Price DOES Matter! Modeling Price and Interest Preferences in Session-based Recommendation

Xiaokun Zhang
Dalian University of Technology
kun@mail.dlut.edu.cn

Bo Xu
Dalian University of Technology
xubo@dlut.edu.cn

Liang Yang
Dalian University of Technology
liang@dlut.edu.cn

Chenliang Li
Wuhan University
cllee@whu.edu.cn

Fenglong Ma
Pennsylvania State University
fenglong@psu.edu

Haifeng Liu
Dalian University of Technology
liuhaifeng@mail.dlut.edu.cn

Hongfei Lin*
Dalian University of Technology
hflin@dlut.edu.cn

ABSTRACT

Session-based recommendation aims to predict items that an anonymous user would like to purchase based on her short behavior sequence. The current approaches towards session-based recommendation only focus on modeling users' interest preferences, while they all ignore a key attribute of an item, *i.e.,* the price. Many marketing studies have shown that the price factor significantly influences users' behaviors and the purchase decisions of users are determined by both price and interest preferences simultaneously. However, it is nontrivial to incorporate price preferences for session-based recommendation. Firstly, it is hard to handle heterogeneous information from various features of items to capture users' price preferences. Secondly, it is difficult to model the complex relations between price and interest preferences in determining user choices.

To address the above challenges, we propose a novel method Co-guided Heterogeneous Hypergraph Network (CoHHN) for session-based recommendation. Towards the first challenge, we devise a heterogeneous hypergraph to represent heterogeneous information and rich relations among them. A dual-channel aggregating mechanism is then designed to aggregate various information in the heterogeneous hypergraph. After that, we extract users' price preferences and interest preferences via attention layers. As to the second challenge, a co-guided learning scheme is designed to model the relations between price and interest preferences and enhance the learning of each other. Finally, we predict user actions based on item features and users' price and interest preferences. Extensive experiments on three real-world datasets demonstrate the effectiveness of the proposed CoHHN. Further analysis reveals the significance of price for session-based recommendation.

*Corresponding Author.

CCS CONCEPTS

• **Information systems → Recommender systems**.

KEYWORDS

Session-based recommendation, Price preferences, Interest preferences, Heterogeneous hypergraph network, Co-guided learning

1 INTRODUCTION

Recommender system (RS), as an effective tool to mitigate information explosion, plays a critical role in modern e-commerce systems. Conventional RS usually uses long-term behaviors of a user to predict her future actions accordingly [18, 25]. In many cases, however, the user's portrait and her rich historical behavior data are unavailable due to privacy policy or non-logged-in nature. To address this issue, session-based recommendation (SBR) is proposed to predict actions of an anonymous user based on limited interactions (*e.g.,* clicked items and their associated features) within the current session [19, 21, 43].

Existing SBR approaches focus on modeling the user's **interest preferences**, *i.e.,* how much a user likes an item, usually based on recurrent neural network (RNN) [13, 19], attention network [17, 20] and graph neural network (GNN) [5, 43]. Although having achieved impressing performance, these approaches all ignore a significantly important factor, *i.e.,* the user's **price preferences**, which aim to describe how much money a user is willing to pay for an item. Many marketing studies have shown that users' buying behaviors are strongly influenced by the price factor [3, 10, 32]. Thus, the price preferences of users should be taken into consideration when predicting their actions. However, we face two main challenges when modeling users' price preferences for SBR.

In fact, price preferences of a user are not fixed, which dramatically vary according to item categories [32, 50]. For instance, a user may purchase an *expensive laptop* out of high requirements

for the performance of computing, while she might buy *affordable pajamas* to wear at home. Thus, when incorporating users' price preferences, we need to take the corresponding categories of items into account. In such a way, multiple types of information will be involved to describe users' historical behaviors, including *a series of items*, *item prices*, and *item categories*. Those **heterogeneous information** may make existing approaches such as RNN [13, 19], attention network [17, 20], and GNN [5, 43] failed since they are designed to only capture a single type of information. Therefore, to model price preferences for SBR, the first challenge is how to handle such heterogeneous information.

Besides, there is a phenomenon called *price elasticity* in economics, which means that the money that a user is willing to pay for an item fluctuates according to her interest in it [1]. This phenomenon clearly shows that the buying behaviors of a user are determined by both price and interest preferences, which is intuitive since users often make price-interest trade-offs when they purchase items. For example, a user is likely to purchase an expensive item out of strong interest, even if the price of the item exceeds her expectations. In this case, the user adjusts her price preferences due to strong interest. Similarly, users will also buy some items that are not their favorite because of its low price. Therefore, in SBR, **both price and interest preferences are indispensable**. However, how to model the complex relations between price and interest preferences is another challenge.

To tackle the aforementioned challenges, we propose a novel model called C̲o-guided H̲eterogeneous H̲ypergraph N̲etwork (CoHHN) for SBR. In particular, for addressing the first challenge, an intuitive way is to directly apply state-of-the-art heterogeneous graph-based approaches to model *heterogeneous information*. However, existing heterogeneous graph-based methods [15, 28] can only capture up to *k*-hop pairwise relations (*i.e., item → category*), where k is always set to a small value (*e.g.,* 3) due to the over-smoothing issue [5]. In our setting, the relations among various information are high-order not pairwise such as the relations among <*item, price, category*>. Such complex high-order dependencies may make general heterogeneous graph-based approaches failed. Recently, hypergraphs with degree-free hyperedges are proposed to encode high-order data correlation [47]. Inspired by both heterogeneous graphs and hypergraphs, in this paper, we propose a **heterogeneous hypergraph network**, which combines the advantages of heterogeneous graphs in modeling heterogeneous information and hypergraphs in capturing complex high-order dependencies, to handle heterogeneous information in SBR.

Specifically, we encode the following heterogeneous nodes in the heterogeneous hypergraph: item price, item ID and item category, all of which are closely related to price and interest preferences. We define three types of hyperedges, *i.e.,* feature hyperedge, price hyperedge and session hyperedge, which are used to represent multi-type relations among various nodes. Then a dual-channel aggregating mechanism is devised to propagate node embeddings via feature hyperedges. Based on the learned node embeddings, we apply attention layers to extract original price and interest preferences via price and session hyperedges, respectively.

