles with different aspects which match user varied interest. The product bundles generated by the BRIDGE model have been shown to exhibit meaningful and useful combinations of items, as illustrated in Figure 8. The occurrence of items in a generated bundles is similar compare to the bundles in retrieval recommendation list of users. Additionally, the distribution of the generated bundles is similar to the distribution of the ground truth bundles, which demonstrates that the generated bundles cover all the aspects of the pre-constructed bundles in a diverse manner.

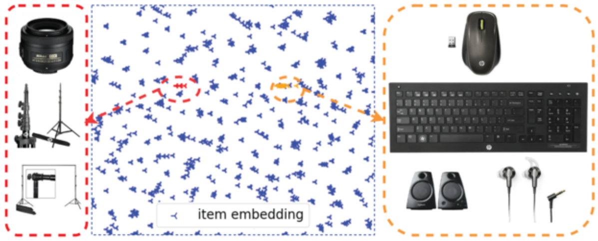


Figure 7. T-SNE visualization of item-sensitive instruction on Electronic.

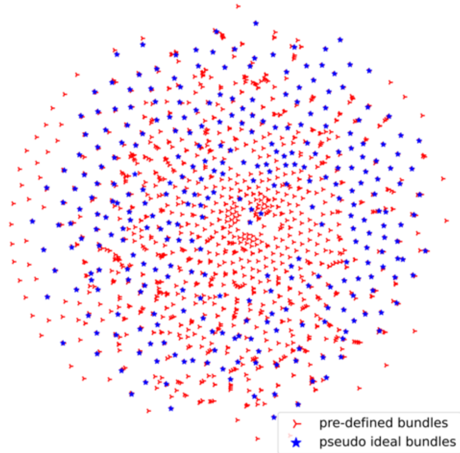


Figure 8. T-SNE visualization of pseudo ideal bundles and ground-truth ones on Electronic.

4.5 *Complexity Analysis*

Space Complexity. For the comparative baseline methods follow matrix factorization-based framework [15, 16, 11], the space complexity include user,bundle and item embedding hence the space complexity is: $O\left((|\mathcal{U}| + |\mathcal{B}| + |\mathcal{I}|) \times d\right)$ where d is the number of embedding space dimension. For BRIDGE the complexity include storing user and item embedding: $O\left((|\mathcal{U}| + |\mathcal{I}|) \times d\right)$ with model's learnable parameters follow [29] and our mentioned configuration.

Time Complexity. Time complexity analysis of CoHeat can be found in [11]. For theoretical complexity analysis of BRIDGE, the *Item-Sensitive Instruction Generation* module is $O(|\mathcal{I}| log(|\mathcal{I}|))$ for sorting and retrieving top-k highest items while the *Pseudo Bundle Generation* module is $O((\mathrm{T}^2 d + d^2 \mathrm{T}) \times L)$ with T, d, L is the number of token, dimension, the total layer of encoder-decoder, respectively. For Steam, CoHeat takes $42.4s$ for each training epoch and $0.66s$ for each mini-batch inference. With a similar efficiency, BRIDGE takes $41s$ for each training epoch, $2.43s$ for each inference.

5 Conclusion

This paper introduces BRIDGE, a novel framework for bundle recommendation inspired by distant supervision strategies and the generative paradigm. The framework integrates item-sensitive instruction generation module that enables the generation of auxiliary information without relying on external data. By combining this with collaborative signals from user historical interactions, BRIDGE can produce pseudo 'ideal' bundles, which allows exploring a broader range of potential bundles beyond those predefined ones. This approach effectively narrows the gap between user expectations and available bundles, leading to the soundness over prominent models for bundle recommendation on four public datasets.

References

[1] T.-N. Bui, H.-S. Nguyen, C.-V. T. Nguyen, H.-Q. Le, and D.-T. Le. Personalized diffusion model reshapes cold-start bundle recommendation. In *Companion Proceedings of the ACM on Web Conference 2025*, pages 3088–3091, 2025.

[2] J. Cao, X. Cong, J. Sheng, T. Liu, and B. Wang. Contrastive cross-domain sequential recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 138–147, 2022.

[3] J. Chang, C. Gao, X. He, D. Jin, and Y. Li. Bundle recommendation with graph convolutional networks. In *Proceedings of the 43rd international ACM SIGIR conference on Research and development in Information Retrieval*, pages 1673–1676, 2020.

[4] L. Chen, Y. Liu, X. He, L. Gao, and Z. Zheng. Matching user with item set: Collaborative bundle recommendation with deep attention network. In *IJCAI*, pages 2095–2101, 2019.

[5] M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery. Learning to extract symbolic knowledge from the world wide web. *AAAI/IAAI*, 3(3.6):2, 1998.

[6] Z. Cui, X. Xu, X. Fei, X. Cai, Y. Cao, W. Zhang, and J. Chen. Personalized recommendation system based on collaborative filtering for iot scenarios. *IEEE Transactions on Services Computing*, 13(4):685–695, 2020.

