

Received November 24, 2020, accepted December 10, 2020, date of publication December 15, 2020, date of current version December 30, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3045092

A Hybrid-Preference Neural Model for Basket-Sensitive Item Recommendation

ZHIQIANG PAN¹, WANYU CHEN, AND HONGHUI CHEN

Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha 410073, China

Corresponding author: Wanyu Chen (panzhiqiang15@nudt.edu.cn)

This work was supported by the Postgraduate Scientific Research Innovation Project of Hunan Province under Grant CX20200055.

ABSTRACT Basket-Sensitive Item Recommendation (BSIR) is a challenging task that aims to recommend an item to add to the current basket given a user's historical behaviors. The recommended item is supposed to be relevant to the items in current basket. Previous works mainly produce a recommendation based on user's current basket, ignoring the inherent preference released by user's long-term behaviors and failing to accurately distinguish the item importance in the basket for detecting user intent. To tackle the above issues, we propose a hybrid model, i.e., Hybrid-Preference Neural Model (HPNM), where a user's inherent preference is recognized by modeling the historical sequential baskets and the recent preference is identified by focusing on the current basket. In detail, we apply an attention mechanism for distinguishing the importance of items in a basket to generate an accurate basket representation. GRU is utilized for modeling the basket-level sequential information to obtain user's long-term preference and the representation of the current session is regarded as user's short-term preference. We evaluate the performance of our proposals against the state-of-the-art baselines in the field of BSIR on two public datasets, i.e., TaFeng and Foursquare. The experimental results show that HPNM can achieve obvious improvements against the baselines in terms of HLU and Recall. In addition, we find HPNM with an attention mechanism can lead to a larger improvement against the baseline for item recommendation in terms of HLU and Recall on testing baskets with relatively fewer items.

INDEX TERMS Basket-sensitive item recommendation, sequential recommendation, recurrent neural networks, attention mechanism.

I. INTRODUCTION

Recommender systems can help people address personalized need out of overloaded information on the internet [1], [2], which are typically implemented by modeling user's historical behaviors [3]–[5]. As we know, user behaviors can be often collected from user's interactions with a set of items rather than a single item at a timestamp [6]–[8]. For instance, people would like to buy more than one product in a purchase event or visit more than one place in a day during the journey. This set of items (i.e., products or places) can be regarded as a basket [9], [10], where items in a basket are often regarded relevant to each other. Basket-sensitive item recommendation (BSIR) aims to produce a recommended item by considering user's interactions with items in the historical baskets and the current basket [9], which can represent user's inherent preference and instant preference, respectively. Here three

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Asif¹.

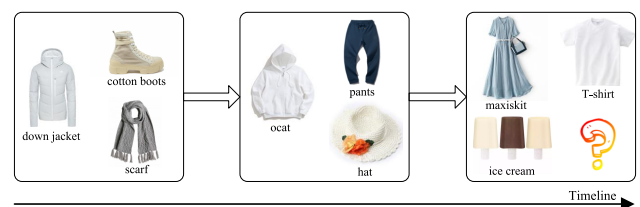


FIGURE 1. An example of basket-sensitive item recommendation.

factors are often considered, i.e., the relationships among items within a particular basket, the items in the current basket, and the basket-level sequential information in historical baskets.

Considering an example in Fig. 1, there is a user who likes purchasing clothes on the e-commerce platform, such as JD.com, Alibaba, etc. At each timestamp, the user may select several items together, i.e., a basket, which may due to

that there will be a discount if those products are consumed at the same time. Moreover, there may be some items that don't conform to user's main interest in the basket because of user's complex behavior pattern. For example, in Fig. 1, the user intends to buy some summer clothes in the current basket. However, during the interaction process, some summer snacks, e.g., the ice cream, may attract the user's attention and is then added into the basket together with these clothes. Thus, the importances of these items are different for predicting the user's main interest in current basket. Moreover, the user's interest from his historical sequential baskets (i.e., all the baskets in the timeline) and the current basket (i.e., the last basket) can represent user's inherent attributes (such as the female gender) and the instant needs (i.e., the purpose of purchasing summer clothes), respectively. Both of them are important for predicting the user's next purchase. Furthermore, the sequential information in historical baskets should also be emphasized, which can reflect the preference migration of the user (e.g., user preference from winter dress to summer clothes). By taking the basket-level sequential signals into consideration, user's dynamic demands can be captured.

Generally, the major approaches for BSIR can be categorized into two folds, i.e., the association rules based recommendation [11] and the factorization machine based recommendation [9]. Association rules based methods can discover the correlations that exist in a large amount of interactions. However, the generated rules are the same for all users. Although there exists "personalized" recommendation by using the rules that the user has interacted with the items on the left side [11], the personalized association is still not well learnt. Factorization machine based methods consider the relationship among users, items in current basket and the target item to be recommended. Commonly, a linear combination is applied to take all relationships into consideration. However, such approaches can not model the dynamic change of a user's preference. For instance, if a user bought a lot of winter clothes in winter. After the winter, the user wants to buy clothes for spring or summer. Using the factorization machine based approach, the winter clothes will be recommended again since the user has bought lots of winter clothes. Obviously, it is unreasonable to buy winter clothes in the summer time. Moreover, the linear combination will cause an issue of information loss [12], [13].

In summary, the main challenges in Basket-Sensitive Item Recommendation (BSIR) are:

- 1) How to mine user's interest in each basket by modeling the items interacted in the basket through distinguishing their different importance?
- 2) How to take both user's inherent preference and instant needs into consideration and meanwhile capture the dynamic interest migration in the historical baskets?

To solve the above issues, we introduce a Hybrid-Preference Neural Model (HPNM) for basket-sensitive item recommendation. First, in the item level, an attention mechanism is applied to distinguish the importance of items in

each basket to generate an accurate basket representation, which can partly solve the issue of information loss that exists in the process of the average pooling and the max pooling. Then, in the basket level, HPNM utilizes Gated Recurrent Units (GRU) to model a user's historical sequential baskets to generate the long-term preference. In addition, the representation of current basket is regarded as user's short-term preference, which is then combined with user's long-term preference as the final user preference for item prediction.

In general, the contributions of our work are summarized as follows: (1) We introduce a hybrid neural model for basket-sensitive item recommendation which can capture user's long-term and short-term preferences simultaneously. (2) We apply an attention mechanism to model the items in each basket to better distinguish the importance of items, solving the problem of information loss when representing the basket. (3) We conduct comprehensive experiments on the TaFeng and Foursquare datasets, finding that our proposal achieves the state-of-the-art performance, returning the target items at an earlier position in the recommendation list.

II. RELATED WORK

According to the number of items interacted by the user at each timestamp, there are item recommendation (one item at each timestamp) and basket-based recommendation (more than one item at each timestamp). In this section, we briefly introduce the previous works on item recommendation in Section II-A and basket-based recommendation in Section II-B.