As we mentioned in the second challenge, both price and interest preferences determine user behaviors. To address this challenge, we propose a novel **co-guided learning schema**, which can model

the complex relations between price and interest preferences and enhance the learning of each other. Concretely, based on the original price and interest preferences learned from the heterogeneous hypergraph network, we further make these two preferences guide the learning of each other to enrich their semantics. Finally, Co-HHN makes the recommendation based on item features and users' price and interest preferences.

In summary, the contributions of our work are as follows:

- We highlight the significance of the price factor in determining user behaviors and take both price and interest preferences into consideration to predict user actions in SBR. To our best knowledge, we are the first to consider the price factor in the task of SBR.
- We propose a novel method called Co-guided Heterogeneous Hypergraph Network (CoHHN) to model price preferences, interest preferences and the relations between them to further boost the performance of SBR.
- Extensive experiments over three public benchmarks demonstrate the superiority of our proposed CoHHN compared with state-of-the-art approaches. Further analysis also justifies the importance of the price factor in SBR.

2 RELATED WORK

2.1 Session-based Recommendation

Session-based recommendation (SBR) has attracted increasing attention recently due to its highly practical value [38]. With the powerful representation ability, many neural models are applied to improve SBR. Most early works [12–14, 19, 30, 37] use RNN and its variants by considering RNN's capability of processing sequential data. Attention mechanism is also commonly used in SBR to distinguish the importance of different items in a session [6, 17, 20, 27, 39, 49]. Furthermore, there are some methods [16, 22, 23, 41] utilizing GNN to capture pairwise relations between items. LESSR [5] further handles information loss of GNN for SBR. In addition, some methods explore the characteristics of user behaviors in SBR, such as local invariance property [4], multi interest [29, 40], uncertainty [31, 48] and repeated consumption [7, 24]. Recently, SHARE [36] and S^2-DHCN [45] introduce hypergraph in the task to model the high-order relations among items. Other methods augment session data by co-training [44] or counterfactual schema [42]. Although price is an important factor in determining user behaviors, to our best knowledge, none of the existing works considers it in SBR.

2.2 Price-aware Recommendation

Price is often a key consideration for purchases, but there is a few works introducing price factor into recommender system. Schafer et al. [26] are pioneers to explore the impact of price factor on user behaviors. Chen et al. [2] use the price to hinder performance degradation caused by the unseen categories. Wadhwa et al. [34] re-rank recommendation results by inferring the user affinity to price levels. PUP [50] builds a price-contained heterogeneous graph to model user purchase behaviors. These works only used the price factor as an auxiliary feature in modeling user actions, while ignoring the decisive role of price preferences in the user decision-making process. Besides, none of them is designed for session scenarios.
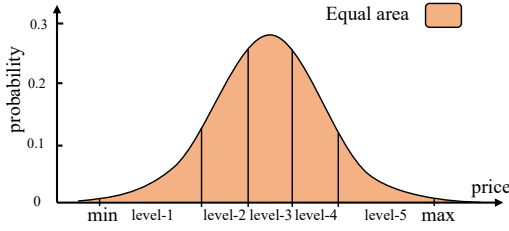
Figure 1: Logistic distribution. We discretize absolute price into different price levels by equally dividing price probability distribution in a category.

2.3 Heterogeneous Graph and Hypergraph

Heterogeneous graph has been demonstrated effective in handling heterogeneous information [15, 28]. For example, Fan et al. [8] propose a metapath-guided method in heterogeneous graph for intent recommendation. Hypergraph is a graph in which a hyperedge can connect more than two nodes [47, 51]. This unique feature enables it to capture complex high-order dependencies among nodes, particularly in recommendation tasks. HyperRec [35] utilizes this capacity of hypergraph to model the short-term user preferences for next-item recommendation. In this paper, we propose a novel heterogeneous hypergraph network which combines the advantages of the heterogeneous graph and hypergraph. The customized structure enables our proposed CoHHN to model the complex high-order dependencies among heterogeneous information and extract users' price and interest preferences more precisely.

3 PRELIMINARIES

3.1 Problem Statement

The focus of our work is to model both price and interest preferences of users for improving session-based recommendation. Let $\mathcal{I}$ denote the set of all unique items and $|\mathcal{I}| = n$ is the total number of items. An item $x_i \in \mathcal{I}$ contains some features, including item ID, price and category. $\mathcal{S} = [x_1, x_2, ..., x_m]$ is a session produced by an anonymous user, where m is the length of $\mathcal{S}$. The proposed CoHHN can be viewed as a function f whose input is the current session $\mathcal{S}$. The output is $\mathbf{y} = f(\mathcal{S}) = [y_1, y_2, ..., y_n]$, where y_j ($j \in [1, ..., n]$) indicates the likelihood that the user will interact with item x_j next.

3.2 Price Discretization

Absolute price of an item is not sufficient to determine if the item is expensive or not [34]. For example, $200 may be a bargain for a cellphone, but the same price would be very high for a lamp. Therefore, we have to discretize item price into different price levels according item categories, such that they can be compared across different categories. As indicated in [9], the price distribution on a certain category is closer to the logistic distribution instead of widely used uniform distribution. As plotted in Figure 1, the probability density function of logistic distribution presents a high preference in the middle and low preference on both sides. Where price is concentrated, users have more items with similar price to choose, leading to higher price sensitivity and the price levels should be divided more finely, and vice versa. Furthermore, to balance the

training data, the number of items contained in each price level should be in the same scale. Therefore, as shown in Figure 1, we discretize price into ρ levels (*e.g.*, $\rho = 5$), where the probability corresponding to each interval is equal. Formally, for an item x_i with price x_p and the price range of its category is $[min, max]$, we determine its price level as follows,

$$p_i = \lfloor \frac{\Phi(x_p) - \Phi(min)}{\Phi(max) - \Phi(min)} \times \rho \rfloor \qquad (1)$$

where $\Phi(x)$ is the cumulative distribution function of logistic distribution, which can be defined as follows,

$$\Phi(x) = P(X \leq x) = \frac{1}{1 + e^{-\pi \frac{x-\mu}{\sqrt{3}\delta}}} \qquad (2)$$

where μ and δ are expected value and standard deviation respectively.

