

Hierarchical Alignment with Polar Contrastive Learning for Next-basket Recommendation

Ting-Ting Su, Chang-Dong Wang, *Member, IEEE*, Wu-Dong Xi, Jian-Huang Lai, *Senior Member, IEEE*, Philip S. Yu, *Life Fellow, IEEE*

Abstract—Next-basket recommendation methods focus on the inference of the next basket by considering the corresponding basket sequence. Although many methods have been developed for the task, they usually suffer from data sparsity. The number of interactions between entities is relatively small compared to their huge bases, so it is crucial to mine as much hidden information as possible from the limited historical interactions for prediction. However, the existing methods mainly just treat the next-basket recommendation task as a single-view sequential prediction problem, which leads to the inadequate mining of the information hidden in multiple views, and the mining of other patterns in the historical interactions is neglected, thus making it difficult to learn high-quality representations and limiting the recommendation effect. To alleviate the above issues, we propose a novel method named HapCL for next-basket recommendation, which mines information from multiple views and patterns with the help of polar contrastive learning. A hierarchical module is designed to mine multiple patterns of historical interactions from different views at two levels. In order to mine self-supervised signals, we design a polar contrastive learning module with a novel graph-based augmentation approach. Experiments on three real-world datasets validate the effectiveness of HapCL.

Index Terms—Next-basket recommendation, contrastive learning, multi-head, distribution alignment

1 INTRODUCTION

WITH the rapid development of the Internet, the number of entities involved has exploded. In many scenarios, users need to search through a large number of items to find the one that fits their needs, but it is impossible for them to go through all of items. In order to help users find the target item quickly, recommendation systems came into being. Recommendation systems try to mine useful information from the historical interactions and match items for users based on the mined information [1], [2], [3]. Hence, the time that users spend in finding their target items is saved greatly.

The next-basket recommendation task, which aims to infer several items that the target user interacts with in the next basket based on the corresponding basket sequence, has drawn increasing attention [3], [4], [5], [6]. Its success is mainly due to the fact that the order of user interaction over a period of time does not necessarily follow a strict chronological order, which is also an important difference between next-basket recommendation and sequential recommendation [7], [8], [9], [10], [11]. However, a common issue in recommendation methods is data sparsity [12], [13], [14]. The number of interactions between users and items is relatively small compared to their huge bases, which results in insufficient data for learning various representations, e.g., basket

representations, in recommendation methods. It is crucial to make full use of the limited historical interactions.

Some recommendation methods attempt to mine self-supervised signals from the original data and employ contrastive learning to help recommendation models with learning high-quality representations [4], [10]. Since the data form of the historical interactions, which is time-aware, is not as easy to be augmented as images, the augmentation approaches in next-basket recommendation remain to be explored. More specifically, Fig. 1 shows a basket sequence consists of three baskets with various sizes. The order of items in every basket has no strict precedence, i.e., the order of items in every basket can be changed. The augmentation approaches, e.g., rotation, do not apply to time-aware data. In addition, the existing time-aware methods [3], [15], [16], [17] mainly concentrate on the sequential pattern of the historical interactions in a single view, which results in the inadequate mining of the information hidden in multiple views, and the mining of other patterns, e.g., graph pattern, in the historical interactions is neglected. Take the sequence that is shown in Fig. 1 for example, the lipstick and pressed powder in the first basket and the eyeliner in the second basket are all cosmetics. Sequential pattern mining pays attention to the view of category, so that it tends to recommend eye shadow in the following basket. But from the view of interest, the buying of the cosmetics can also imply the user attaches importance to appearance, where the perfume matches the preference too. And from the view of consumability, cosmetics and skin care products are consumable, which would not be changed during consumption and need to be re-purchased when they are used up, while hairpins can be reused and users always buy a variety of styles for recycling. The sequential methods that focus only on sequential pattern belittle the complex relationships between baskets and items, which are intricately correlated across different views and need to be mined from more patterns. Note that the concrete views mentioned above

- Ting-Ting Su, Chang-Dong Wang and Jian-Huang Lai are with School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China, Guangdong Province Key Laboratory of Computational Science, Guangzhou, China, and Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, Guangzhou, China. sutt8@mail2.sysu.edu.cn, changdongwang@hotmail.com, stsljh@mail.sysu.edu.cn.
- Wu-Dong Xi is with UX center, NetEase Games at Guangzhou, China. xiwudong@corp.netease.com.
- Philip S. Yu is with Department of Computer Science, University of Illinois at Chicago, Chicago, IL, USA. psyu@cs.uic.edu.

Chang-Dong Wang is the corresponding author.

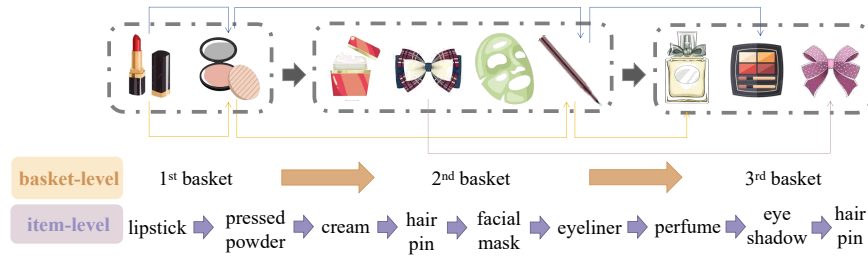


Fig. 1. An example of the historical interactions in the basket sequence form with the length of 3. The three baskets have different sizes and the order of items in every basket can be changed. The sequence can be displayed at both basket-level and item-level. Next-basket recommendation aims at inferring the items belonging to the third basket based on the preceding two baskets.

are hypotheses for the purpose of illustrating motivation. The views in real-world are implicit, and too convoluted to explain their meaning unambiguously. Different views can be explored and decoupled through the combination of different patterns.

All these deficiencies lead to insufficient mining of information from the limited historical interactions, thus making it difficult to learn high-quality representations and limiting the recommendation effect. In this paper, we propose a novel method for next-basket recommendation (named HapCL), which designs a **H**ierarchical **a**lignment module and a **p**olar **C**ontrastive **L**earning module to alleviate the above issues. The HapCL method constructs a weighted bipartite graph based on the historical interactions to help with mining higher-order correlation between baskets and items. Basket sequences are encoded by a hierarchical module to mine the information at basket-level and item-level, respectively. At both levels, hidden information will be mined in multiple views, and be integrated into an overall probability prediction for capturing the complex relationship between baskets and items. In order to make the predictions at different levels supervise each other, they are constrained to achieve distribution alignment. Moreover, a novel graph-based augmentation approach and a polar contrastive learning strategy are designed for the mining of self-supervised signals. Every basket in basket sequences will be augmented to obtain a positive augmentation and a negative augmentation. The two augmentations are used for polar contrastive learning, which is an auxiliary task to help the next-basket recommendation task mine more hidden information from the limited historical interactions and learn high-quality representations for baskets and items. To sum up, the contributions of this paper are as follows:

- We propose a novel HapCL method that designs hierarchical alignment with polar contrastive learning for next-basket recommendation. The method adopts a hierarchical framework to mine information from multiple views at both basket-level and item-level for taking full advantage of the limited historical interactions.
- We propose a graph-based augmentation approach to construct polar self-supervised signals for contrastive learning. The polar contrastive learning is applied as an auxiliary task to promote the next-basket recommendation. The impact of the polar self-supervised signals is validated by ablation study.
- We conduct extensive experiments on three real-world datasets to demonstrate the effectiveness of the proposed HapCL method.

The rest of this paper is arranged as follows. In Section 2, we will

briefly introduce the research status by reviewing related work. In Section 3, each part of the proposed method will be detailed. In Section 4, extensive experiments will be designed and conducted to confirm the effectiveness of the proposed method. In the end, we will draw the conclusion of this paper in Section 5.

2 RELATED WORK

In this section, we briefly review three tasks related to our work, namely graph-based recommendation, next-basket recommendation and contrastive learning.

2.1 Graph-based Recommendation

Graph is a form of data which can represent the association between nodes through the connection between them, which well fits the association between entities in recommendation. Some works have realized the consistency, and as deep neural networks have proven to be powerful in data mining in recent years, they have proposed some graph-based deep learning methods that model the association between entities by graph and learn representations for nodes in graph for recommendation. [18] employs random walk to extract some node sequences from graph. All the sequences are regarded as a corpus where each node is encoded into an embedding. [19] and [20] design graph convolutional neural network for learning the representations of the nodes in graph, and recommend items for users according to the similarity between the representations. Considering the different importance of different nodes, [21] introduces attention mechanism into graph convolution neural network, which assigns different nodes with different weights when aggregating neighbor nodes. [1] constructs a directed graph for item sequences and employs self-attention layers to assign different weights for the items in sequences. Moreover, in order to provide more information for model to learn high-quality representations, some methods [22], [5], [23] attempt to add additional information, e.g., attributes of items, properties of associations, and social relations of users, into graph, which are outside the scope of this paper where we focus on how to make the best use of the limited historical interactions.