A. ITEM RECOMMENDATION

General recommendation methods can obtain a user's inherent preference by modeling his historical interactions, where Collaborative Filtering (CF) is the most widely used approach which mainly utilizes Matrix Factorization (MF) to take the user-item interaction matrix into consideration. Recently, some neural methods are proposed to better model the interactions between users and items. He *et al.* [12] propose Neural Collaborative Filtering (NCF) to combine the generalized matrix factorization and a multi-layer perceptron to nonlinearly learn the user-item interactions. Then Chen *et al.* [13] propose a Joint Neural Collaborative Filtering (JNCF) to extract deep feature representations of users and items, and capture the deep interactions between users and items through the user-item interaction matrix. Furthermore, Wang *et al.* [14] utilize Graph Neural Networks (GNN) [15] to learn the high-order connectivities in the user-item integration graph by performing embedding propagation to explicitly encode the collaborative signal. However, general recommendation methods can merely focus on the inherent user preference, they often neglect the migration of user interests, thus failing to capture user's instant needs.

Different from general recommendation methods, sequential recommendation is proposed to model users' dynamic interest [16]–[18]. For example, Hidasi *et al.* [16] propose GRU4REC, which utilizes Gated Recurrent Units (GRU) to

model the sequential information in a session to obtain users' dynamic interests. Moreover, Li *et al.* [19] introduce attention mechanism into GRU4REC to capture users' main purpose. And Liu *et al.* [20] distinguish the item importance using an attention mechanism to obtain users' long-term preference and explicitly take the current action as the short-term interest to make predictions. Furthermore, Kang and McAuley [17] propose SASRec to introduce self-attention mechanism into sequential recommendation to identify the related items for predicting the next item. Considering the ability of graph neural networks for modeling the complex transitions between items [15], [21], GNNs have been widely applied in the sequential recommendation. For instance, Wu *et al.* [22] first convert each session into a session graph and then use Gated Graph Neural Networks (GGNN) [23] to obtain the accurate representation of each item. However, all the above sequential recommenders are designed for item recommendation, without considering the situation that a user may interact with a whole basket rather than a single item at each timestamp.

Generally, current item recommendation methods cannot be directly applied on basket-sensitive item recommendation because of the ignorance of sequential information or merely considering the interactions with a single item at each timestamp. In contrast, our proposed method can take both of those two factors into consideration to achieve considerable recommendation accuracy.

B. BASKET-BASED RECOMMENDATION

Basket-based recommendation has been well investigated in recent years. In this section, we divide the major basket-based recommendation approaches into two categories, i.e., basket-sensitive item recommendation (BSIR) and next basket recommendation (NBR), according to the difference of recommendation aim.

As for basket-sensitive item recommendation, the association rules can find the associations from historical data and can be employed in recommender systems [24], [25]. The typical association rule based methods, like FP-tree [26], mainly focus on the computational efficiency. In addition, these methods are not personalized. Further more, Sarwar *et al.* [11] apply association rules in recommendation and make it "personalized". However the association still has no difference for users. In addition, factorization machine has also been applied in BSIR. Le *et al.* [9] employ a factorization machine to consider the relations, e.g., user and target item, basket item and target item. Further more, it integrates the constraints for baskets with similar intent. So far, this method achieves the state-of-the-art performance in the task of BSIR [9].

Different from BSIR, Next Basket Recommendation (NBR) recommends a whole basket to a user given the historical behaviors [27]. Traditional methods like Markov Chains [28] and deep learning based methods [2], [29], [30], have been well studied. For instance, Rendle *et al.* [28] propose Factorizing Personalized Markov Chains (FPMC) for next basket recommendation, which takes both the

sequential information and user's general taste into consideration. Wang *et al.* [31] utilize the average and max pooling to aggregate the items in each basket and propose a hierarchical representation model (HRM) to represent user's preference. In addition, Yu *et al.* [32] apply Recurrent Neural Networks (RNN) in NBR and achieve an excellent performance, which verifies the effectiveness of RNN in NBR. Le *et al.* [33] propose to consider the relative importance of items when modeling the sequential information of baskets.

However, in the group of methods for aggregating the basket, the average pooling treats each item equally and the max pooling only takes the most significant one in every dimension, which results in an issue of information loss. In this paper, we propose to apply an attention mechanism to distinguish the importance of items in each basket. Moreover, we consider user's long-term preference for recommendation by modeling the whole historical baskets.

III. APPROACHES

In this section, we describe our Hybrid-Preference Neural Model (HPNM) for Basket-Sensitive Item Recommendation (BSIR). We first formally present the BSIR task in Section III-A. The major architecture of HPNM is shown in Fig. 2. In the first stage, the item embeddings in the basket are inputted to an aggregation layer to generate the basket representation (See Section III-B). Then, long-term and short-term preferences are modeled separately and combined to generate user preference representation (See Section III-C). Finally, the user preference is inputted to a similarity layer, together with all item embeddings, to calculate the probability scores of all potentially recommended items and make prediction (See Section III-D).

A. PROBLEM FORMULATION AND NOTATION

Assuming there are a set of users denoted by U and a set of items denoted by I , the number of users and items are $|U|$ and $|I|$, respectively. Each user is represented as $u \in U$, and each item is represented as $i \in I$, respectively. Given a user u , the historical baskets can be denoted as $S^u = \{S_1^u, S_2^u, \dots, S_n^u\}$, which includes user's current basket S_n^u . Intuitively, user u 's basket at timestamp t is denoted by S_t^u and consists of a set of m items, i.e., $S_t^u = \{x_{t1}, x_{t2}, \dots, x_{tm}\}$. Given user's interaction history, the purpose of BSIR is to recommend an item to add to the current basket S_n^u .

To begin with, for every item, we use an one-hot vector to represent it, represented as $e_i \in \mathbb{R}^{|I|}$ for item i . Specifically, for item i , the value at i -th position in the one-hot vector is 1, while in other positions they are all set to 0. After that, the one-hot vector will be inputted to an embedding layer to get its embedding representation. To be more specific, we create an item embedding matrix $V \in \mathbb{R}^{|I| \times d}$, and retrieve the corresponding vectors from the item embedding matrix for items as their representations, i.e., regarding the i -row in V as the representation of item i . After the embedding layer, the item i is represented as $v_i \in \mathbb{R}^d$, where d is the dimension of item embedding vector.