3.3 Heterogeneous Hypergraph Construction

To extract users' price and interest preferences from heterogeneous information, we devise a customized heterogeneous hypergraph $\mathcal{G} = (V, E)$, where V is the set of all nodes and E is the set of all hyperedges. Each node $v^\tau \in V$ owns a type τ and the nodes with same type form a set of homogeneous nodes V^τ. The types of nodes we encode include item price (V^p), item ID (V^{id}) and item category (V^c), all of which are closely related with both price and interest preferences, such that $V = V^p \bigcup V^{id} \bigcup V^c$. Each hyperedge $\epsilon \in E$ connects two or more nodes with arbitrary types. We define following three types of hyperedges to represent multi-type relations among the various nodes: (1) a *feature hyperedge* connects all features of an item; (2) a *price hyperedge* connects the price nodes of all the items in a session. (3) a *session hyperedge* connects the ID nodes of all the items in a session. We utilize feature hyperedges to propagate embeddings of heterogeneous nodes. The price and session hyperedges are used to extract the user's price and interest preferences, respectively. Two nodes are adjacent if they are connected by a hyperedge. The process of constructing the heterogeneous hypergraph is illustrated in the leftmost part of Figure 2.

4 THE PROPOSED METHOD

4.1 Overview of the proposed CoHHN

The proposed CoHHN is illustrated in Figure 2. Based ob the constructed heterogeneous hypergraph, we devise a dual-channel aggregating mechanism to propagate various information among nodes in two channels, *i.e.,* intra-type channel and inter-type channel. After that, we utilize attention layers to extract the user's price and interest preferences based on price and session hyperedges, respectively. A co-guided learning schema is then proposed to model the complex relations between these two preferences in determining the user choices. Finally, we make prediction of user actions based on item features and her price and interest preferences. Next, we elaborate the proposed model CoHHN.

4.2 Dual-channel Aggregating Mechanism

The customized heterogeneous hypergraph endows the proposed CoHHN with the potential to model the complex high-order dependencies among heterogeneous nodes. However, it is hard to extract
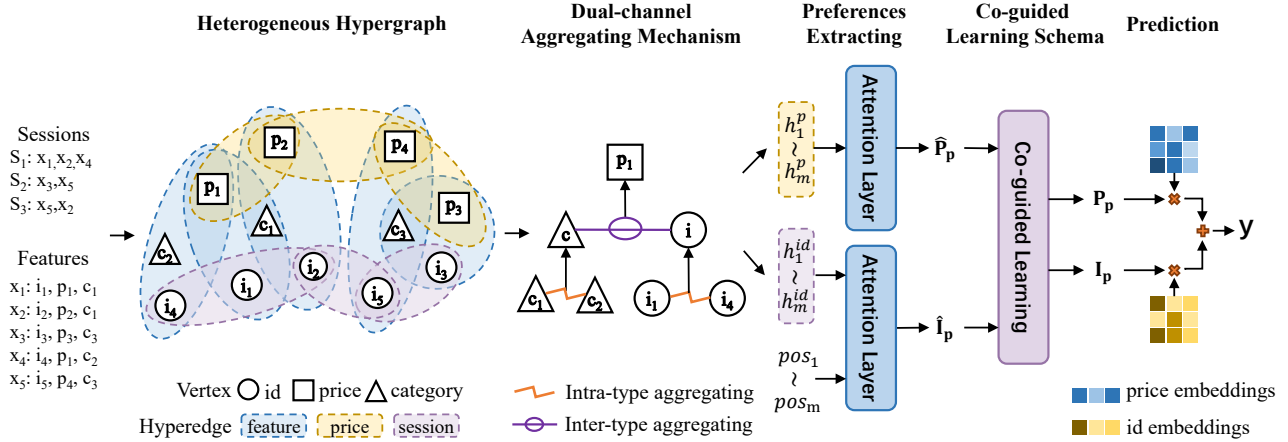
Figure 2: The workflow of the proposed CoHHN. We first construct a heterogeneous hypergraph based on all sessions. And a dual-channel aggregating mechanism is devised to aggregate various information among heterogeneous nodes. After that, we utilize attention layers to extract user price and interest preferences based on price and session hyperedges, respectively. A co-guided learning schema is then proposed to model the complex relations between these two preferences. Lastly, we predict user future actions based on item features and users' price and interest preferences. (*Best viewed in color*)

useful information from heterogeneous nodes. In the constructed heterogeneous hypergraph, a target node can be adjacent with nodes from many types. It is obvious that nodes with same type contain homogeneous information, and different types have different semantics. Therefore, we summarize the message aggregation into two channels, *i.e., intra-type channel* and *inter-type channel*. In the intra-type channel, for a target node, adjacent nodes with the same type have different importance. Also, in the inter-type channel, different types may have different impacts on the target node. To aggregate rich information from two channels, we devise a novel dual-channel aggregating mechanism.

Intra-type aggregating. $v^t \in \mathbb{R}^d$ is the embedding of a target node with type t. Its adjacent nodes with a certain type (τ) form a set N_t^τ. The intra-type aggregating aims to distinguish the importance of nodes in N_t^τ and aggregate the information from these nodes to represent the type (τ). Formally, we learn an representation of type τ for a target node v^t as follows,

$$e_t^\tau = \sum_i \alpha_i v_i^\tau \tag{3}$$

$$\alpha_i = \frac{exp(u_\tau v_i^\tau)}{\sum_{v_i^\tau \in N_t^\tau} \exp(u_\tau v_i^\tau)} \tag{4}$$

where $u_\tau^\top \in \mathbb{R}^d$ is the attention vector which determines the relevance of a node $v_i^\tau \in N_t^\tau$ on v^t, and $e_t^\tau \in \mathbb{R}^d$ is the type embedding for v^t about type τ. Considering that the nodes with same type contain homogeneous information, the intra-type aggregating merges the information at the embedding level. It is worth noting that we apply different attention vectors for different types to extract the type information, which enables our model to learn customized type embeddings for every node. We can abstract the process of intra-type aggregating defined by Eq.(3-4) as a function,

$$e_t^\tau = f_a(N_t^\tau) \tag{5}$$

Inter-type aggregating. Based on the learned type embedding e_t^τ, the inter-type aggregating propagates the embeddings of different types to the target node. Given that different types provide heterogeneous information for the target node, the inter-type aggregating merges the information at the feature level as follows,