2.2 Next-basket Recommendation

With the assumption that the shopping preference of users and the correlation between items are often reflected in the order of the historical interactions, temporal recommendation methods are proposed to mine the sequential pattern hidden in interactions and make recommendation based on it. Temporal recommendation can be divided into two main categories according to the

units of sequences: sequential recommendation which aims at recommending next item based on the item sequence and basket recommendation which aims at recommending next basket based on the basket sequence. Note that the baskets in basket sequences consist of several items, and have no fixed size. Although the units of sequences in these two tasks are different, the solutions for them share many similarities. [24] combines the advantages of matrix factorization and Markov chain to consider both the overall taste of users and their current preferences. [6] encodes basket sequence by recurrent neural network to learn the dynamic representation of users, and captures sequential features among baskets. With the hypothesis that the correlation between items would help to infer more coherent basket recommendation, [3] mines the correlation and infers the next basket with the correlation for a basket sequence by a hierarchical network, thus validating the hypothesis. Some studies have noted that most methods in next-basket recommendation focus on the mining of sequential patterns but ignore other patterns, e.g., graph pattern. [25] models the relationship among the items, users and baskets by a heterogeneous graph, and regards the next-basket recommendation task as a link prediction task. However, the methods that can integrate the superiority of different patterns effectively remain to be explored.

2.3 Contrastive Learning

Although deep learning has achieved great success in various fields in recent years, the issue of data sparsity still plagues many deep models. In the recommendation field, the interaction data between users and items is relatively small compared to their huge bases, but it requires a lot of labeled data to train the large number of parameters in deep models, e.g., historical interactions. The conflict between the supply of labeled data and the demand of models requires us to make full use of the limited historical interactions and mine as much information as possible. Contrastive learning, which mines self-supervised signals from unlabeled data by constructing contrastive pairs, has been widely used in some fields for alleviating the issue caused by sparse data. The approach of data augmentation and the way of constructing contrastive pairs are the key problems in contrastive learning. In the field of computer vision, [26] proposes a texture-based augmentation and a patch-based augmentation to generate negative samples which preserve superfluous instead of the semantic features for contrastive learning. [27] introduces Gaussian noise to perturb the feature embedding during training, thus mining self-supervised signals. When it comes to the field of recommendation, [10] performs data augmentation by masking or segmenting the items in item sequences, and learns the representations of items and attributes with the correlation among attributes, items, segments and sequences. [4] derives two sub-basket sequences at item-level according to the relevance among the items, and encodes the sequence pairs for contrastive learning. Although there have been some successful attempts in making use of contrastive learning to benefit the recommendation tasks, the research on combining contrastive learning with the next-basket recommendation task for addressing the issues caused by data sparsity is still limited.

3 THE PROPOSED METHOD

In this section, we first formalize the next-basket recommendation problem. Then, we represent an overview of the proposed **HapCL** method, followed by describing the two main modules of this

TABLE 1
Summary of the main notations in this paper.

Notations	Descriptions
$\mathcal{I}$	The set of items
$\mathcal{U}$	The set of users
b	A basket consisting of several items
$\mathcal{P}$	The set of baskets
S^u	The basket sequence of user u at basket-level
$S^{(u,I)}$	The basket sequence of user u at item-level
V	The set of nodes
E	The set of edges
$\mathcal{G}$	The weighted bipartite graph
w_{ib}	The weight of edge connecting item i and basket b
$\mathcal{N}_i$	The set of basket nodes connecting with item node i
$\mathcal{N}_b$	The set of item nodes connecting with basket node b
$\mathbf{e}_b^{(l)}$	The embedding of basket b after l layers' propagation
$\mathbf{e}_i^{(l)}$	The embedding of item i after l layers' propagation
$\mathbf{e}_{(t)}^{(u,B)}$	The embedding of the t -th basket in S^u
$\mathbf{e}_{(t)}^{(u,I)}$	The embedding of the t -th item in $S^{(u,I)}$
$\mathbf{s}^{(u,B)}$	The representation of S^u
$\mathbf{s}^{(u,I)}$	The representation of $S^{(u,I)}$
α_{kh}^B	The attention weight of the k -th view over the h -th view at basket-level
α_{kh}^I	The attention weight of the k -th view over the h -th view at item-level
$m_{(i,h)}^{(u,B)}$	The probability that item i is in the next basket following S^u from the h -th view at basket-level
$m_{(i,h)}^{(u,I)}$	The probability that item i is in the next basket following S^u from the h -th view at item-level
w_h^B	The weight of the h -th view at basket-level
w_h^I	The weight of the h -th view at item-level
$y_i^{(u,B)}$	The probability that item i is in the next basket following S^u at basket-level
$y_i^{(u,I)}$	The probability that item i is in the next basket following S^u at item-level
y_i^u	The final prediction of the probability that item i is in the next basket following S^u
r_{bi}	The relevance between basket b and item i

method, i.e., *Hierarchical Alignment for Next-basket Recommendation* and *Graph-based Polar Contrastive Learning*. Finally, the multi-task learning strategy is described.

3.1 Problem Formulation

The main notations used in this paper are summarized in Table 1.

Let $\mathcal{I} = \{i_1, i_2, \dots, i_{|\mathcal{I}|}\}$ denote the set of items and $\mathcal{U} = \{u_1, u_2, \dots, u_{|\mathcal{U}|}\}$ denote the set of users, where $|\mathcal{I}|$ and $|\mathcal{U}|$ represent the number of items and users, respectively. Each basket $b = \{i_1^b, i_2^b, \dots, i_{|b|}^b\}$ is composed of $|b|$ items in $\mathcal{I}$ that are interacted by the same user over a period of time, and the basket size $|b|$ may be different from one basket to another. Let $\mathcal{P} = \{b_1, b_2, \dots, b_{|\mathcal{P}|}\}$ denote the set of baskets, where $|\mathcal{P}|$ represents the number of baskets. And the historical interactions of user $u \in \mathcal{U}$ can be denoted by basket sequence

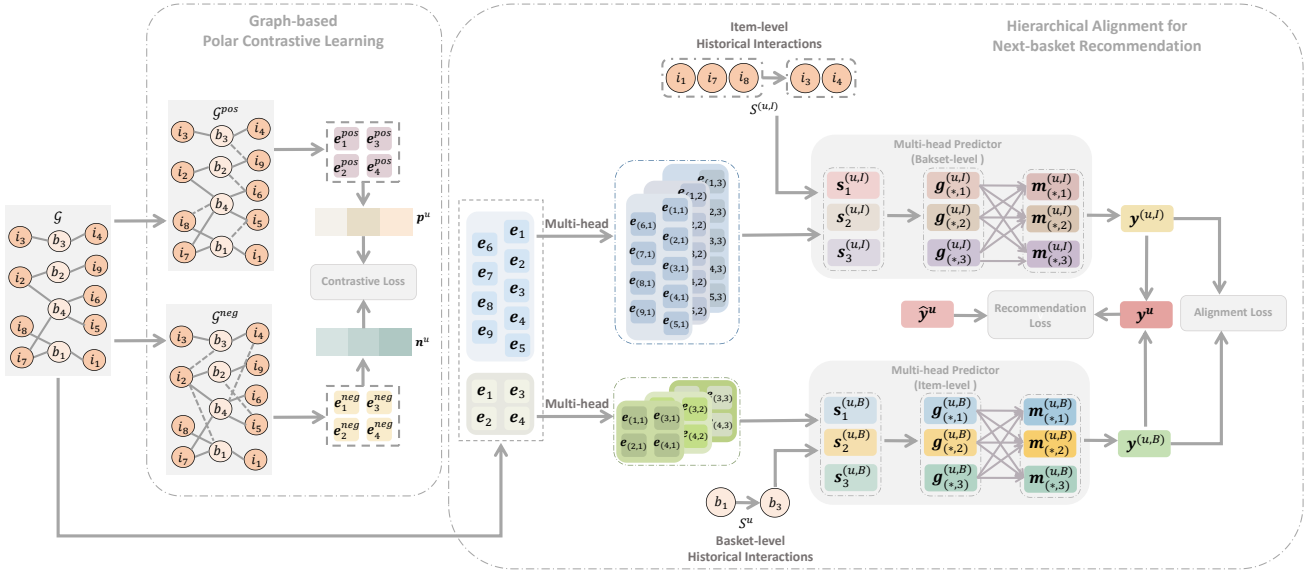


Fig. 2. An overview of the architecture of the proposed HapCL method, which can be divided to two parts, i.e., Hierarchical Alignment for Next-basket Recommendation and Graph-based Polar Contrastive Learning.

$S^u = [b_1^u, b_2^u, \dots, b_{|S^u|}^u]$ in chronological order, where $|S^u|$ represents the length of S^u .