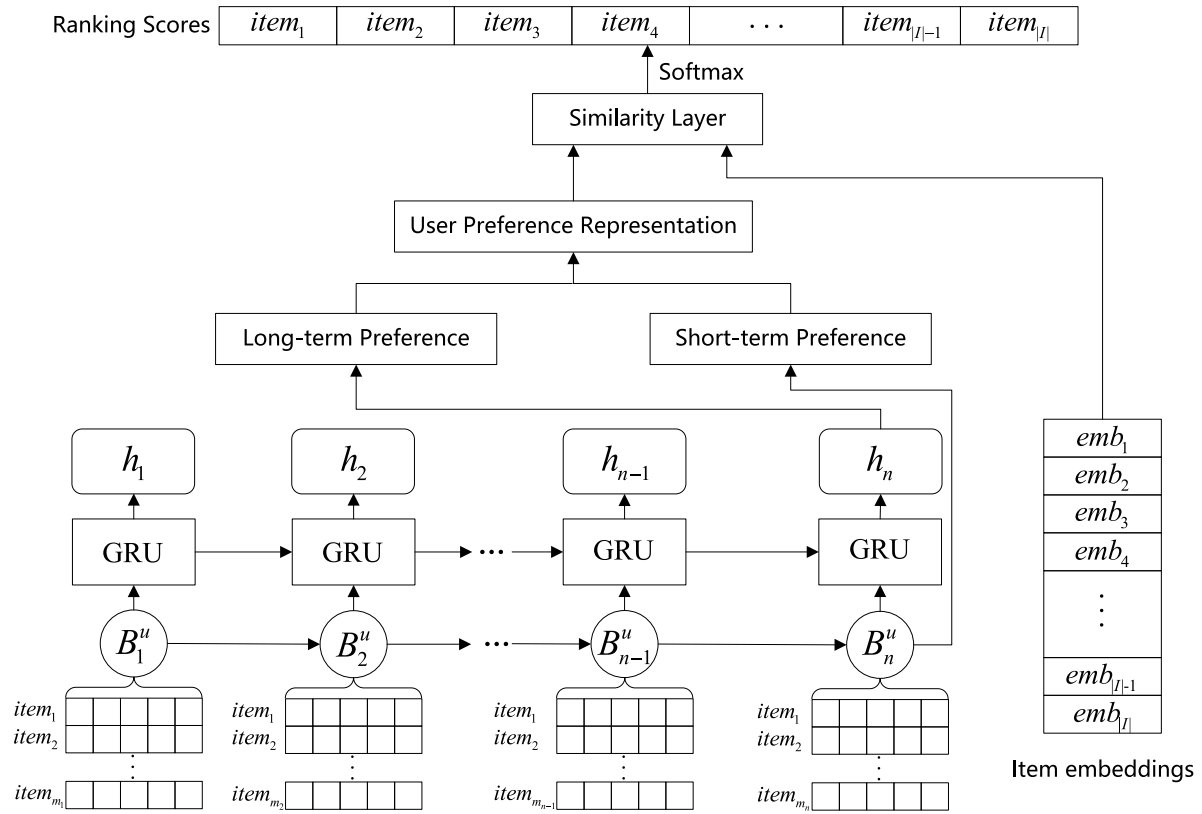


FIGURE 2. Structure of our hybrid-preference neural model.

B. AGGREGATION FOR BASKET REPRESENTATION

In recommender systems, the item prediction is commonly conducted by comparing the similarity of user representation with the embeddings of the candidate items [WWW17/NCF,IJCAI18/SHAN]. Specifically, in basket-sensitive item recommendation, the generation of user preference is hierarchical, which means that we need to provide the representation of the baskets for the user preference modeling. Moreover, before basket-level fusion, we need to firstly conduct the aggregation on the item embeddings to obtain basket representations.

To identify the importance of different items in a basket, we first input the item embeddings to an importance function. After generating the item importance, the item embeddings are combined according to the importance weight to vectorize the basket S_i^u as follows.

$$B_i^u = \sum_{i=1}^m \lambda_{ti} g(v_{ti}) = \sum_{i=1}^m f(v_{ti}, q) g(v_{ti}), \quad (1)$$

where $g(\cdot)$ is the representation function and $f(\cdot)$ is the importance function, λ_{ti} and v_{ti} are the importance and the embedding of the i -th item in S_i^u , and q is the vector for item importance calculation.

For the importance function $f(\cdot)$, we apply an attention mechanism [35] to produce the importance weight of all items in the basket as shown in Fig. 3. Attention mechanism is a

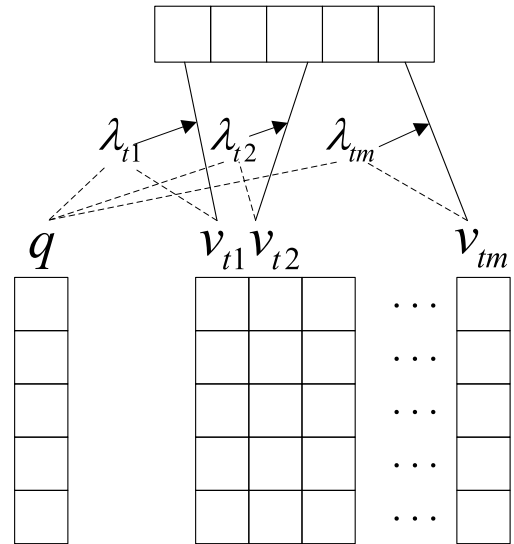


FIGURE 3. Aggregation for basket representation.

method that can dynamically determine the importance of different components, which is widely applied in document classification [36], recommender systems [19], [34], [37], etc. Moreover, attention mechanism is usually used together with the softmax function, where the softmax function is utilized to

normalize the attention scores. We first feed the embedding vectors into an one-layer Multi Layer Perception (MLP) to get the hidden representation $\{k_{t1}, k_{t2}, \dots, k_{tm}\}$:

$$k_{ti} = \tanh(W_k v_{ti} + b), \quad (2)$$

where $W_k \in \mathbb{R}^{d \times d}$, $b \in \mathbb{R}^d$ are a weight matrix and a bias term to be learnt, respectively. Then, we initialize an vector for the attention mechanism, i.e., $q \in \mathbb{R}^d$, to caculate the weight of each item. Following [34], [36], the weights of items in the basket is calculated with a softmax function as follows:

$$f(v_{ti}, q) = \frac{\exp(q^T k_{ti})}{\sum_{i=1}^m \exp(q^T k_{ti})}, \quad (3)$$

For the representation function $g(\cdot)$ in Eq. (1), we calculate it as follows:

$$g(v_{ti}) = W_g v_{ti}, \quad (4)$$

where $W_g \in \mathbb{R}^{d \times d}$ is a weight matrix that projects v_{ti} into a new embedding space.

After aggregating the items in the basket, the basket is eventually represented by $B_t^u \in \mathbb{R}^d$ as produced by Eq. (1), where d is the dimension of embedding dimension.

C. LONG- AND SHORT-TERM PREFERENCES

After generating the representation of all historical baskets, we identify user's final preference by considering the interaction history in two aspects, i.e., user's long-term preference and short-term preference, respectively.

As for the long-term preference, we apply GRU to model the sequential information, which applies a linear interpolation to combine the previous activation h_{t-1} and the candidate activation $\hat{h}_t$ together:

$$h_t = (1 - z_t)h_{t-1} + z_t \hat{h}_t \quad (5)$$

where z_t is the update gate that controls how much information in the former state h_{t-1} should be preserved. z_t can be formalized as:

$$z_t = \sigma(W_z B_t^u + U_z h_{t-1}) \quad (6)$$

where B_t^u is the input basket representation at timestamp t . The candidate activation function is computed as:

$$\hat{h}_t = \tanh(WB_t^u + U(r_t \odot h_{t-1})) \quad (7)$$

where r_t is the reset gate, controlling how much information of previous state should be written in the candidate activation, which is formalized as:

$$r_t = \sigma(W_r B_t^u + U_r h_{t-1}) \quad (8)$$

where σ represents the activation function sigmoid for all the above formulas, and $W_z, U_z, W, U, W_r, U_r \in \mathbb{R}^{d \times d}$ are the trainable neural network parameters in the network. Here we adopt the last hidden output h_t as user's long-term preference P_t^u , i.e., $P_t^u = h_t$.