[7] A. M. Elkahky, Y. Song, and X. He. A multi-view deep learning approach for cross domain user modeling in recommendation systems. In *Proceedings of the 24th international conference on world wide web*, pages 278–288, 2015.

[8] M. Gutmann and A. Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 297–304. JMLR Workshop and Conference Proceedings, 2010.

[9] X. Han, Z. Wu, Y.-G. Jiang, and L. S. Davis. Learning fashion compatibility with bidirectional lstms. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1078–1086, 2017.

[10] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 639–648, 2020.

[11] H. Jeon, J.-e. Lee, J. Yun, and U. Kang. Cold-start bundle recommendation via popularity-based coalescence and curriculum heating. In *Proceedings of the ACM on Web Conference 2024*, pages 3277–3286, 2024.

[12] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[13] D.-T. Le, H. W. Lauw, and Y. Fang. Correlation-sensitive next-basket recommendation.(2019). In *28th International Joint Conference on Artificial Intelligence, Macao, China*, pages 10–16, 2019.

[14] X. Lin, F. Petroni, G. Bertasius, M. Rohrbach, S.-F. Chang, and L. Torresani. Learning to recognize procedural activities with distant supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13853–13863, 2022.

[15] Y. Ma, Y. He, A. Zhang, X. Wang, and T.-S. Chua. Crosscbr: Cross-view contrastive learning for bundle recommendation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1233–1241, 2022.

[16] Y. Ma, Y. He, X. Wang, Y. Wei, X. Du, Y. Fu, and T.-S. Chua. Multicbr: Multi-view contrastive learning for bundle recommendation. *ACM Transactions on Information Systems*, 42(4):1–23, 2024.

[17] M. Mintz, S. Bills, R. Snow, and D. Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011, 2009.

[18] H.-S. Nguyen, T.-N. Bui, L.-H. Nguyen, D.-C. Can, C.-V. T. Nguyen, D.-T. Le, and H.-Q. Le. Hhmc: a heterogeneous x homogeneous graph-based network for multimodal cross-selling recommendation. In *2023 15th International Conference on Knowledge and Systems Engineering (KSE)*, pages 1–6. IEEE, 2023.

[19] H.-S. Nguyen, T.-N. Bui, L.-H. Nguyen, H. Hoang, C.-V. Thi Nguyen, H.-Q. Le, and D.-T. Le. Bundle recommendation with item-level causation-enhanced multi-view learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 324–341. Springer, 2024.

[20] A. Pathak, K. Gupta, and J. McAuley. Generating and personalizing bundle recommendations on steam. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1073–1076, 2017.

[21] Y. Qi, W. Zhao, and X. Wu. Relational distant supervision for image captioning without image-text pairs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 4524–4532, 2024.

[22] C. Quirk and H. Poon. Distant supervision for relation extraction beyond the sentence boundary. *arXiv preprint arXiv:1609.04873*, 2016.

[23] S. Rajput, N. Mehta, A. Singh, R. Hulikal Keshavan, T. Vu, L. Heldt, L. Hong, Y. Tay, V. Tran, J. Samost, et al. Recommender systems with generative retrieval. *Advances in Neural Information Processing Systems*, 36, 2024.

[24] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618*, 2012.

[25] C. Sun, H. Liu, M. Liu, Z. Ren, T. Gan, and L. Nie. Lara: Attribute-to-feature adversarial learning for new-item recommendation. In *Proceedings of the 13th international conference on web search and data mining*, pages 582–590, 2020.

[26] M. Sun, L. Li, M. Li, X. Tao, D. Zhang, P. Wang, and J. X. Huang. A survey on bundle recommendation: Methods, applications, and challenges. *arXiv preprint arXiv:2411.00341*, 2024.

[27] W. Sun, R. Xie, J. Zhang, W. X. Zhao, L. Lin, and J.-R. Wen. Generative next-basket recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pages 737–743, 2023.

[28] Z. Sun, J. Yang, K. Feng, H. Fang, X. Qu, and Y. S. Ong. Revisiting bundle recommendation: Datasets, tasks, challenges and opportunities for intent-aware product bundling. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2900–2911, 2022.

[29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[30] Y. Wei, X. Liu, Y. Ma, X. Wang, L. Nie, and T. Chua. Strategy-aware bundle recommender system. In *SIGIR*. ACM, 2023.

[31] Z. Yang, J. Wu, Z. Wang, X. Wang, Y. Yuan, and X. He. Generate what you prefer: Reshaping sequential recommendation via guided diffusion. *Advances in Neural Information Processing Systems*, 36, 2024.

[32] S. Zhao, W. Wei, D. Zou, and X. Mao. Multi-view intent disentangle graph networks for bundle recommendation. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence*, 2022.

[33] Y. Zhou, J. Xu, J. Wu, Z. Taghavi, E. Korpeoglu, K. Achan, and J. He. Pure: Positive-unlabeled recommendation with generative adversarial network. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 2409–2419, 2021.