For every anonymous user $u \in \mathcal{U}$, the next-basket recommendation aims at the inference of $|b_{|S^u|+1}|$ items that user u will interact with in the next basket based on basket sequence S^u , where $|b_{|S^u|+1}|$ is the size of the next basket.

3.2 Overview

In order to mine as much information as possible from the historical interactions, and alleviate the issues caused by data sparsity, we propose the HapCL method as shown in Fig. 2. The method models historical interactions by a weighted bipartite graph to mine graph pattern, and designs a module with hierarchical framework to mine sequential pattern at basket-level and item-level. At both levels, every sequence is encoded to multiple representations to explore hidden information from different views. With the expectation that predictions at different levels can help supervise each other, distribution alignment is applied to put a constraint on them. Furthermore, we design a novel graph-based augmentation approach to construct positive augmentations and negative augmentations of baskets for polar contrastive learning, which is an auxiliary task to help with mining self-supervised signals and learning high-quality representations of baskets and items for recommendation.

3.3 Hierarchical Alignment for Next-basket Recommendation

3.3.1 Graph-based Embedding

To extract higher-order correlation between baskets and items from the historical interactions, we construct the weighted bipartite graph $\mathcal{G} = (V, E)$, where $V = \mathcal{P} \cup \mathcal{I}$ denotes the set of nodes, and $E \subseteq \{(b, i) | b \in \mathcal{P} \wedge i \in \mathcal{I}\}$ denotes the set of edges that represent the association between baskets and items. The weight of every edge in $\mathcal{G}$ is assigned with the number of occurrences of the corresponding basket. For example, there are historical interactions of two users in Fig. 3, which contains 4

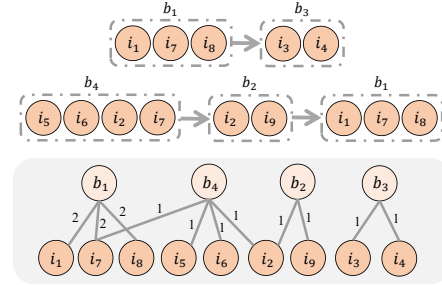


Fig. 3. An example for the construction of weighted bipartite graph. The two sequences contain 4 baskets involving 9 items, and the corresponding weighted bipartite graph is shown in the grey block.

baskets involving 9 items. The bipartite graph can be constructed by adding edges between every basket node and the nodes of items assembled into it. Since the basket b_1 occurs twice in this example, the weight of the edges connected to b_1 is assigned with 2, while the weight of the edges connected to the other baskets that only occur once is assigned with 1. The graph can help with the propagation of correlation between baskets and items, and the aggregation of higher-order information for their embeddings.

For the basket node b in $\mathcal{G}$, it can be represented by the embedding $\mathbf{e}_b \in \mathbb{R}^{1 \times d}$, where d denotes the dimension. Similarly, $\mathbf{e}_i \in \mathbb{R}^{1 \times d}$ represents the embedding of the item node i in $\mathcal{G}$. In order to propagate information between nodes and aggregate higher-order information for updating embeddings, as well as reduce computational cost, we perform the light graph convolution operation as follows [20]:

$$\mathbf{e}_b^{(l+1)} = \sum_{i \in \mathcal{N}_b} \frac{w_{ib}}{\sqrt{\sum_{i' \in \mathcal{N}_b} w_{i'b}} \sqrt{\sum_{b' \in \mathcal{N}_i} w_{ib'}}} \mathbf{e}_i^{(l)}, \quad (1)$$

$$\mathbf{e}_i^{(l+1)} = \sum_{b \in \mathcal{N}_i} \frac{w_{ib}}{\sqrt{\sum_{b' \in \mathcal{N}_i} w_{i'b'}} \sqrt{\sum_{i' \in \mathcal{N}_b} w_{i'b}}} \mathbf{e}_b^{(l)} \quad (2)$$

where w_{ib} denotes the weight of edge connecting item node i and basket node b , $\mathcal{N}_i$ denotes the set of basket nodes connecting with item node i (i.e., the baskets containing item i), $\mathcal{N}_b$ denotes the set of item nodes connecting with basket node b (i.e., the items contained by basket b), and $\mathbf{e}_b^{(l)}$ and $\mathbf{e}_i^{(l)}$ denote the embedding of basket b and item i after l layers' propagation, respectively. Note that the more times basket b appears in the historical interactions, the higher the weight of w_{ib} , which indicates the frequency that items belong to the same basket, so the correlation between items and basket should be hidden in the weights and mined by iterative propagation and aggregation.

Only the embeddings in the 0-th layer, i.e., $\mathbf{e}_b^{(0)}$ for baskets and $\mathbf{e}_i^{(0)}$ for items, will be trained, and the embeddings in higher layers will be updated according to them. After L iterations, we can obtain additional L embeddings for every basket and every item, which encode higher-order information. They will be further combined for the final embeddings as follows:

$$\mathbf{e}_b = \frac{1}{L+1} \sum_{l=0}^L \mathbf{e}_b^{(l)}, \quad (3)$$

$$\mathbf{e}_i = \frac{1}{L+1} \sum_{l=0}^L \mathbf{e}_i^{(l)}. \quad (4)$$

Since the graph $\mathcal{G}$ is sparse, it is difficult to learn high-quality embeddings for items and baskets. To alleviate the issue, we design a novel graph-based augmentation approach for polar contrastive learning in Section 3.4 to derive self-supervised signals based on the original graph.

3.3.2 Hierarchical Multi-head Predictor

For tasks with sparse data, it is crucial to mine as much hidden information as possible from the data. It is obvious that the basket sequence $S^u = [b_1^u, b_2^u, \dots, b_{|S^u|}^u]$ can be further expanded into $S^{(u,I)} = [\{i_1^u, \dots, i_{|b_1^u|}^u\}, \{i_1^u, \dots, i_{|b_2^u|}^u\}, \dots, \{i_1^u, \dots, i_{|b_{|S^u|}^u|}^u\}]$.

The basket sequence S^u and its expanded version $S^{(u,I)}$ are the different formats of the same historical interactions, but they contain the information at different levels, i.e., basket-level and item-level. It is worth mentioning that items in any basket in $S^{(u,I)}$ have no strict chronological order, which is different from item sequence with strict chronological order in sequential recommendation methods.

The order of the historical interactions reflects the dynamic interest of users. We employ Gated Recurrent Unit (GRU) [28] to encode the historical interactions for the purpose of capturing the dynamic interests as follows:

$$\mathbf{s}_{(t+1)}^{(u,B)} = \text{GRU}(\mathbf{e}_{(t)}^{(u,B)}, \mathbf{s}_{(t)}^{(u,B)}), \quad (5)$$

$$\mathbf{s}_{(t+1)}^{(u,I)} = \text{GRU}(\mathbf{e}_{(t)}^{(u,I)}, \mathbf{s}_{(t)}^{(u,I)}) \quad (6)$$

where $\mathbf{e}_{(t)}^{(u,B)} \in \mathbb{R}^{1 \times d}$ denotes the embedding of the t -th basket in the basket sequence of the user u at basket-level S^u , $\mathbf{s}_{(t)}^{(u,B)} \in \mathbb{R}^{1 \times d_m}$ encodes the preceding t baskets, $\mathbf{e}_{(t)}^{(u,I)} \in \mathbb{R}^{1 \times d}$ denotes the embedding of the t -th item in the basket sequence of the user u at item-level $S^{(u,I)}$, and $\mathbf{s}_{(t)}^{(u,I)} \in \mathbb{R}^{1 \times d_m}$ encodes the preceding t items, with d_m denoting their hidden dimensions. The final representation of sequence S^u will be denoted by $\mathbf{s}^{(u,B)}$, i.e., $\mathbf{s}_{|S^u|}^{(u,B)}$, which encodes the dynamic interests of the user u at

basket-level. Similarly, the final representation of sequence $S^{(u,I)}$ will be denoted by $\mathbf{s}^{(u,I)}$. Note that the GRU layer can be replaced with any layer capable of capturing sequential information, e.g., LSTM [29] and Transformer [30]. Considering that we adopt GRU4Rec [31] as a baseline method in our experiments, we select GRU for HapCL in this paper to serve as a control to reflect the performance of HapCL itself.

With the representations $\mathbf{s}^{(u,B)}$ and $\mathbf{s}^{(u,I)}$, the probability that item i is in the next basket following S^u can be inferred as follows:

$$g_i^{(u,B)} = \mathbf{s}^{(u,B)} \cdot \mathbf{e}_i^T, \quad (7)$$

$$g_i^{(u,I)} = \mathbf{s}^{(u,I)} \cdot \mathbf{e}_i^T \quad (8)$$

where $g_i^{(u,B)}$ and $g_i^{(u,I)}$ denote the probability at basket-level and item-level, respectively.