As for the short-term preference, which is denoted as P_s^u , we regard the representation of the last basket as the current preference, i.e.,

$$P_s^u = B_n^u, \quad (9)$$

where B_n^u can be generated by the similar scheme introduced in Eq. (1).

D. PREFERENCE FUSION AND PREDICTION

After getting user's long-term and short-term preferences, we combine them for item recommendation prediction. We identify user's final preference as P_u by

$$P_u = \tanh(W_l P_t^u) \odot \tanh(W_s P_s^u) \quad (10)$$

where $W_l, W_s \in \mathbb{R}^{d \times d}$ are the trainable parameters; $\odot$ is the Hadamard product used for combination.

Thus, given user u 's historical baskets B^u and a set of items I to be recommended, we calculate the probability of each item $i \in I$ being added to user's current basket by multiplying the user preference P_u with the embeddings of all candidate items, and a softmax function is utilized to normalize the predictions scores:

$$\hat{y} = \frac{\exp(P_u^T v_i)}{\sum_{i=1}^{|I|} \exp(P_u^T v_i)}, \quad \forall i = 1, 2, \dots, |I| \quad (11)$$

where $v_i \in \mathbb{R}^d$ is item i 's embedding, and $\hat{y} = \{\hat{y}_1, \hat{y}_2, \dots, \hat{y}_{|I|}\}$ are the prediction scores on all candidate items. Finally, we recommend the items with the highest probabilities.

To effectively train the model, we adopt the cross-entropy as the optimization objective to learn the parameters, which is defined as:

$$L = -\sum_{u \in U} \sum_{i \in I} y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i), \quad (12)$$

where y denotes an one-hot vector of the ground truth. That is, if a user interacts with item k , then, $y_i = 1$ if $i = k$ and $y_i = 0$ if $i \neq k$.

IV. EXPERIMENTS

In this section, we list research questions that guide our experiments in Section IV-A, then briefly summarize the baseline models we choose to compare with as well as our proposed models in Section IV-B. Next, datasets and evaluation metrics are introduced in Section IV-C and Section IV-D, respectively. At last, we give the details of our experimental settings and hyper-parameters in Section IV-E.

A. RESEARCH QUESTIONS

To guide our experiments, we list three research questions:

- RQ1** Can our proposed Hybrid-Preference Neural Model (HPNM) perform better than the state-of-the-art baselines for basket-sensitive item recommendation?
- RQ2** How does the number of historical baskets affect the recommendation performance?

TABLE 1. Statistics of the datasets used in our experiments.

Datasets	#Users	#Items	#Baskets	#Interactions	Ave. Items per Basket	Ave. Baskets per User
TaFeng	11,945	11,022	74,102	537,053	7.2	6.2
Foursquare	1,783	4,581	40,317	120,065	3.0	22.6

RQ3 How well will the HPNM models perform when the average number of items in the historical baskets change?

B. MODEL SUMMARY

To prove the effectiveness of our proposal, we conduct experiments by comparing the performance against several baselines. We list the major models discussed in this paper below.

- **POP** [38]: This is a non-personalized recommendation method, which recommends the most popular items to each user.
- **ASR** [11]: An Association Rule based approach for basket recommendation, which collects all items that the user has interacted with in the historical baskets to make recommendation “personalized”.
- **CBFM** [9]: A personalized model for BSIR, which considers the relationship among the users, the items in the current basket and the target item.
- **SHAN** [34]: A two-layer hierarchical attention network based approach for item recommendation, which jointly applies two attention layers to generate user representation, respectively.
- **HPNM_{ave}**: Our proposed Hybrid-Preference Neural Model, which considers user’s long-term and short-term preferences and the average pooling is utilized for basket representation.
- **HPNM_{max}**: Different from HPNM_{ave}, the max pooling mechanism in HPNM is utilized for basket representation.
- **HPNM_{att}**: Different from HPNM_{ave}, an attention mechanism in HPNM is utilized for basket representation.

C. DATASETS

We evaluate the models on two publicly available datasets, i.e., TaFeng and Foursquare.

- **TaFeng**: TaFeng is a public retail market dataset, which covers products from many aspects. It contains 817,741 transactions between 32,266 users and 23,812 items.
- **Foursquare**: Foursquare is a point-of-interest dataset, which contains user’s check-ins at various places. It contains 194,108 transactions between 2,321 users and 5,596 items [39].

Following CBFM [9], we preprocess the datasets as follows before training. We filter out items with few transactions, i.e., 10 and 5 for TaFeng and Foursquare, respectively. And items that behave like “stop words” with presence in a

large fraction of transactions are removed, which is set as 5 % for both two datasets. There are 2 such items in TaFeng, none in Foursquare. Moreover, baskets with just one item are filtered. We split the data into train/validation/test data, and retain the users who have more than 3 historical baskets. The details of the processed datasets are presented in Table 1.

For both datasets, we use the last transaction of each user for testing, the penultimate transaction as the validation data, and the rest are regarded as the training data. The hyper-parameters are tuned on validation data and applied in the test period.

D. EVALUATION METRICS

In order to evaluate the performance of different models, similar to CBFM [9], we use Recall@K and HLU as metrics.

Recall@K: The Recall@K score (i.e., R@K) is a well-known metric that frequently used in the task of recommendation, which evaluates if the target item is in the recommendation list.

$$\text{Recall@K} = \frac{n_{hit}}{N}, \quad (13)$$

where N is the number of test sequences in the dataset and n_{hit} is the number of cases that the target item is ranked in the top-K positions of the recommendation list. We set K as 10, 20 and 50 for comparisons in our experiments.

HLU: The HLU is an evaluation metric considering the ranking position of target item. HLU can be defined as:

$$HLU = \frac{1}{N} \times C \times \sum_{t_j \in T_{test}} 2^{\frac{1-r_{t_j}}{\beta-1}}, \quad (14)$$

where N is the number of all tests, T_{test} is the test set, r_t is the rank of target item in the recommender list, and C is the scaling parameter. Following CBFM [9], we set β to 5 and C to 100.

E. EXPERIMENTAL SETTINGS

We employ the Adaptive Moment (Adam) as the optimization method for cross-entropy loss. The item embedding dimension is set to 8, and the epochs for two datasets are set to 500 with early stopping. The hyper-parameters are optimized by a grid search on both datasets and set as follows: learning rate α in [0.001, 0.005, 0.01, 0.1], mini-batch size B_s in [32, 64, 128, 256, 512].