$$g_t = \sigma(W_t[v^t; e_t^{\tau 1}; e_t^{\tau 2}] + W_t^{\tau 1}e_t^{\tau 1} + W_t^{\tau 2}e_t^{\tau 2}) \tag{6}$$

$$h^t = v^t + g_t * e_t^{\tau 1} + (1 - g_t) * e_t^{\tau 2} \tag{7}$$

where $W_t \in \mathbb{R}^{d \times 3d}$, $W_t^{\tau 1}$ and $W_t^{\tau 2} \in \mathbb{R}^{d \times d}$ are learnable parameters, [;] is concatenation, σ is sigmoid function and $*$ is element-wise product. $h^t \in \mathbb{R}^d$ is the target node's embedding whose semantics are enriched by its adjacent nodes. The process of inter-type aggregating as illustrated in Eq.(6-7) can be defined as follows,

$$h^t = f_b(v^t, e_t^{\tau 1}, e_t^{\tau 2}) \tag{8}$$

With the dual-channel aggregating mechanism, we update the embeddings of all nodes as follows,

$$h^{id} = f_b(v^{id}, f_a(N_{id}^p), f_a(N_{id}^c)) + avg(N_{id}^{id}) \tag{9}$$

$$h^p = f_b(v^p, f_a(N_p^{id}), f_a(N_p^c)) \tag{10}$$

$$h^c = f_b(v^c, f_a(N_c^p), f_a(N_c^{id})) \tag{11}$$

where v^{id}, v^p and $v^c \in \mathbb{R}^d$ and N_{id}^{id} consists of ID nodes adjacent with v^{id}. The $avg(N_{id}^{id})$ calculates the averaged ID embeddings in N_{id}^{id}. We iteratively repeat the dual-channel aggregating operation r times to fully extract the useful information from various nodes. As a result, our proposed CoHHN is able to model complex high-order dependencies among various features of items and capture price and interest preferences of users. Note that, we mainly use feature hyperedges to propagate information in the dual-channel aggregating mechanism. At last, we obtain embeddings for all nodes as h^{id}, h^p and $h^c \in \mathbb{R}^d$.

4.3 Preferences Extracting

Based on the learned node embeddings, we further extract users' price and interest preferences for recommendation.

Extracting price preferences. We rely on the price hyperedge, *i.e.*, $[\mathrm{h}_1^p, \mathrm{h}_2^p, \mathrm{h}_3^p, ..., \mathrm{h}_m^p]$, to extract the user's price preferences, given that the user's price preferences are reflected in the price of items she has purchased. Self-attention has been demonstrated to be good at capturing item-item transition patterns across the entail input sequence[33, 46]. Therefore, we use self-attention with multi-head mechanism to model the price preferences as follows,

$$E_p = [\mathrm{h}_1^p; \mathrm{h}_2^p; ...; \mathrm{h}_m^p] \tag{12}$$

$$head_i = Attention(W_i^Q E_p, W_i^K E_p, W_i^V E_p) \tag{13}$$

$$S_p = [head_1; head_2; ...; head_h] \tag{14}$$

where W_i^Q, W_i^K and $W_i^V \in \mathbb{R}^{\frac{d}{h} \times d}$ are parameter matrices used to map the input to query, key and value respectively, and h is the number of heads in the self-attention block. The hidden state S_p represents original price preferences of a user $\hat{P}_p \in \mathbb{R}^d$,

$$\hat{P}_p = S_p^{(t)} \tag{15}$$

Extracting interest preferences. We then extract the user's interest preferences based on the session hyperedge $[\mathrm{h}_1^{id}, \mathrm{h}_2^{id}, \mathrm{h}_3^{id}, ...\mathrm{h}_m^{id}]$. The user's interest preferences drift along with time [52]. Therefore, we integrate the position embeddings with the learned ID embeddings to model the dynamic interest of users. Specifically, we utilize the reversed position embeddings [41, 45] to encode the position information, *i.e.*, $\mathrm{pos}_i \in \mathbb{R}^d$. And then, we merge the h_i^{id} and pos_i as follows,

$$\mathrm{v}_i^* = tanh(W_f[\mathrm{h}_i^{id}; \mathrm{pos}_i] + \mathrm{b}_f) \tag{16}$$

where $W_f \in \mathbb{R}^{d \times 2d}$ and b_f are learnable parameters. $\mathrm{v}_i^* \in \mathbb{R}^d$ is the embedding of i-th item in the session. Following [20, 43], we obtain the original interest preferences as follows,

$$\hat{I}_p = \sum_{i=1}^{t} \beta_i \mathrm{h}_i^{id} \tag{17}$$

$$\beta_i = \mathrm{u}\sigma(A_1 \mathrm{v}_i^* + A_2 \bar{\mathrm{v}}^* + \mathrm{b}) \tag{18}$$

where A_1, $A_2 \in \mathbb{R}^{d \times d}$ and b are learnable parameters, $\mathrm{u}^T \in \mathbb{R}^d$ is the attention vector and $\bar{\mathrm{v}}^* = \frac{1}{m} \sum_{i=1}^{m} \mathrm{v}_i^*$. $\hat{I}_p \in \mathbb{R}^d$ is the representation for original interest preferences of a user.

4.4 Co-guided Learning Schema

In a session, we have learned the user's original price preferences $\hat{P}_p$ and interest preferences $\hat{I}_p$. Given an item x_i with its ID embedding v_i^{id} and price embedding v_i^p, we can simply define the probability the user will interact with it as follows,

$$y_i = \hat{P}_p^\top \mathrm{v}_i^p + \hat{I}_p^\top \mathrm{v}_i^{id} \tag{19}$$

However, as stated in former sections, the price preferences and interest preferences influence each other and jointly determine users' choices. Obviously, simple addition is unable to tackle such

a complex situation. Thus, we propose a novel co-guided learning schema to model the mutual relations between price and interest preferences in determining user choices.