Inspired by [30], we map the embeddings of baskets and items into multiple hidden spaces with multi-head, so as to explore hidden information from different views. Thus, every sequence will infer multiple basket-level probabilities $\{g_{(i,1)}^{(u,B)}, \dots, g_{(i,H)}^{(u,B)}\}$ and multiple item-level probabilities $\{g_{(i,1)}^{(u,I)}, \dots, g_{(i,H)}^{(u,I)}\}$ for item i , where H denotes the number of heads. It's worth noting that different views are not independent of each other, as they pay attention to each other in different degrees. In order to get closer to the fact, we introduce attention mechanism for the inference of probabilities as follows:

$$m_{(i,k)}^{(u,B)} = \sum_{h=1}^H \alpha_{kh}^B g_{(i,h)}^{(u,B)}, \quad (9)$$

$$m_{(i,k)}^{(u,I)} = \sum_{h=1}^H \alpha_{kh}^I g_{(i,h)}^{(u,I)} \quad (10)$$

where α_{kh}^B and α_{kh}^I denote the attention weight of the k -th view over the h -th view at basket-level and item-level, respectively. They will be calculated by

$$\alpha_{kh}^B = \frac{\exp(\mathbf{s}_k^{(u,B)} \cdot \mathbf{s}_h^{(u,B)T})}{\sum_{v=1}^H \exp(\mathbf{s}_k^{(u,B)} \cdot \mathbf{s}_v^{(u,B)T})}, \quad (11)$$

$$\alpha_{kh}^I = \frac{\exp(\mathbf{s}_k^{(u,I)} \cdot \mathbf{s}_h^{(u,I)T})}{\sum_{v=1}^H \exp(\mathbf{s}_k^{(u,I)} \cdot \mathbf{s}_v^{(u,I)T})} \quad (12)$$

where $\mathbf{s}_k^{(u,B)}$ and $\mathbf{s}_k^{(u,I)}$ denote the basket-level and item-level representations of S^u in the k -th view.

Because the probabilities inferred based on a context from different views have different importance in the final probability, it is indispensable to assign different weights to them. We perform the combination of the probabilities as follows:

$$y_i^{(u,B)} = \sum_{h=1}^H w_h^B \times m_{(i,h)}^{(u,B)}, \quad (13)$$

$$y_i^{(u,I)} = \sum_{h=1}^H w_h^I \times m_{(i,h)}^{(u,I)} \quad (14)$$

where w_h^B and w_h^I denote the weight of the h -th view at basket-level and item-level, respectively.

As we obtain the two probabilities at different levels by the hierarchical network, the final prediction of the probability that

item i is in the next basket following S^u can be calculated as follows:

$$y_i^u = \frac{1}{1 + \exp(-(y_i^{(u,B)} + y_i^{(u,I)}))}. \quad (15)$$

The next-basket recommendation task will be optimized with the $\mathcal{L}_{rec}$ defined as follows:

$$\begin{aligned} \mathcal{L}_{rec} = & -\frac{1}{|\mathcal{B}_u^+|} \sum_{i \in \mathcal{B}_u^+} \hat{y}_i^u \log y_i^u \\ & -\frac{1}{|\mathcal{I}/\mathcal{B}_u^+|} \sum_{i \in \mathcal{I}/\mathcal{B}_u^+} (1 - \hat{y}_i^u) \log(1 - y_i^u) \end{aligned} \quad (16)$$

where $\mathcal{B}_u^+$ denotes the set of items in the next basket following S^u , while $\mathcal{I}/\mathcal{B}_u^+$ denotes the set of items that are not in the next basket following S^u . And $\hat{y}_i^u$ denotes the label corresponding to y_i^u , with $\hat{y}_i^u$ being equal to 1 if item $i \in \mathcal{B}_u^+$ and 0 otherwise.

3.3.3 Distribution Alignment

It is obvious that the prediction inferred from the same historical interactions have the same target, i.e., the probability $y_i^{(u,B)}$ inferred at basket-level and the probability $y_i^{(u,I)}$ inferred at item-level both indicate the probability that item i is in the next basket following S^u . Therefore, we align the probability distributions of the two levels by the following constraint:

$$\mathcal{L}_{ali} = \frac{1}{|\mathcal{T}|} \sum_{\{u,i\} \in \mathcal{T}} (y_i^{(u,B)} - y_i^{(u,I)})^2 \quad (17)$$

where $\mathcal{T}$ denotes all the user-item pairs in the training set.

3.4 Graph-based Polar Contrastive Learning

In this section, we will introduce the novel graph-based augmentation approach for polar contrastive learning in detail. When it comes to contrastive learning, how to perform data augmentation and how to construct self-supervised signals are two key points. Hence, we will elaborate the approach by focusing on the two key points.

3.4.1 Basket Augmentation

The simple intention of data augmentation is to transform the original data by adding some noises while retaining most of the information, so as to help with the learning of high-quality representations. For basket b , we can distinguish its relevance to other items by:

$$r_{bi} = \mathbf{e}_b \cdot \mathbf{e}_i^T. \quad (18)$$

According to the relevance, we can find the item set $\mathcal{I}_b^{pos}$, which consists of the K items most relevant to basket b , as well as the item set $\mathcal{I}_b^{neg}$ that is least relevant. With the item set $\mathcal{I}_b^{pos}$ of each basket, the positive graph $\mathcal{G}^{pos}$ can be constructed by adding edges, which connect basket b and items $i \in \mathcal{I}_b^{pos}$, to the original graph $\mathcal{G}$. Take the positive graph $\mathcal{G}^{pos}$ that is in Fig. 2 for example, the size of item set $\mathcal{I}_b^{pos}$ of each basket is set to 1, e.g., $\{i_9\}$ is the item set corresponding to basket b_3 , so the edge connecting i_9 and b_3 is added, which is shown by a dotted line. Similarly, the negative graph $\mathcal{G}^{neg}$ can be constructed with the item set $\mathcal{I}_b^{neg}$ of each basket. Specifically, the weights of the additional edges are set to 1.

3.4.2 Polar Contrastive Learning

By iterative propagation and aggregation based on the positive graph $\mathcal{G}^{pos}$ and the negative graph $\mathcal{G}^{neg}$, i.e., a pair of polar augmented graphs, we can get the positive embedding $\mathbf{e}_b^{pos}$ and the negative embedding $\mathbf{e}_b^{neg}$ of basket b , respectively. The multiple positive representations $\{\mathbf{p}_1^u, \dots, \mathbf{p}_H^u\}$, as well as the multiple negative representations $\{\mathbf{n}_1^u, \dots, \mathbf{n}_H^u\}$ of sequence S^u at basket-level can be calculated with the embeddings. We concatenate the representations as follows:

$$\mathbf{p}^u = [\mathbf{p}_1^u, \dots, \mathbf{p}_H^u], \quad (19)$$

$$\mathbf{n}^u = [\mathbf{n}_1^u, \dots, \mathbf{n}_H^u] \quad (20)$$

which are regarded as the polar contrastive pair of basket sequence S^u for mining self-supervised signal.

Considering a mini-batch with z basket sequences, we can obtain z polar contrastive pairs. With the expectation that a polar contrastive pair from the same basket sequence will be more consistent with each other than that from different basket sequences, the objective of polar contrastive learning is defined as follows:

$$\mathcal{L}_{cl}(\mathbf{p}^u, \mathbf{n}^u) = -\log \frac{\exp(\mathbf{p}^u \cdot \mathbf{n}^{uT})}{\exp(\mathbf{p}^u \cdot \mathbf{n}^{uT}) + \sum_{\mathbf{c} \in \mathcal{C}^-} \exp(\mathbf{p}^u \cdot \mathbf{c}^T)} \quad (21)$$

where $\mathcal{C}^-$ denotes the set of representations of sequences in the polar pairs except $\mathbf{p}^u$ and $\mathbf{n}^u$.

3.5 Multi-task Learning

Since the polar contrastive learning is introduced for helping next-basket recommendation task learn high-quality representations, we optimize them jointly by a multi-task learning strategy as follows:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{rec} + \lambda_2 \mathcal{L}_{ali} + \lambda_3 \mathcal{L}_{cl} + \lambda_4 \|\Phi\|^2 \quad (22)$$

where λ_1 , λ_2 and λ_3 denote the trade-off weights of the losses, Φ denotes the learnable parameters of HapCL, and $\|\cdot\|^2$ denotes L2 regularization with λ_4 being the regularization coefficient.

4 EXPERIMENTS

In this section, we conduct extensive experiments on three real-world datasets to evaluate the effectiveness of the proposed HapCL method. The experiments are designed to answer the following research questions:

- RQ1. How does the proposed HapCL method perform compared to the state-of-the-art baselines in the next-basket recommendation task?
- RQ2. Whether the different components of the HapCL method benefit the performance?
- RQ3. Can the polar contrastive pairs provide stronger self-supervised signals for the HapCL method?
- RQ4. How do the key hyper-parameters, i.e., the number of additional edges in augmentation K , the number of graph convolution layers L and the number of heads in multi-head H , affect the performance?