In addition, the attention weight is initialized from a normal distribution $N(0, 0.005^2)$ and the bias are set to 0. All

TABLE 2. Model performance in terms of HLU and Recall@K(%) on the TaFeng and Foursquare datasets. The results generated by the best baseline and the best performer in each column are underlined and boldfaced, respectively.

Method	TaFeng				Foursquare			
	HLU	R@10	R@20	R@50	HLU	R@10	R@20	R@50
POP	1.37	2.18	3.76	8.29	4.77	7.16	11.10	20.78
ASR	1.97	2.72	3.56	4.83	7.52	11.06	17.31	25.04
CBFM	2.29	3.60	6.00	11.11	9.38	13.87	18.84	27.43
SHAN	<u>3.46</u>	<u>5.42</u>	<u>8.04</u>	<u>13.22</u>	<u>13.15</u>	<u>18.28</u>	<u>23.77</u>	<u>31.58</u>
HPNM _{ave}	2.89	4.48	6.92	11.81	10.03	14.97	20.05	28.46
HPNM _{max}	2.91	4.58	7.06	11.70	14.96	20.99	27.15	37.60
HPNM _{att}	3.59	5.55	8.28	13.32	15.39	21.29	27.40	36.67

items embeddings are initialized with a normal distribution $N(0, 0.002^2)$.

V. RESULTS AND DISCUSSION

In this section, we first evaluate the overall performance of the discussed models in Section V-A. Then, we zoom in on the effect on item recommendation across various number of historical baskets in Section V-B as well as across various average number of items in historical baskets in Section V-C.

A. OVERALL PERFORMANCE

To answer **RQ1**, we conduct experiments of all discussed models on the TaFeng and Foursquare datasets, and present the results in terms of HLU and Recall@K in Table 2.

First of all, for the baselines, as shown in Table 2, we can find that between the non-personalized approaches, ASR outperforms POP, indicating the utility of association rules in historical interactions. Moreover, we can observe that the personalized approaches, i.e., CBFM and SHAN, generally outperform the non-personalized methods, i.e., POP and ASR, in terms of all metrics. In addition, SHAN beats CBFM in terms of all metrics on two datasets. It could be explained that SHAN considers users' long-term as well as short-term preference for item recommendation while CBFM makes recommendation only based on users' short-term preference. Hence, in the later experiments, we use SHAN as our baseline for comparisons.

Next, we zoom in on the performance of our proposals. We can observe that between the variants of our proposal (i.e., HPNM_{ave} and HPNM_{max}), HPNM_{max} shows obviously better performance than HPNM_{ave}. And clearly, HPNM_{max} beats SHAN in terms of all metrics on Foursquare, which indicates the utility of modeling the basket-level sequential information in the historical baskets. However, HPNM_{max} loses the competition against SHAN on TaFeng, which may be due to the fact that using max pooling for aggregating item embeddings in the basket will introduce bias. Differently, our HPNM_{att} can consistently beat the baseline SHAN in terms

of all metrics on TaFeng and Foursquare. It could be due to the fact that the attention mechanism in HPNM can adaptively select the important items to help generate an accurate basket representation, which leads to the best performance on all datasets.

In detail, the improvements of HPNM_{att} against SHAN in terms of HLU are 3.76% on the TaFeng dataset and 17.03% on the Foursquare dataset. The Recall improvements of HPNM_{att} against SHAN are 2.40%, 2.99% and 0.76% when recommendation numbers $K = 10, 20$ and 50 on TaFeng, respectively, as well as 16.47%, 15.27% and 16.12% on Foursquare. A higher improvement in terms of HLU indicates that our proposal can return the target item at a higher position in the recommendation list.

B. EFFECT OF NUMBER OF HISTORICAL BASKETS

To answer **RQ2**, we analyze the impact on the performance of item recommendation brought by the number of historical baskets. We examine the performance of our three proposals and the baseline SHAN across various number of historical baskets. We group the tests according to the number of historical baskets, i.e., short (no more than 3), medium (between 4 and 6) and long (more than 6) on TaFeng, and short (no more than 12), medium (between 13 and 24) and long (more than 24) on Foursquare as the users in Foursquare have longer histories than TaFeng in general. We plot the results of our proposals and SHAN in terms of Recall@10 and HLU on both two datasets in Fig. 4.

For TaFeng dataset, as shown in Fig. 4a and 4c, we can find that HPNM_{att} performs better than SHAN for most cases with an exception that HPNM_{att} loses the competition against SHAN for the group "long". In addition, on TaFeng, the performance of HPNM_{ave} in terms of both Recall@10 and HLU both show a decreasing trend in general, while the performance of HPNM_{max} first increases and then decreases in terms of both metrics. The difference could be due to the fact that HPNM_{ave} comes up with a serious issue of information loss when the number of historical baskets increases while