We first integrate the price preferences $\hat{P}_p$ with interest preferences $\hat{I}_p$ in two ways,

$$\mathrm{m}_c = tanh(W_1^{pi}(\hat{P}_p * \hat{I}_p) + \mathrm{b}_c) \tag{20}$$

$$\mathrm{m}_j = tanh(W_2^{pi}(\hat{P}_p + \hat{I}_p) + \mathrm{b}_j) \tag{21}$$

where W_1^{pi}, $W_2^{pi} \in \mathbb{R}^{d \times d}$, b_c and b_j are learnable parameters. The m_c and $\mathrm{m}_j \in \mathbb{R}^d$ represent interactive relations between price and interest preferences in different semantic spaces. And then, we utilize gating mechanism to further model the mutual relations between $\hat{P}_p$ and $\hat{I}_p$ as follows,

$$\mathrm{r}_P = \sigma(W_1^r \mathrm{m}_c + U_1^r \mathrm{m}_j) \tag{22}$$

$$\mathrm{r}_I = \sigma(W_2^r \mathrm{m}_c + U_2^r \mathrm{m}_j) \tag{23}$$

$$\mathrm{m}_P = tanh(W^P(\mathrm{r}_P * \hat{P}_p) + U^P((1 - \mathrm{r}_I) * \hat{I}_p)) \tag{24}$$

$$\mathrm{m}_I = tanh(W^I(\mathrm{r}_I * \hat{I}_p) + U^I((1 - \mathrm{r}_P) * \hat{P}_p)) \tag{25}$$

where $W_1^r, W_2^r, W^P, W^I, U_1^r, U_2^r, U^P$ and $U^I \in \mathbb{R}^{d \times d}$. The r_P (r_I) $\in \mathbb{R}^d$ represents 'the remember gate' which controls how much price (interest) preferences are retained when modeling the relations between them. Note that we also apply the complementary of r_I (r_P), *i.e.*, $1 - \mathrm{r}_I$ ($1 - \mathrm{r}_P$), to incorporate interest (price) preferences to guide the learning of price (interest) preferences, which enriches the semantics of these two preferences. Finally, we obtain the user's price and interest preferences as follows,

$$P_p = \mathrm{m}_P * (\hat{P}_p + \mathrm{m}_I) \tag{26}$$

$$I_p = \mathrm{m}_I * (\hat{I}_p + \mathrm{m}_P) \tag{27}$$

where P_p and $I_p \in \mathbb{R}^d$ are the user's price preferences and interest preferences whose semantics are enriched by each other. Note that the price and interest preferences ($\hat{P}_p$ and $\hat{I}_p$) extract information from each other to guide the learning process, which enables our CoHHN to model the complex relations between them in determining user choices.

4.5 Prediction and Training

Given a session, we can calculate the likelihood score y_i corresponding to all items $x_i \in \mathcal{I}$ based on the user's preferences (P_p and I_p) and item's features (v_i^{id} and v_i^p) as follows,

$$y_i = P_p^\top \mathrm{v}_i^p + I_p^\top \mathrm{v}_i^{id} \tag{28}$$

Then we process it with softmax function to obtain the final score as follows,

$$y_i = \frac{exp(y_i)}{\sum_{j=1}^{n} exp(y_j)} \tag{29}$$

Our model can be trained by using cross-entropy loss as follows:

$$\mathcal{L}(\mathrm{p}, \mathrm{y}) = -\sum_{j=1}^{n} p_j \log(y_j) + (1 - p_j) \log(1 - y_j) \tag{30}$$

where p_j is the ground truth that indicates whether the user clicks on item x_j and y_j is the predicted probability of item x_i to be interacted with next.

Table 1: Statistics of the datasets.

Datasets	Cosmetics	Diginetica-buy	Amazon
#item	23,194	24,889	9,114
#price level	10	100	50
#category	301	721	613
#interaction	1,058,263	855,070	487,701
#session	156,922	187,540	204,036
avg.length	6.74	4.56	2.39

5 EXPERIMENTAL SETUP

5.1 Research Questions

We conduct experiments on three real-world datasets for session-based recommendation to evaluate our proposed CoHHN, with the purpose of answering following research questions:

- **RQ1** Does the proposed model CoHHN achieve state-of-the-art performance? (see Section 6.1)
- **RQ2** What is the effect of the price factor in SBR? (see Section 6.2)
- **RQ3** What is the effect of the co-guided learning schema? (see Section 6.3)
- **RQ4** How well does the CoHHN perform at different price levels? (see Section 6.4)
- **RQ5** What is the influence of the key hyper-parameters on CoHHN? (see Section 6.5)

5.2 Datasets and Preprocessing

We evaluate the proposed CoHHN and all baselines on following three real-world public datasets:

- Cosmetics [1] is a kaggle competition dataset, which records user behaviors in a medium cosmetics online store. We use one month (October 2019) records and only retain the interactions with type 'add_to_cart' or 'purchase' in our work.
- Diginetica-buy [2] is an e-commerce dataset which is composed of user purchasing behaviors on the websites.
- Amazon [3], as a widely used benchmark dataset for recommendation, contains users' purchase behavior from Amazon [11]. One representative sub-dataset 'Grocery and Gourmet Food' is adopted in our experiments. We view sequential actions within one day of each user as a session.

Following [5, 19, 43–45, 50], the sessions of length 1 and items that appear less than 10 times are filtered out in all datasets. For a session, the last item is viewed as the label and the remaining sequence is used to model user preferences. We use the earliest 70% (chronologically) of the sessions as the training set and use the next 20% of sessions as the validation set for hyper-parameter tuning on CoHHN and all baselines. The remaining 10% constitutes the test set for reporting model performance. Statistical details of three datasets are shown in Table 1.

---

[1] https://www.kaggle.com/mkechinov/ecommerce-events-history-in-cosmetics-shop
[2] https://competitions.codalab.org/competitions/11161
[3] http://jmcauley.ucsd.edu/data/amazon/

5.3 Evaluation Metrics

As in [5, 13, 19, 27, 43–45], we use following two common metrics to evaluate the performance of all methods.

- Prec@k: Precision measures the proportion of cases in which the ground-truth item is within the recommendation list.
- MRR@k: Mean Reciprocal Rank is the average of reciprocal ranks of the ground-truth items among the recommendation lists.

Note that, the Prec@k does not consider the rank of the item as long as it is amongst the recommendation list. While the MRR@k takes the rank of the item into account, which is important in setting where the order of recommendations matters. And larger metric values indicate better performances for both Prec@k and MRR@k. In this work, we report the results with $k = 10, 20$.