TABLE 2
Dataset statistics.

Dataset	Users	Items	Baskets	Average size of baskets	Average length of basket sequences
Beauty	5818	23018	20329	4.12	3.89
Grocery	6007	17941	22824	4.05	4.36
Tafeng	12814	11208	79625	6.41	7

4.1 Experiments Settings

4.1.1 Datasets

We adopt three real-world datasets in our experiments, namely Beauty¹, Grocery² and Tafeng³. Beauty and Grocery consist of the interactions of subcategory “Beauty” and subcategory “Grocery” on Amazon, which is a famous e-commercial platform, respectively. And Tafeng contains the transaction data of a Chinese grocery store. Following [3], which is a well-known next-basket recommendation method, the users and the items with less than 10 interaction records are discarded for all datasets. We generate a basket sequence for each user by sort the corresponding interactions according to the timestamp, and filter out the sequences with length fewer than 3. The statistics of the three datasets after preprocessing are summarized in Table 2. For each dataset, we divide the basket sequences into training set, validation set, and testing set in the ratio of 6:2:2.

4.1.2 Baselines

To verify the effectiveness of the HapCL method, we compare it against the following eight baselines:

- **POP** is a non-personalized method which recommends items with the greatest number of the historical interactions for each user.
- **GRU4Rec** [31] introduces recurrent neural network to model sequential information hidden in the historical interactions.
- **STAMP** [32] is a short-term memory priority model, which captures general interests of users from the long-term memory of a session context, and captures current interests from the short-term memory.
- **SASRec** [16] is a self-attention based model, which can not only capture long-term semantics, but also make prediction based on relatively few actions with an attention mechanism.
- **NextItNet** [33] consists of a stack of holed convolutional layers, and it can learn representation from both short-range and long-range item dependencies.
- **LightSANs** [34] extracts a constant number of latent interests by the low-rank decomposed self-attention, and generates the context-aware representation by making use of item-to-interest interaction.
- **CLEA** [4] designs a denoising generator to extract items relevant to the target item automatically, and proposes a two-stage anchor-guided contrastive learning to guide relevance learning.

- **Beacon** [3] encodes baskets with a correlation matrix to take into account the relative importance of items and correlations among item pairs.
- **SINE** [35] attempts to infer the set of concepts for each user adaptively, and predict current intention of users.
- **GCSAN** [1] is a graph-based method which utilizes graph neural network to capture local dependencies and self-attention mechanism to learn long-range dependencies for prediction.

These baselines can be divided into three groups based on the type of technology involved: (1) the traditional methods that do not consider sequence relationships, i.e., POP; (2) the methods focusing on sequential pattern, i.e., GRU4Rec, STAMP, SASRec, NextItNet, LightSANs, CLEA and Beacon; (3) the methods that introduce graph structure to facilitate sequence modeling, i.e., SINE and GCSAN. It should be noted that the models which require user profile are not selected for the experiments, because the users in the next-basket recommendation task are anonymous.

4.1.3 Evaluation Metrics

The baskets of the same user are spliced into a basket sequence in chronological order for evaluation. Following [16], we adopt the leave-one-out strategy to evaluate the performance of the methods, which will provide each user with M items as a basket for evaluation. We employ the widely used F1-score@ M ($F1@M$) [36], [37], [3] and Hit Ratio@ M ($HR@M$) [10], [38], [25] as evaluation metrics in our experiments where M is set to 5 and 30.

4.1.4 Parameter Settings

For a fair comparison, we initialize all embedding parameters randomly in the range of (0, 1), and optimize them by Adam optimizer with learning rate of $1e - 2$. The mini-batch size and embedding size are set to 32. For the key hyper-parameters of the HapCL method, we tune the number of additional edges in augmentation K and the number of graph convolution layers L within the range of $\{1, 2, 3, 4, 5\}$, the number of heads in multi-head H within the range of $\{2, 3, 4, 5, 6\}$. All the parameters are tuned on the validation set with early stopping whose patience is set to 10 epochs, and the results on the testing set are reported.

4.2 Overall Performance Comparison

The experimental results of different methods are summarized in Table 3. Here, we have the following observations:

- It is obvious that the proposed HapCL method shows competitive performance against the baselines on the three real-world datasets (RQ1). Although SINE and GCSAN both introduce graph structure to facilitate sequence modeling, SINE shows little change in performance among the three datasets, while GCSAN achieves relatively competitive performance. This indicates that the mining of multiple patterns can not guarantee the improvement of performance. We need to combine different patterns carefully so as to enhance the data mining ability of the method and achieve better performance together.
- Making use of the order of the historical interactions would help the methods mine more information from the limited historical interactions, and facilitate the modeling of relationship between baskets and items. The methods focusing on sequential pattern and the methods that

1. <https://www.kaggle.com/skillsmugger/amazon-ratings>
2. <https://github.com/RUCAIBox/RecSysDatasets>
3. <https://www.kaggle.com/chiranjivdas09/ta-feng-grocery-dataset>

TABLE 3

Overall Performance comparison of HapCL against the baselines. The best results obtained by baselines are underlined, and the results obtained by the proposed method are in bold. The performance improvement (Improv.) is defined as $\text{Improv.} = \frac{\text{Our result} - \text{The best baseline result}}{\text{The best baseline result}}$.

Dataset	Beauty				Grocery				Tafeng			
Metrics	F1@5	F1@30	HR@5	HR@30	F1@5	F1@30	HR@5	HR@30	F1@5	F1@30	HR@5	HR@30
POP	0.0008	0.0037	0.0020	0.0172	0.0057	0.0051	0.0081	0.0262	0.0307	0.0243	0.0270	0.0744
GRU4Rec	0.0055	0.0127	0.0095	0.0600	0.0166	0.0192	0.0233	0.0966	0.0352	0.0246	0.0301	0.0748
STAMP	0.0073	<u>0.0131</u>	<u>0.0101</u>	<u>0.0610</u>	0.0154	0.0175	0.0214	0.0876	0.0358	0.0247	0.0301	0.0746
SASRec	0.0049	0.0109	<u>0.0101</u>	0.0519	0.0157	<u>0.01987</u>	0.0205	<u>0.0990</u>	0.0344	0.0243	0.0278	0.0738
NextItNet	0.0064	0.0116	0.0095	0.0546	0.0163	0.0176	0.0209	0.0883	<u>0.0370</u>	0.0221	0.0299	0.0667
LightSANS	0.0049	0.0123	0.0091	0.0584	0.0158	0.0181	0.0219	0.0909	0.0330	<u>0.0256</u>	0.0293	<u>0.0779</u>
CLEA	0.0051	0.0116	0.0049	0.0274	0.0122	0.0167	0.0082	0.0417	0.0333	0.0245	0.0150	0.0369
Beacon	0.0065	0.0110	0.0093	0.0485	<u>0.0196</u>	0.0178	<u>0.0264</u>	0.0899	0.0336	0.0213	0.0298	0.0645
SINE	<u>0.0083</u>	0.0086	<u>0.0101</u>	0.0384	0.0094	0.0078	0.0133	0.0400	0.0086	0.0105	0.0072	0.0321
GCSAN	0.0060	0.0111	0.0087	0.0522	0.0144	0.0187	0.0186	0.0940	0.0355	0.0254	<u>0.0306</u>	0.0772
HapCL	0.0094	0.0133	0.0131	0.0619	0.0195	0.01993	0.0262	0.1004	0.0385	0.0264	0.0312	0.0794
Improv.	11.41%	1.69%	22.73%	1.60%	-0.5%	0.31%	-0.7%	1.42%	4.06%	2.91%	3.55%	1.94%

introduce graph structure show better performance than the traditional method that ignores time information of the historical interactions. Specifically, POP achieves the worst performance due to underutilization of data.

- The performance improvement tends to be much larger on Beauty than that on Grocery and Tafeng. And Beauty is a relatively sparse dataset, while Grocery and Tafeng are relatively dense. The results imply that the proposed method could alleviate the issue of data sparsity. Note that the Tafeng dataset contains the largest average size of baskets, the largest average length of basket sequences and the fewest items among the three datasets. And all the methods tend to show the best performance on the Tafeng dataset, while Beauty at the other extreme, which confirms that data sparsity would lead to poor performance.