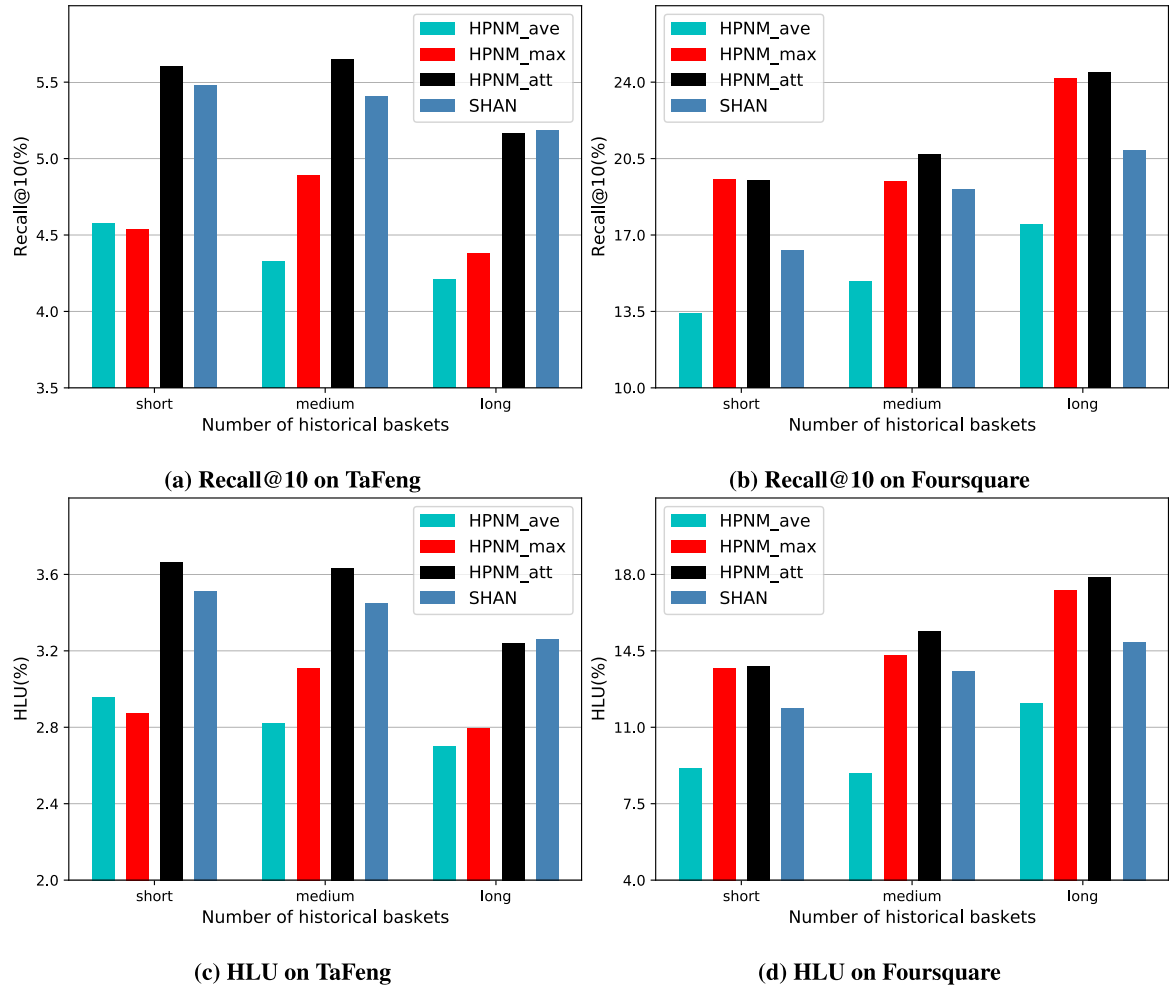


FIGURE 4. Effect of number of historical basket.

HPNM_{max} has the ability to distinguish different importance of the items in these baskets to some extent. However, RNN in the HPNM models has difficulty in modeling overlong sequential information, leading to a drop performance for cases with many baskets. Moreover, HPNM_{att} can consistently outperform other models on all three groups. With the basket number increasing, the performance of HPNM_{att} keeps increasing in terms of HLU while the performance of Recall@10 first increases and then decreases. This indicates that it is more difficult to push the target item at the top position in the recommendation list for users with relatively more historical baskets.

Moreover on Foursquare, the performance of HPNM models as well as SHAN in terms of both metrics show an increasing trend in general with the number of historical baskets increasing. We believe that this could be due to the fact that Foursquare is a point-of-interest dataset, users' interests are narrowed to the geographic location and hence can be identified more accurately when there are more historical baskets.

C. EFFECT OF AVERAGE NUMBER OF ITEMS IN THE BASKET

To answer **RQ3**, we conduct experiments to analyze the impact of average number of items in historical baskets. The results of our proposed models and the baseline SHAN in terms of Recall@10 and HLU are shown in Fig. 5. Note that since the average number of items in historical baskets varies sharply on TaFeng, we split the tests into groups, i.e., [1, 2], [3, 4], ..., and [11, +∞).

Taking Recall@10 as an example, from Fig. 5a and 5b we can observe that HPNM_{att} outperforms SHAN on most cases except the longest group (i.e., [11, +∞)) on TaFeng and relatively small numbers on Foursquare. This phenomenon may be explained by the fact that attention mechanism also has difficulty in distinguishing the items importances accurately on baskets with too many items. In addition, HPNM_{att} outperforms HPNM_{ave} and HPNM_{max} for most cases on both datasets, which shows the ability of attention mechanism for distinguishing items importance. Interestingly, as the average number of items increases, the performance of all models

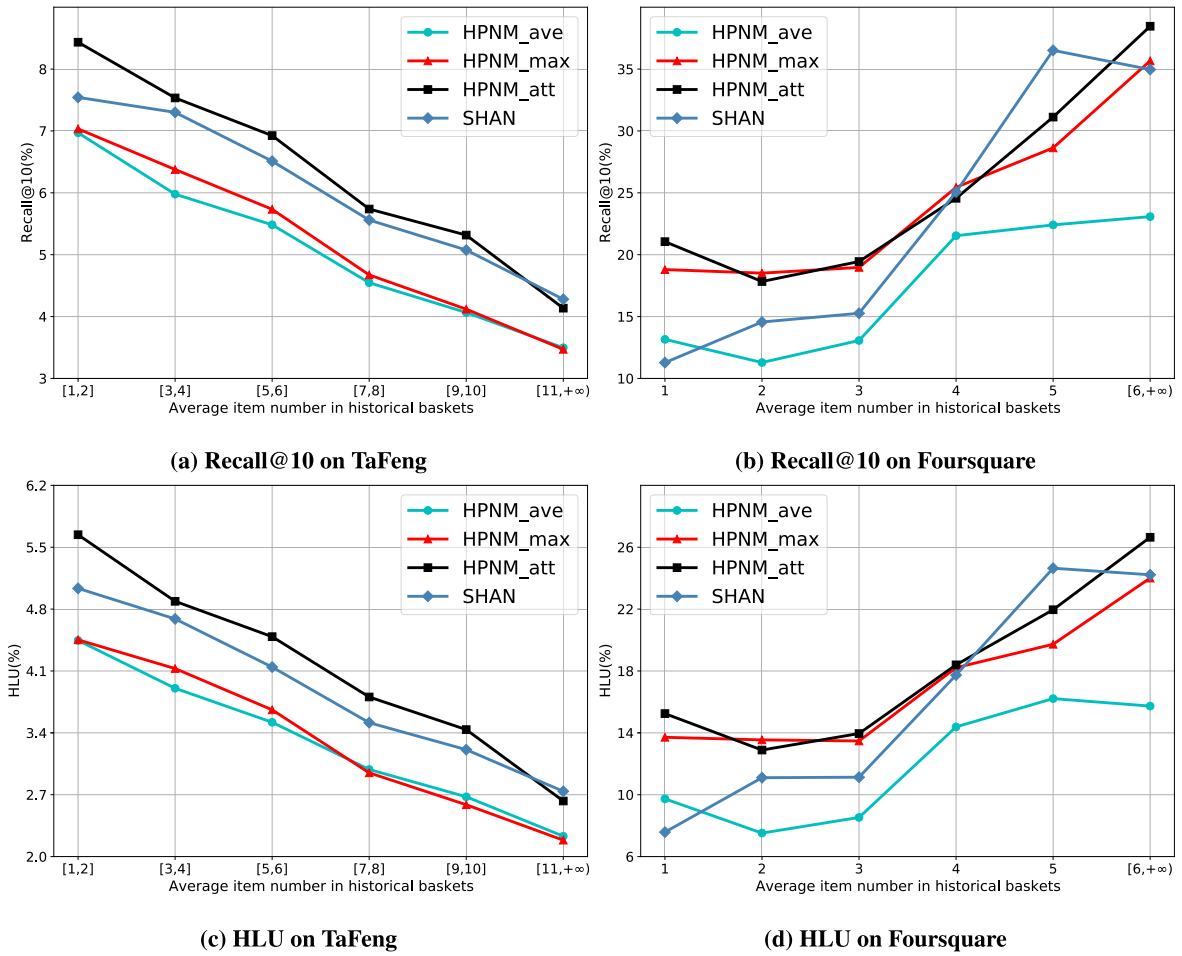


FIGURE 5. Effect of average item number in historical baskets.

decreases in general on TaFeng while increases in general on Foursquare. It could be explained by the fact that the average number of items in historical baskets is quite different, i.e., 7.2 for TaFeng and 3.0 for Foursquare. Specifically, the relatively larger number on TaFeng means that more unrelated items may be included in the basket, resulting in a difficulty on generating accurate basket representation when the average number of items is increasing. In addition, the gap between $HPNM_{att}$ and SHAN is obviously decreasing with the average number increasing on both two datasets. Hence, we would like to conclude that HPNM with the attention mechanism can improve the performance more obviously on baskets with relatively fewer items. Similar phenomenon can also be found in terms of HLU, which indicates that the impacts of the average item number in historical baskets are approximate on Recall and HLU metrics.