5.4 Baselines

To evaluate the performance of the proposed model CoHHN, we choose following competitive methods as baselines:

- **S-POP** recommends the most frequent items in the current session.
- **SKNN** computes the scores of candidate items according to their occurrences in the neighbor sessions.
- **GRU4Rec** [13] utilizes GRU to mine the sequential patterns within sessions.
- **NARM** [19] employs RNNs with attention mechanism to capture the user's main purpose.
- **BERT4Rec** [27] employs a bidirectional self-attention architecture to encode historical interaction sequences.
- **SR-GNN** [43] builds session graph and utilizes graph neural networks to capture the pairwise transition between items.
- **LESSR** [5] tackles information loss of GNN-based models in SBR by introducing shortcut graph attention and edge-order preserving aggregation layers.
- S^2-**DHCN** [45] captures beyond-pairwise relations among items by hypergraph and uses self-supervised learning to enhance performance.
- **COTREC** [44] combines self-supervised learning with co-training to mine informative self-supervision signals for SBR.

5.5 Implementation Details

For fair comparison, the embedding size is set to 128 for all models. The hyper-parameters of CoHHN and all baselines are optimized via the grid search on the validation set. For CoHHN, the number of self-attention heads is 4 ($h = 4$); the number of price levels (*i.e.*, ρ) is 10, 100 and 50 for Cosmetics, Diginetica-buy and Amazon respectively; the iterative number of dual-channel aggregating (*i.e.*, r) is 3 for Cosmetics/Diginetica-buy and 2 for Amazon. The model optimization is performed by using Adam with the initial learning rate 0.001, and the mini-batch size is fixed at 100. As in the methods [19], we also truncate Back-Propagation Through Time (BPTT) at 19 time steps. We have released the source code and datasets of our model online.[4]

---

[4] https://github.com/Zhang-xiaokun/CoHHN

Table 2: Performance comparison of CoHHN with baselines over three datasets. The results (%) produced by the best baseline and the best performer in each column are underlined and boldfaced respectively. Statistical significance of pairwise differences of CoHHN against the best baseline (*) is determined by the t-test ($p < 0.01$).

Method	Cosmetics				Diginetica-buy				Amazon			
	Prec@10	MRR@10	Prec@20	MRR@20	Prec@10	MRR@10	Prec@20	MRR@20	Prec@10	MRR@10	Prec@20	MRR@20
S-POP	32.83	26.63	38.43	27.32	25.51	18.82	25.91	19.84	34.60	31.96	38.03	32.19
SKNN	40.22	30.40	47.63	30.80	45.68	20.24	55.76	21.10	61.55	46.07	64.23	46.30
GRU4Rec	19.41	14.43	21.80	14.60	22.04	11.32	27.88	11.73	55.43	51.43	56.41	51.70
NARM	42.63	34.17	46.29	34.52	46.56	21.76	57.34	23.27	63.21	57.07	65.38	57.23
Bert4Rec	38.52	23.38	46.30	23.92	47.50	20.44	60.89	21.51	62.83	56.57	64.52	56.74
SR-GNN	44.11	34.59	48.01	34.96	45.74	21.32	56.80	22.87	63.52	<u>57.46</u>	65.83	<u>57.89</u>
LESSR	38.80	24.45	46.32	24.97	47.88	20.82	61.35	22.64	62.48	56.53	64.18	56.69
S^2-DHCN	40.48	32.86	47.95	33.13	45.89	21.08	54.91	22.03	58.67	49.86	60.47	50.03
COTREC	<u>45.11</u>	<u>34.83</u>	<u>50.12</u>	<u>35.24</u>	<u>48.14</u>	<u>22.07</u>	<u>61.82</u>	<u>23.75</u>	<u>64.01</u>	50.06	<u>66.44</u>	50.25
CoHHN	**47.88***	**36.38***	**53.56***	**36.79***	**50.57***	**24.81***	**64.02***	**25.76***	**65.32***	**58.78***	**67.69***	**59.01***

6 RESULTS AND ANALYSIS

6.1 Overall Performance (RQ1)

Table 2 shows the comparison results of CoHHN over other competitive approaches, where the following observations are noted:

(1) Among baselines, different methods perform differently on different datasets. All of the baselines aim at modeling single type of information to capture users' interest preferences, while ignoring that other factors, *e.g.,* price, can also affect user choices. This leads to that the baselines are incapable of accurately predicting user actions across different datasets.

(2) NARM and BERT4Rec outperform GRU4Rec by a large margin, which can own to the attention mechanism they introduce for capturing the main intents in a session. However, BERT4Rec does not perform as good as expected, considering that it has achieved tremendous success in the task of natural language processing. The main advantage of BERT4Rec's structure is that it can capture distant dependencies among items within sequences. As shown in the Table 1, the length of sequence is generally short in session data. We argue that this is the reason for its poor performance.

(3) Relying on the ability for modeling pairwise relations between nodes, GNN-based methods (SR-GNN and LESSR) obtain competitive performance. COTREC further improves the performance of GNN by constructing session graphs in two views. We argue that the view augmentation of COTREC enables the model to capture beyond-pairwise relations among items. Also, S^2-DHCN utilizes the unique structure of hypergraph to do so. However, in essence, the above methods can only capture the single type of information, while hard to model the price and interest preferences hidden in the various features of items. Therefore, its performance is still inferior to the proposed CoHHN.

(4) Our proposed CoHHN shows consistent superiority over all baselines in terms of all evaluation metrics on all datasets. Specifically, CoHHN outperforms best performer in Prec@20 and MRR@20 by 6.86% and 4.40% on Cosmetics, 3.56% and 8.46% on Diginetica and 1.88% and 1.93% on Amazon. We argue that the improvements of the proposed CoHHN benefit from taking price preferences into account. To model the user's price and interest preferences, we devise a novel heterogeneous hypergraph network

Table 3: Performance of variants of CoHHN.

variants	Cosmetics		Diginetica-buy		Amazon	
	Prec@20	MRR@20	Prec@20	MRR@20	Prec@20	MRR@20
COTREC	50.12	35.24	61.82	23.75	66.44	50.25
CoHHN-c	53.05	36.28	63.21	24.97	67.06	58.51
CoHHN-p	52.81	35.92	61.83	22.99	66.93	58.12
CoHHN-pp	52.59	36.08	62.30	23.23	67.14	58.38
CoHHN	**53.56***	**36.79***	**64.02***	**25.76***	**67.69***	**59.01***

which is able to capture complex high-order dependencies among various features of items. Besides, the co-guided learning schema models the relations between price preferences and interest preferences in users' decision-making process, helping the model accurately formulate user future actions.