4.3 Further Analysis

4.3.1 Ablation Study

We design the ablation study to validate the effectiveness of several major components of the proposed HapCL method. Since the part of hierarchical alignment and the part of polar contrastive learning are the two most important components of HapCL, we decide to make comparison with the following variants:

- HapCL^{-cl}: We remove the part of polar contrastive learning on the basis of HapCL to examine its validity. The embeddings of the baskets and the items will be obtained by the iterative propagation and aggregation based on the original weighted bipartite graph, and the positive graph and the negative graph will not be constructed for the mining of the self-supervised signals.
- HapCL^{-ali}: We remove the part of distribution alignment on the basis of HapCL to examine its validity. The predictions of the probability at basket-level and the predictions of the probability at item-level will not be required to be close to each other, but will be merged into the final predictions directly.

TABLE 4
Performance comparison of HapCL and its two variants.

Dataset	Method	F1@5	F1@30	HR@5	HR@30
Beauty	HapCL	0.0094	0.0133	0.0131	0.0619
	HapCL ^{-cl}	0.0062	0.0115	0.0079	0.0522
	Improv.1	33.80%	13.46%	39.39%	15.65%
	HapCL ^{-ali}	0.0057	0.0107	0.0087	0.0483
Grocery	Improv.2	19.53%	33.33%	22.04%	19.73%
	HapCL	0.0195	0.0199	0.0262	0.1004
	HapCL ^{-cl}	0.0175	0.0166	0.0243	0.0835
	Improv.1	10.41%	16.76%	7.27%	16.82%
Tafeng	HapCL ^{-ali}	0.0157	0.0197	0.0214	0.0987
	Improv.2	19.73%	1.15%	18.18%	1.66%
	HapCL	0.0385	0.0264	0.0312	0.0794
	HapCL ^{-cl}	0.0326	0.0235	0.0275	0.0706
	Improv.1	15.53%	10.94%	11.83%	11.09%
	HapCL ^{-ali}	0.0343	0.0246	0.0297	0.0741
	Improv.2	10.99%	6.72%	4.73%	6.67%

The results obtained by HapCL and its two variants on the three datasets are reported in Table 4.

We can see the performance degradation in different degrees of the variants, which validates the effectiveness of the two components (RQ2). Moreover, on the Beauty dataset, whose data is relatively sparse among the three datasets, the absence of either of the two important components, i.e., the part on distribution alignment and the part on polar contrastive learning, tends to have a greater impact on the performance of the HapCL method. This indicates that both components enhance mining information to

TABLE 5
Performance comparison of different contrastive types.

Dataset	Contrastive type	Metrics			
		F1@5	F1@30	HR@5	HR@30
Beauty	Ori-Pos	0.0073	0.0123	0.0095	0.0564
	Ori-Neg	0.0074	0.0128	0.0109	0.0604
	Pos-Neg	0.0094	0.0133	0.0131	0.0619
Grocery	Ori-Pos	0.0166	0.0191	0.0224	0.0961
	Ori-Neg	0.0178	0.0189	0.0243	0.0949
	Pos-Neg	0.0195	0.0199	0.0262	0.1004
Tafeng	Ori-Pos	0.0318	0.0242	0.0269	0.0733
	Ori-Neg	0.0346	0.0262	0.0293	0.0793
	Pos-Neg	0.0385	0.0264	0.0312	0.0794

help with the learning of better representations, thus improving the performance.

The addition of the part on the polar contrastive learning makes the HapCL method achieve performance improvement up to 39.39%. This suggests that the polar contrastive pairs indeed provide useful self-supervised signals, and thus enable HapCL to mine representative patterns and learn high-quality representations. In order to further verify the superiority of polar contrastive pairs, we design the other two types of contrastive pairs for experiments, and the experimental results are reported in Table 5. The contrastive type ‘‘Ori-Pos’’ denotes that self-supervised signals are mined based on the original graph and the positive graph, the contrastive type ‘‘Ori-Neg’’ denotes that self-supervised signals are mined based on the original graph and the negative graph, and the contrastive type ‘‘Pos-Neg’’ denotes that self-supervised signals are mined based on the positive graph and the negative graph. From the results of the three different contrastive types, we can find that the polar contrast, i.e., the contrastive type ‘‘Pos-Neg’’, shows overwhelming superiority over the other two contrastive types in all metrics on the three datasets. The results confirm the hypothesis that compared with non-polar contrastive types, i.e., the contrastive type ‘‘Ori-Pos’’ and the contrastive type ‘‘Ori-Neg’’, the polar contrastive pairs will be more conducive to mine representative patterns (RQ3). It is worth noting that the performance of the contrastive type ‘‘Ori-Pos’’ is generally worse than that of the contrastive type ‘‘Ori-Neg’’, suggesting that telling the method what is negative information is more important than telling the method what is positive information.

On the three datasets, the performance of the HapCL method degrades by up to 33.33% due to the absence of the part of the distribution alignment. This indicates that the distribution alignment allows the predictions of probability at different levels, i.e., basket level and item level, to guide each other so as to switch to the right direction. The finding will be further verified in Section 4.3.2.

4.3.2 Hyper-parameter Analysis

The number of additional edges in augmentation K , the number of graph convolution layers L and the number of heads in multi-head H are the three key hyper-parameters of the HapCL method.

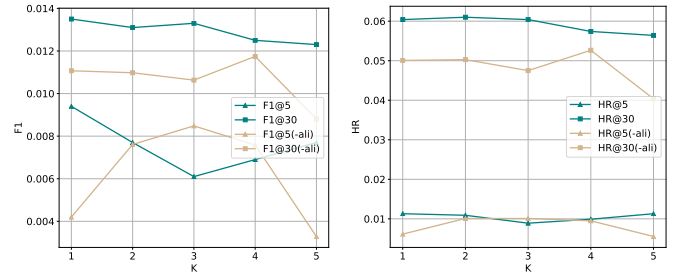


Fig. 4. The performance of HapCL and HapCL-*ali* with different numbers of additional edges in augmentation, which is denoted by K , on Beauty. The left subfigure corresponds to F1@ M and the right subfigure corresponds to HR@ M . The K varies in $\{1, 2, 3, 4, 5\}$. The lines in dark green denote the results of HapCL and the lines in khaki denote the results of HapCL-*ali*.

We will analyze the impact of the hyper-parameters on the performance of HapCL in this section (RQ4).

In the part of basket augmentation, the HapCL method tries to construct polar contrastive pairs by adding new edges to connect baskets and items. By fixing the number of graph convolution layers L at 1 and the number of heads in multi-head H at 4, we test the effect of the number of additional edges in augmentation K within the range of $\{1, 2, 3, 4, 5\}$. The performance of the HapCL method with different K is plotted in Fig. 4. The lines in dark green denotes the results of HapCL. It can be observed that the HapCL method tends to show the best performance when K is 1, and when K is larger than 1, the performance decreases. Moreover, we can see that more stable performance is achieved when K is fewer than 3. Considering that the average basket size of Beauty is 4.12, it suggests that there should not be too many additional edges, as too many new edges would obscure the extraction of relationship between basket and the corresponding items.

Graph convolution is responsible for the information propagation and aggregation among the nodes of the weight bipartite graph. The more layers of graph convolution, the more higher-order information would be contained in the embedding of each node. By fixing the number of the additional edges K at 2 and the number of heads in multi-head H at 4, we test the effect of the number of graph convolution layers L within the range of $\{1, 2, 3, 4, 5\}$. The performance of the HapCL method with various L is plotted in Fig. 5. The lines in violet denotes the results of HapCL. The performance of the HapCL method reaches the optimal value when L is 2, and decreases when the number of graph convolution layers is larger than or smaller than 2. This makes sense because too few layers lead to insufficient mining of the graph pattern, while too many layers make too much noise that disrupts the extraction of the information.

The diversity of the views on information mining is dominated by the number of heads in multi-head H . By fixing the number of the additional edges K at 2 and the number of graph convolution layers L at 1, we test H within the range of $\{2, 3, 4, 5, 6\}$. The performance of the HapCL method with various H is plotted in Fig. 6. The lines in turquoise denotes the results of HapCL. The performance of HapCL fluctuates, and it is relatively good when the value of H is 2 and 5. Since the meaning of a view is not artificially defined, but is divided by HapCL itself based on a given number of views, the model will show good performance when the granularity of decomposition is appropriate

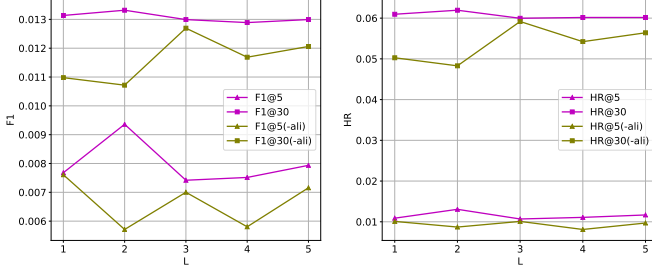


Fig. 5. The performance of HapCL and HapCL^{-ali} with different numbers of graph convolution layers, which is denoted by L , on Beauty. The left subfigure corresponds to F1@M and the right subfigure corresponds to HR@M. The L varies in $\{1, 2, 3, 4, 5\}$. The lines in violet denote the results of HapCL and the lines in olive drab denote the results of HapCL^{-ali}.