D. VISUALIZATION OF ATTENTION MECHANISM

To intuitively show the working mechanism of attention mechanism for aggregating the items in the basket, we conduct a case study of the attention visualization. Specifically, we randomly select two training samples from the TaFeng

dataset, i.e., user A and B, who have 5 and 6 historical baskets, respectively. The attention scores obtained by Eq. (3) are plotted in Fig. 6, the depth of color denotes the attention value of the corresponding item. The numbers on the left and at the bottom denotes the basket index of each user and the item index in each basket, respectively. In the attention visualization for each user, different rows have different lengths since the number of items in user's historical baskets varies.

From Fig. 6, we can get some interesting findings. First, the attention scores are generally more diverse in long sessions than in short sessions. For instance, in basket 1 of user A and basket 1 of user B, the attention scores on most items in the basket are small, while on several specific items (such as item 2 and 9 in basket 1 of user A) the attention scores are relatively larger. This is reasonable, since it is likely to include irrelevant items if the item number of the basket is large. Moreover, we can observe that even for baskets which have the same length, the attention score distributions are also different. For example, as for user B, the basket 0 and basket 3 are both of length 5, however in basket 0 the attention scores are highly similar, while in basket 3 the scores are quite different. This may be due to that the user intent in two baskets

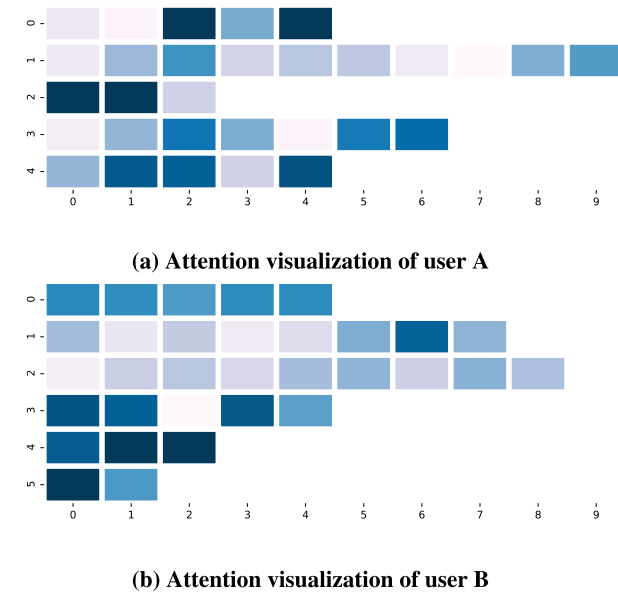


FIGURE 6. Visualization of attention mechanism.

are different, for instance the user aims to buy products of a specific aspect in basket 0, while in basket 3 the user just consumes several unrelated items together for convenience. In general, the attention mechanism can accurately locate the important items in sessions of different lengths and in sessions with different user intent, so as to distinguish the items for generating accurate basket representation.

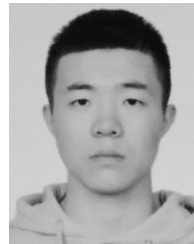
VI. CONCLUSION

In this paper, we propose a Hybrid-Preference Neural Model (HPNM) for basket-sensitive item recommendation, that incorporates a user's long-term and short-term preferences for item recommendation. In particular, an attention mechanism is utilized for distinguishing the importance of items in a basket when representing the basket. Our experimental results show our proposed model can outperform the state-of-the-art baselines. In addition, the experimental results show that our proposed HPNM_{att} can lead to a larger improvement against the baseline in terms of HLU and Recall on baskets with relatively fewer items. As future work, we would like to investigate the item details to generate an accurate basket presentation. For instance, we can take item's attributes into consideration. Moreover, it is potential for us to study the importance of different baskets in sequential purchase history, to distinguish them in modeling long-term preference. In addition, we have interest in investigating different fusion strategies to combine the long-term and short-term preference for generating accurate user preference representation like in [19], [40].