6.2 The Effect of the Price Factor (RQ2)

A key novelty of CoHHN is that it incorporates the price factor and models both price and interest preferences to predict user actions in SBR. Moreover, we also incorporate item categories into CoHHN, considering that users' price preferences vary widely across different categories. CoHHN-p and CoHHN-c represent the versions of CoHHN without considering price and category respectively. And CoHHN-pp only incorporates the price information to update node embeddings but does not explicitly extract users' price preferences. As can be observed in Table 3, CoHHN-pp outperforms CoHHN-p and CoHHN-c performs better than CoHHN-p, which indicates that incorporating price into SBR does improve the recommendation accuracy. It is also in line with intuition that price plays a critical role in user purchase process. CoHHN achieves large improvements over CoHHN-pp. It demonstrates that price preferences should be considered as a decisive factor rather than auxiliary one when predicting user future actions. CoHHN also defeats CoHHN-c, which reveals that items' category contributes to predicting users' choices. We argue that the introduction of category information enables the CoHHN to learn users' price preferences more precisely.
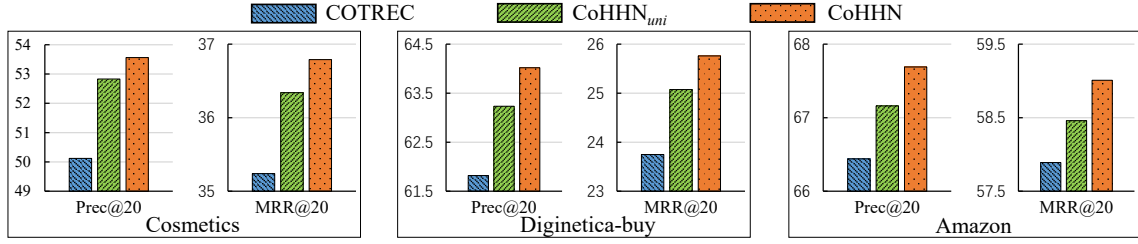
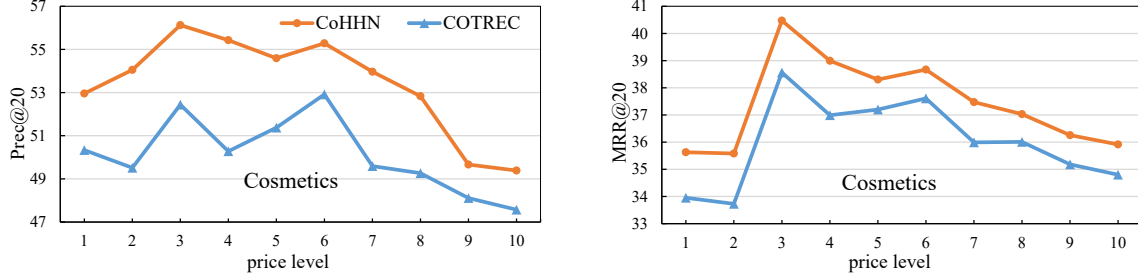Figure 3: The Effect of the price discretization

Figure 4: Performance on different price levels in Cosmetics

Table 4: Effect of the Co-guided Learning

variants	Cosmetics		Diginetica-buy		Amazon	
	Prec@20	MRR@20	Prec@20	MRR@20	Prec@20	MRR@20
COTREC	50.12	35.24	61.82	23.75	66.44	50.25
CoHHN-co	52.79	36.33	62.49	23.54	67.28	58.63
CoHHN	53.56*	36.79*	64.02*	25.76*	67.69*	59.01*

In order to encode price information across different categories, we discretize absolute price into different price levels by equally dividing logistic probability distribution. We demonstrate the effectiveness of this design choice with the variant: $CoHHN_{uni}$ discretizes price into different price levels via uniform quantization as in [50], *i.e.,* $p_i = \lfloor \frac{x_p - min}{max - min} \times \rho \rfloor$. As shown in Figure 3, CoHHN performs better than the $CoHHN_{uni}$. It suggests that our method could encode price information more accurate and reveals the price preferences of users. Furthermore, $CoHHN_{uni}$ achieves a large performance margin over the best baseline COTREC, which indicates the significance of incorporating price preferences in the task again.

6.3 The Effect of the Co-guided Learning (RQ3)

The price and interest preferences influence each other and jointly determine user behaviors. Therefore, we propose a co-guided learning schema to model the complex relations between these two preferences in determining user choices. CoHHN-co uses simple addition to calculate the probability of interactions, that is, as formulated by Eq.(19). According to Table 4, CoHHN defeats CoHHN-co in a large margin. It indicates that the relations between price and interest preferences in determining user choices are complex and the simple addition can not handle the complex situation. The

proposed co-guided learning schema is able to model the mutual relations between price and interest preferences, thus improving the performance for SBR.

6.4 Performance in Different Price Levels(RQ4)

As demonstrated in the previous sections, price factor has a great impact on users' behaviors. Modeling price and interest preferences can improve the accuracy of predicting user actions. Nevertheless, users show different behavior characteristics in these two preferences. For interest preferences, users present huge personalized differences. As for price preferences, most users have a bias towards low prices. Therefore, average accuracy might increase if we blindly recommend low-price items to users. However, such a method may decrease the revenue of manufacturers, because high-price items often bring high profits. It requires us to improve accuracy in not only low-price items but also high-price items, when incorporating price factor into SBR. Therefore, we report the results of CoHHN and the best baseline COTREC under different price levels. Due to space limit, only the results on Cosmetics are presented, while we obtained similar observations on other datasets.