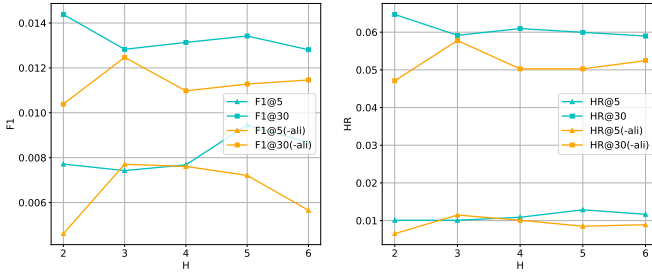


Fig. 6. The performance of HapCL and HapCL^{-ali} with different numbers of heads in multi-head, which is denoted by H , on Beauty. The left subfigure corresponds to F1@M and the right subfigure corresponds to HR@M. The H varies in $\{2, 3, 4, 5, 6\}$. The lines in turquoise denote the results of HapCL and the lines in orange denote the results of HapCL^{-ali}.

for the information contained in the data, while an inappropriate granularity of decomposition will harm the model's performance.

In addition, the performance of HapCL^{-ali} with the different number of additional edges K , the different number of graph convolution layers L and the different number of heads H are also plotted in Fig. 4, Fig. 5 and Fig. 6 with the color of khaki, olive and orange, respectively. It is clear that most of the results of HapCL^{-ali} are worse than the corresponding results of HapCL, where the same variation of the hyper-parameters would lead to greater fluctuation in the performance of HapCL^{-ali}. The above analysis further verifies that the distribution alignment can not only achieve mutual supervision and improve the performance, but also enhance the stability of the method.

4.3.3 Attention Mechanism Analysis

The attention mechanism is introduced in hierarchical multi-head predictor to boost the interchange of information among different views. While Fig. 6 shows the impact of the numbers of heads in multi-head, we further investigate the effectiveness of the introduced attention mechanism (RQ2) in this part. Two anonymous users are selected from the testing set of Beauty randomly, and their historical interactions are shown in the form of basket sequence in Fig. 7, where the items are represented by IDs. The basket sequence length of the first user is 2 and that of the second user is 4. We set the numbers of heads in multi-head H to 4, so as to obtain an attention weight matrix within size 4×4 at basket-level and item-level, respectively. The matrices are displayed by heatmaps in Fig. 8 for a more intuitive observation.

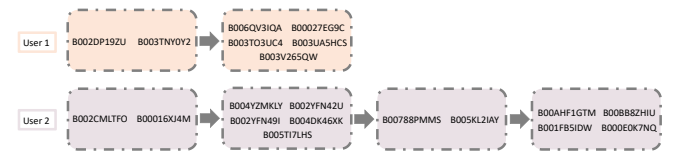


Fig. 7. The basket sequences of two anonymous users selected from Beauty.

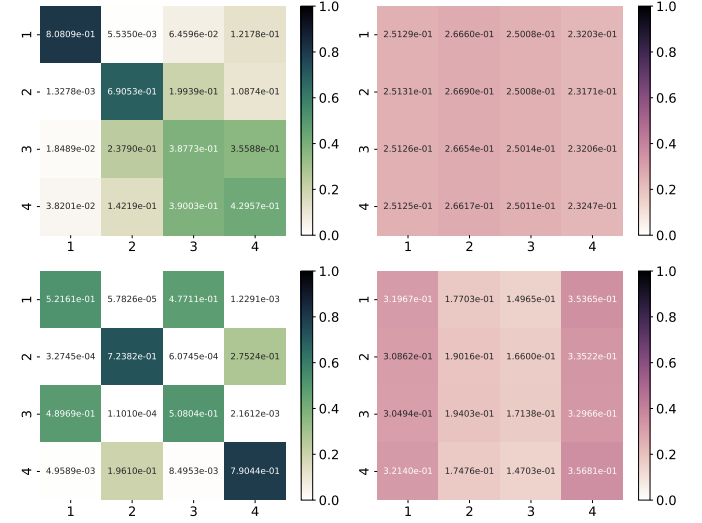


Fig. 8. Heatmaps of attention weights of two anonymous user across different views on Beauty. The upper left subfigure corresponds to the weights of the first user at basket-level, the upper right subfigure corresponds to the weights of the first user at item-level, the lower left subfigure corresponds to the weights of the second user at basket-level and the lower right subfigure corresponds to the weights of the second user at item-level. The x-axis and y-axis, which vary within the range of $\{1, 2, 3, 4\}$, denote the serial number of views. The darker color of the cells represents the higher weight and the specific values are marked in the corresponding cells.

The cell in the i -th row and j -th column denotes the attention weight of the i -th view over the j -th view. Note that the attention of the i -th view over the j -th view and the attention of the j -th view over the i -th view are not equivalent. We can find that the views tend to pay more attention to themselves at basket-level, while the views at item-level tend to obtain similar weight from different views. This phenomenon indicates diverse characteristics of different formats of the same historical interactions, and the baskets that aggregate the information of items can generate the representations with more discrimination. The combination of sequential patterns at different levels is beneficial to the utilization of the limited historical interactions.

5 CONCLUSION

Since sparse data impedes the performance of recommendation methods, it is necessary to facilitate the mining of the limited historical interactions. The existing next-basket recommendation methods mostly neglect the fact that the information of the historical interactions is hidden in multiple views, and treat the next-basket recommendation task as a single-view sequential prediction problem. In this paper, we propose a novel method that combines hierarchical alignment with polar contrastive learning for next-basket recommendation (named HapCL). The historical interac-

tions are modeled by a weight bipartite graph for information propagation and aggregation. And a module with hierarchical framework is designed to mine information from multiple views at two levels, i.e., basket-level and item-level. The probabilities inferred at the two levels are aligned and integrated into the final prediction. In order to mine self-supervised signals from the original data, we design a novel graph-based augmentation approach for constructing polar contrastive pairs. Extensive experiments on three real-world datasets validate the effectiveness of the proposed HapCL method in mining more information for the next-basket recommendation task.

ACKNOWLEDGMENTS

The work was supported by National Key Research and Development Program of China (2021YFF1201200), NSFC (62276277), Guangdong Basic and Applied Basic Research Foundation (2022B1515120059), and Guangdong Province Key Laboratory of Computational Science at the Sun Yat-sen University (2020B1212060032).