REFERENCES

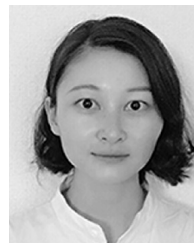
- [1] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowl.-Based Syst.*, vol. 46, pp. 109–132, Jul. 2013.
- [2] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Comput. Surveys*, vol. 52, no. 1, pp. 5:1–5:38, Feb. 2019.
- [3] W. Chen, F. Cai, H. Chen, and M. de Rijke, "A dynamic co-attention network for session-based recommendation," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.*, Beijing, China, Nov. 2019, pp. 1461–1470.
- [4] Z. Yu, J. Lian, A. Mahmood, G. Liu, and X. Xie, "Adaptive user modeling with long and short-term preferences for personalized recommendation," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Macao, China, Aug. 2019, pp. 4213–4219.
- [5] Y. Guo, D. Zhang, Y. Ling, and H. Chen, "A joint neural network for session-aware recommendation," *IEEE Access*, vol. 8, pp. 74205–74215, 2020.
- [6] T. Zhu, P. Harrington, J. Li, and L. Tang, "Bundle recommendation in e-commerce," in *Proc. 37th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, Gold Coast, QLD, Australia, Jul. 2014, pp. 657–666.
- [7] L. Chen, Y. Liu, X. He, L. Gao, and Z. Zheng, "Matching user with item set: Collaborative bundle recommendation with deep attention network," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Macao, China, Aug. 2019, pp. 2095–2101.
- [8] J. Bai, C. Zhou, J. Song, X. Qu, W. An, Z. Li, and J. Gao, "Personalized bundle list recommendation," in *Proc. World Wide Web Conf.*, San Francisco, CA, USA, May 2019, pp. 60–71.
- [9] D.-T. Le, H. W. Lauw, and Y. Fang, "Basket-sensitive personalized item recommendation," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, Melbourne, VIC, Australia, Aug. 2017, pp. 2060–2066.
- [10] D.-T. Le, H. W. Lauw, and Y. Fang, "Modeling contemporaneous basket sequences with twin networks for next-item recommendation," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Stockholm, Sweden, Jul. 2018, pp. 3414–3420.
- [11] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Analysis of recommendation algorithms for e-commerce," in *Proc. 2nd ACM Conf. Electron. Commerce (EC)*, Minneapolis, MN, USA, 2000, pp. 158–167.
- [12] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T. Chua, "Neural collaborative filtering," in *Proc. 26th Int. Conf. World Wide Web*, Perth, WA, Australia, Apr. 2017, pp. 173–182.
- [13] W. Chen, F. Cai, H. Chen, and M. de Rijke, "Joint neural collaborative filtering for recommender systems," *ACM Trans. Inf. Syst.*, vol. 37, no. 4, pp. 39:1–39:30, 2019.
- [14] X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua, "Neural graph collaborative filtering," in *Proc. 42nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Paris, France, Jul. 2019, pp. 165–174.
- [15] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. 5th Int. Conf. Learn. Represent. (ICLR)*, Toulon, France, Apr. 2017, pp. 1–14.
- [16] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," in *Proc. 4th Int. Conf. Learn. Represent. (ICLR)*, San Juan, Puerto Rico, May 2016, pp. 1–10.
- [17] W.-C. Kang and J. McAuley, "Self-attentive sequential recommendation," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Singapore, Nov. 2018, pp. 197–206.
- [18] Y. Guo, Y. Ling, and H. Chen, "A neighbor-guided memory-based neural network for session-aware recommendation," *IEEE Access*, vol. 8, pp. 120668–120678, 2020.
- [19] J. Li, P. Ren, Z. Chen, Z. Ren, T. Lian, and J. Ma, "Neural attentive session-based recommendation," in *Proc. ACM Conf. Inf. Knowl. Manage.*, Singapore, Nov. 2017, pp. 1419–1428.
- [20] Q. Liu, Y. Zeng, R. Mokhosi, and H. Zhang, "STAMP: Short-term attention/memory priority model for session-based recommendation," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, London, U.K., Jul. 2018, pp. 1831–1839.
- [21] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *Proc. 6th Int. Conf. Learn. Represent. (ICLR)*, Vancouver, BC, Canada, Apr./May 2018, pp. 1–12.
- [22] S. Wu, Y. Tang, Y. Zhu, L. Wang, X. Xie, and T. Tan, "Session-based recommendation with graph neural networks," in *Proc. 33rd AAAI Conf. Artif. Intell. (AAAI)*, Honolulu, HI, USA, Jan./Feb. 2019, pp. 346–353.
- [23] Y. Li, D. Tarlow, M. Brockschmidt, and R. S. Zemel, "Gated graph sequence neural networks," in *Proc. 4th Int. Conf. Learn. Represent. (ICLR)*, San Juan, Puerto Rico, May 2016, pp. 1–20.
- [24] F. Wang, Y. Wen, J. Chen, and B. Cao, "Integrating collaborative filtering and association rule mining for market basket recommendation," in *Proc. 19th Int. Conf., Web Inf. Syst. Eng. (WISE)*, Dubai, United Arab Emirates, Nov. 2018, pp. 19–34.
- [25] T. Osadchiy, I. Poliakov, P. Olivier, M. Rowland, and E. Foster, "Recommender system based on pairwise association rules," *Expert Syst. Appl.*, vol. 115, pp. 535–542, Jan. 2019.

- [26] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," in *Proc. ACM SIGMOD Int. Conf. Manage. Data (SIGMOD)*, Dallas, TX, USA, 2000, pp. 1–12.
- [27] B. Che, P. Zhao, J. Fang, L. Zhao, V. S. Sheng, and Z. Cui, "Inter-basket and intra-basket adaptive attention network for next basket recommendation," *IEEE Access*, vol. 7, pp. 80644–80650, 2019.
- [28] S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "Factorizing personalized Markov chains for next-basket recommendation," in *Proc. 19th Int. Conf. World Wide Web (WWW)*, Raleigh, NC, USA, Apr. 2010, pp. 811–820.
- [29] S. Wang, L. Cao, and Y. Wang, "A survey on session-based recommender systems," *CoRR*, vol. abs/1902.04864, pp. 1–34, Feb. 2019.
- [30] T. Bai, J.-Y. Nie, W. X. Zhao, Y. Zhu, P. Du, and J.-R. Wen, "An attribute-aware neural attentive model for next basket recommendation," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Ann Arbor, MI, USA, Jun. 2018, pp. 1201–1204.
- [31] P. Wang, J. Guo, Y. Lan, J. Xu, S. Wan, and X. Cheng, "Learning hierarchical representation model for NextBasket recommendation," in *Proc. 38th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, Santiago, Chile, Aug. 2015, pp. 403–412.
- [32] F. Yu, Q. Liu, S. Wu, L. Wang, and T. Tan, "A dynamic recurrent model for next basket recommendation," in *Proc. 39th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, Pisa, Italy, Jul. 2016, pp. 729–732.
- [33] D.-T. Le, H. W. Lauw, and Y. Fang, "Correlation-sensitive next-basket recommendation," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Macao, China, Aug. 2019, pp. 2808–2814.
- [34] H. Ying, F. Zhuang, F. Zhang, Y. Liu, G. Xu, X. Xie, H. Xiong, and J. Wu, "Sequential recommender system based on hierarchical attention networks," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Stockholm, Sweden, Jul. 2018, pp. 3926–3932.
- [35] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst., Annu. Conf. Neural Inf. Process. Syst.*, Long Beach, CA, USA, Dec. 2017, pp. 5998–6008.
- [36] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, San Diego, CA, USA: The Association for Computational Linguistics, Jun. 2016, pp. 1480–1489.
- [37] Y. Guo, Z. Cheng, L. Nie, Y. Wang, J. Ma, and M. S. Kankanhalli, "Attentive long short-term preference modeling for personalized product search," *ACM Trans. Inf. Syst.*, vol. 37, no. 2, pp. 19:1–19:27, 2019.
- [38] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734–749, Jun. 2005.
- [39] Q. Yuan, G. Cong, Z. Ma, A. Sun, and N. Magnenat-Thalmann, "Time-aware point-of-interest recommendation," in *Proc. 36th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, Dublin, Ireland, Jul./Aug. 2013, pp. 363–372.
- [40] M. Wang, P. Ren, L. Mei, Z. Chen, J. Ma, and M. de Rijke, "A collaborative session-based recommendation approach with parallel memory modules," in *Proc. 42nd Int. Conf. Res. Develop. Inf. Retr. (ACM SIGIR)*, Paris, France, Jul. 2019, pp. 345–354.



ZHIQIANG PAN received the B.S. degree in information system engineering from the National University of Defense Technology, Hunan, China, in 2019, where he is currently pursuing the M.S. degree in management science and engineering.

His research interests include recommender systems and information retrieval.



WANYU CHEN received the M.S. degree in information system engineering from the National University of Defense Technology, Hunan, China, in 2017, where she is currently pursuing the Ph.D. degree in management science and engineering.

Her research interests include recommender systems and information retrieval.



HONGHUI CHEN received the Ph.D. degree in operational research from the National University of Defense Technology, Hunan, China, in 2007.

He is currently a Professor with the National University of Defense Technology, Hunan, China. His research interests include information system and information retrieval. He has published several papers at SIGIR, IPM, and other top journals.

...