REFERENCES

- [1] C. Xu, P. Zhao, Y. Liu, V. S. Sheng, J. Xu, F. Zhuang, J. Fang, and X. Zhou, "Graph contextualized self-attention network for session-based recommendation," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, 2019, pp. 3940–3946.
- [2] C. Wang, W. Xi, L. Huang, Y. Zheng, Z. Hu, and J. Lai, "A BP neural network based recommender framework with attention mechanism," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 7, pp. 3029–3043, 2022.
- [3] D. Le, H. W. Lauw, and Y. Fang, "Correlation-sensitive next-basket recommendation," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, 2019, pp. 2808–2814.
- [4] Y. Qin, P. Wang, and C. Li, "The world is binary: Contrastive learning for denoising next basket recommendation," in *the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 859–868.
- [5] X. Wang, D. Wang, C. Xu, X. He, Y. Cao, and T. Chua, "Explainable reasoning over knowledge graphs for recommendation," in *the Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference*, 2019, pp. 5329–5336.
- [6] F. Yu, Q. Liu, S. Wu, L. Wang, and T. Tan, "A dynamic recurrent model for next basket recommendation," in *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, 2016, pp. 729–732.
- [7] X. Fan, Z. Liu, J. Lian, W. X. Zhao, X. Xie, and J. Wen, "Lighter and better: Low-rank decomposed self-attention networks for next-item recommendation," in *the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 1733–1737.
- [8] M. M. Tanjim, C. Su, E. Benjamin, D. Hu, L. Hong, and J. J. McAuley, "Attentive sequential models of latent intent for next item recommendation," in *The ACM Web Conference*, 2020, pp. 2528–2534.
- [9] Q. Cui, S. Wu, Q. Liu, W. Zhong, and L. Wang, "MV-RNN: A multi-view recurrent neural network for sequential recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 2, pp. 317–331, 2020.
- [10] K. Zhou, H. Wang, W. X. Zhao, Y. Zhu, S. Wang, F. Zhang, Z. Wang, and J. Wen, "S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization," in *the 29th ACM International Conference on Information and Knowledge Management*, 2020, pp. 1893–1902.
- [11] X. Zheng, M. Wang, R. Xu, J. Li, and Y. Wang, "Modeling dynamic missingness of implicit feedback for sequential recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 1, pp. 405–418, 2022.
- [12] Y. Chen, Z. Liu, J. Li, J. J. McAuley, and C. Xiong, "Intent contrastive learning for sequential recommendation," in *The ACM Web Conference*, 2022, pp. 2172–2182.
- [13] R. Qiu, Z. Huang, H. Yin, and Z. Wang, "Contrastive learning for representation degeneration problem in sequential recommendation," in *The Fifteenth ACM International Conference on Web Search and Data Mining*, 2022, pp. 813–823.
- [14] D. Yang, C. Shen, B. Liu, L. Xue, and Y. Xiao, "Generating knowledge-based attentive user representations for sparse interaction recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 9, pp. 4270–4284, 2022.
- [15] P. Zhao, A. Luo, Y. Liu, J. Xu, Z. Li, F. Zhuang, V. S. Sheng, and X. Zhou, "Where to go next: A spatio-temporal gated network for next POI recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 5, pp. 2512–2524, 2022.
- [16] W. Kang and J. J. McAuley, "Self-attentive sequential recommendation," in *IEEE International Conference on Data Mining*, 2018, pp. 197–206.
- [17] Y. Wu, K. Li, G. Zhao, and X. Qian, "Personalized long- and short-term preference learning for next POI recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 4, pp. 1944–1957, 2022.
- [18] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: online learning of social representations," in *the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 701–710.
- [19] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, "Graph convolutional neural networks for web-scale recommender systems," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Y. Guo and F. Farooq, Eds., 2018, pp. 974–983.
- [20] 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*, 2020, pp. 639–648.
- [21] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *the 6th International Conference on Learning Representations*, 2018.
- [22] Z. Ren, S. Liang, P. Li, S. Wang, and M. de Rijke, "Social collaborative viewpoint regression with explainable recommendations," in *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, 2017, pp. 485–494.
- [23] W. Chen, Y. Gu, Z. Ren, X. He, H. Xie, T. Guo, D. Yin, and Y. Zhang, "Semi-supervised user profiling with heterogeneous graph attention networks," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, 2019, pp. 2116–2122.
- [24] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "Factorizing personalized markov chains for next-basket recommendation," in *Proceedings of the 19th International Conference on World Wide Web*, 2010, pp. 811–820.
- [25] Z. Liu, M. Wan, S. Guo, K. Achan, and P. S. Yu, "Basconv: Aggregating heterogeneous interactions for basket recommendation with graph convolutional neural network," in *Proceedings of the 2020 SIAM International Conference on Data Mining*, 2020, pp. 64–72.
- [26] S. Ge, S. Mishra, C. Li, H. Wang, and D. Jacobs, "Robust contrastive learning using negative samples with diminished semantics," in *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021*, 2021, pp. 27 356–27 368.
- [27] P. Li, D. Li, W. Li, S. Gong, Y. Fu, and T. M. Hospedales, "A simple feature augmentation for domain generalization," in *IEEE/CVF International Conference on Computer Vision*, 2021, pp. 8866–8875.
- [28] K. Cho, B. van Merriënboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, A. Moschitti, B. Pang, and W. Daelemans, Eds., 2014, pp. 1724–1734.
- [29] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Annual Conference on Neural Information Processing Systems*, 2017, pp. 5998–6008.
- [31] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," in *the 4th International Conference on Learning Representations*, 2016.
- [32] Q. Liu, Y. Zeng, R. Mokhosi, and H. Zhang, "STAMP: short-term attention/memory priority model for session-based recommendation," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 1831–1839.
- [33] F. Yuan, A. Karatzoglou, I. Arapakis, J. M. Jose, and X. He, "A simple convolutional generative network for next item recommendation," in *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 2019.
- [34] X. Fan, Z. Liu, J. Lian, W. X. Zhao, X. Xie, and J. Wen, "Lighter and better: Low-rank decomposed self-attention networks for next-item recommendation," in *the 44th International ACM SIGIR Conference on*

Research and Development in Information Retrieval, 2021, pp. 1733–1737.

- [35] Q. Tan, J. Zhang, J. Yao, N. Liu, J. Zhou, H. Yang, and X. Hu, “Sparse-interest network for sequential recommendation,” in *the Fourteenth ACM International Conference on Web Search and Data Mining*, 2021, pp. 598–606.
- [36] Y. Du, H. Liu, and Z. Wu, “Modeling multi-factor and multi-faceted preferences over sequential networks for next item recommendation,” in *Machine Learning and Knowledge Discovery in Databases*, vol. 12976, 2021, pp. 516–531.
- [37] J. Li, Z. Li, J. Huang, J. Zhang, X. Wang, X. Lu, and J. Zhou, “Large-scale fake click detection for e-commerce recommendation systems,” in *the 37th IEEE International Conference on Data Engineering*, 2021, pp. 2595–2606.
- [38] D. Cao, X. He, L. Miao, G. Xiao, H. Chen, and J. Xu, “Social-enhanced attentive group recommendation,” *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 3, pp. 1195–1209, 2021.



Ting-Ting Su received her bachelor's degree in computer science in 2021 from China University of Mining and Technology. She is currently pursuing the master's degree at Sun Yat-sen University. Her current research interest is data mining.



Wu-Dong Xi received B.S. degree in 2019 and his Master degree in 2021 respectively in computer science from Sun Yat-sen University, Guangzhou, China. He joined UX center, NetEase Games in 2021 as Data Mining Researcher. He has published several papers in IJCAI, IEEE TNNLS, IEEE TKDE, IEEE TSMC-Systems. His research interest is data mining, in particular, recommender systems.



Chang-Dong Wang received the Ph.D. degree in computer science in 2013 from Sun Yat-sen University, Guangzhou, China. He is a visiting student at University of Illinois at Chicago from Jan. 2012 to Nov. 2012. He joined Sun Yat-sen University in 2013, where he is currently an associate professor with School of Computer Science and Engineering. His current research interests include machine learning and data mining. He has published over 100 scientific papers in international journals and conferences

such as IEEE TPAMI, IEEE TKDE, IEEE TCYB, IEEE TNNLS, KDD, AAAI, and IJCAI. His ICDM 2010 paper won the Honorable Mention for Best Research Paper Awards. He was awarded 2015 Chinese Association for Artificial Intelligence (CAAI) Outstanding Dissertation. He is an Associate Editor in Journal of Artificial Intelligence Research.



Jian-Huang Lai received the M.Sc. degree in applied mathematics in 1989 and the Ph.D. degree in mathematics in 1999 from Sun Yat-sen University, China. He joined Sun Yat-sen University in 1989 as an Assistant Professor, where he is currently a Professor with the School of Computer Science and Engineering. His current research interests include the areas of digital image processing, pattern recognition, multimedia communication, wavelet and its applications. He has published more than 200 scientific papers

in the international journals and conferences on image processing and pattern recognition, such as IEEE TPAMI, IEEE TKDE, IEEE TNNLS, IEEE TIP, IEEE TSMC-B, Pattern Recognition, ICCV, CVPR, IJCAI, ICDM and SDM. Prof. Lai serves as a Standing Member of the Image and Graphics Association of China, and also serves as a Standing Director of the Image and Graphics Association of Guangdong.



Philip S. Yu is a Distinguished Professor in Computer Science at the University of Illinois at Chicago and also holds the Wexler Chair in Information Technology. Before joining UIC, Dr. Yu was with IBM, where he was manager of the Software Tools and Techniques group at the Watson Research Center. His research interest is on big data, including data mining, data stream, database and privacy. He has published more than 1,000 papers in refereed journals and conferences. He holds or has applied for more

than 300 US patents.

Dr. Yu is a Fellow of the ACM and the IEEE. He is on the steering committee of the ACM Conference on Information and Knowledge Management and was a member of the steering committee of IEEE Data Engineering and IEEE Conference on Data Mining. He was the Editor-in-Chief of ACM Transactions on Knowledge Discovery from Data (2011-2017) and the Editor-in-Chief of IEEE Transactions on Knowledge and Data Engineering (2001-2004). Dr. Yu is the recipient of ACM SIGKDD 2016 Innovation Award for his influential research and scientific contributions on mining, fusion and anonymization of big data, the IEEE Computer Society's 2013 Technical Achievement Award for “*pioneering and fundamentally innovative contributions to the scalable indexing, querying, searching, mining and anonymization of big data*”, and the Research Contributions Award from IEEE Intl. Conference on Data Mining (ICDM) in 2003 for his pioneering contributions to the field of data mining. He also received the ICDM 2013 10-year Highest-Impact Paper Award, and the EDBT Test of Time Award (2014). He had received several IBM honors including 2 IBM Outstanding Innovation Awards, an Outstanding Technical Achievement Award, 2 Research Division Awards and the 94th plateau of Invention Achievement Awards. He was an IBM Master Inventor. Dr. Yu received the B.S. Degree in E.E. from National Taiwan University, the M.S. and Ph.D. degrees in E.E. from Stanford University, and the M.B.A. degree from New York University.