As shown in Figure 4, CoHHN outperforms COTREC at all price levels on both metrics. It demonstrates that our CoHHN is able to percept users' price preferences at different price levels and make personalized recommendation accordingly. The consistent improvements over all price levels ensure that our CoHHN can help manufacturers' earning to grow while improving the prediction accuracy. The method we use to discretize price can optimize price allocation. We speculate that this is one of the reasons why our model works well in all price levels. Furthermore, CoHHN and COTREC have high accuracy for items with mid and low price levels (*i.e.,* 3-6). It is in line with the facts that most people tend to purchase items with mid and low prices.
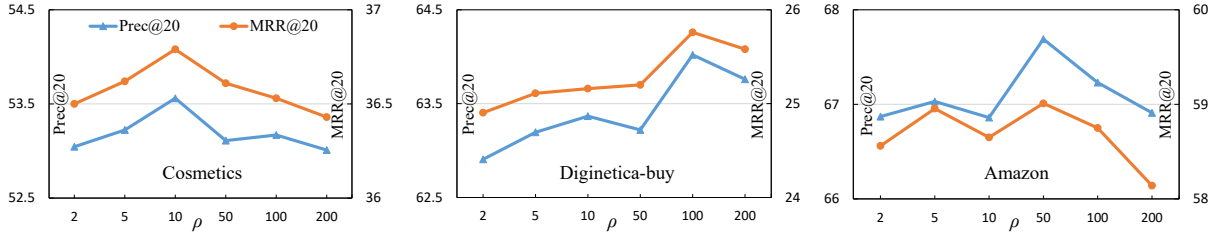
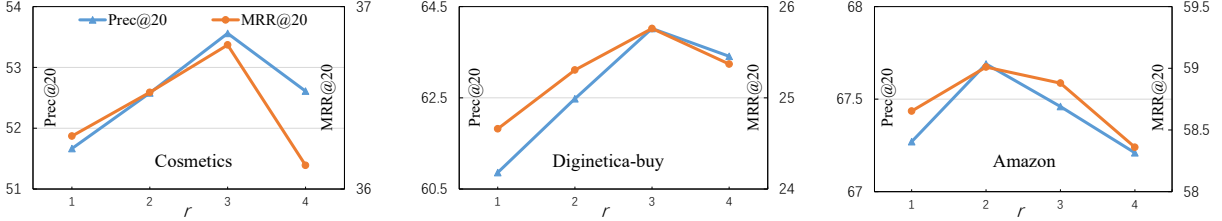**Figure 5: Performance of CoHHN on different number of price levels**

Figure 6: Performance of CoHHN on different iterative number of the dual-channel aggregating

6.5 Hyper-parameter Study (RQ5)

In this section, we evaluate the influence of two key hyper-parameters on the proposed CoHHN, *i.e.*, the number of price levels ρ and the iterative number of the dual-channel aggregating r.

The number of price levels ρ determines the granularity of the price preferences modeling. We conduct experiments on three datasets under different values of ρ to study its influence on model's performance. The performance patterns are plotted in the Figure 5. When the ρ is set extremely low like $\rho = 2$ which means the model only roughly classifies price as expensive or cheap, the model is unable to learn the various price preferences, thus the performance is unsatisfactory. While if we set the ρ too high as 200 in the Cosmetics, items with near price are assigned to different price levels. In this case, there is no significant difference between different price levels, causing performance degradation. In absolute terms, the values close to the optimal ρ present similar performance. We argue that the stable performance benefits from our proposed method for price discretization. The method of equal probability segmentation makes the price reasonably arranged into each level, thus contributing to learning the user's price preferences. Furthermore, the optimal value for ρ is different in different datasets, *i.e.,* 10 in Cosmetics, 100 in Diginetica-buy and 50 in Amazon. We speculate that the diversity of users' price preferences on different datasets results in different optimal values for ρ.

The iterative number of the dual-channel aggregating r controls the degree of information fusion among nodes. For a node in the heterogeneous hypergraph, with each iteration, it incorporates some information of adjacent nodes. As shown in the Figure 6, with the increase of r, the model's performance improves first, and then drops. The reason for this is that the degree of information fusion among nodes increases when r becomes larger, but performing iteration too many times does not help because that the over-smooth problem occurs. Moreover, the optimal value of r is different for different datasets, *i.e.,* 3 for Cosmetics/Diginetica-buy and 2 for

Amazon. As shown in Table 1, the Amazon contains less items than other two datasets. We believe that fewer nodes in the heterogeneous hypergraph make the model learn accurate node embeddings through fewer iterations.

7 CONCLUSION AND FUTURE WORK

Price factor plays an important role in determining user purchase behaviors, which motivates us to propose a novel model Co-guided Heterogeneous Hypergraph Network (CoHHN) to incorporate price into session-based recommendation. Specifically, we innovatively devise a customized heterogeneous hypergraph network to model complex high-order dependencies among various features of items to mine users' price and interest preferences. In the heterogeneous hypergraph network, a dual-channel aggregating mechanism is devised to aggregate various information among heterogeneous nodes and multiple relations. Based on the learned node embeddings, we extract users' price and interest preferences within a session via attention layers. After that, we propose a co-guided learning schema to model complex relations between these two preferences in determining user choices. Comprehensive experiments conducted on three real-world datasets demonstrate the superiority of the proposed CoHHN over state-of-the-art methods. Further analysis also validates the significance of price factor in session-based recommendation.

As to future work, we plan to extend CoHHN to incorporate more information such as item brand and seller's reputation to further explore the preferences of users. Moreover, although specifically devised for modeling price and interest preferences in session-based recommendation, our proposed heterogeneous hypergraph can be easily modified to address other tasks which need to mine knowledge from complex heterogeneous information. In addition, it is also an interesting direction to model the dynamics of users' price preferences.

REFERENCES

[1] Erik Brynjolfsson and Michael D Smith. 2000. Frictionless commerce? A comparison of Internet and conventional retailers. *Management science* 46, 4 (2000), 563–585.

[2] Jia Chen, Qin Jin, Shiwan Zhao, Shenghua Bao, Li Zhang, Zhong Su, and Yong Yu. 2014. Does product recommendation meet its waterloo in unexplored categories?: no, price comes to help. In *SIGIR*. 667–676.

[3] Shih-Fen S Chen, Kent B Monroe, and Yung-Chien Lou. 1998. The effects of framing price promotion messages on consumers' perceptions and purchase intentions. *Journal of retailing* (1998), 353–372.

[4] Tianwen Chen and Raymond Chi-Wing Wong. 2019. Session-Based Recommendation with Local Invariance. In *ICDM*. 994–999.

[5] Tianwen Chen and Raymond Chi-Wing Wong. 2020. Handling Information Loss of Graph Neural Networks for Session-based Recommendation. In *KDD*. 1172–1180.

[6] Wanyu Chen, Fei Cai, Honghui Chen, and Maarten de Rijke. 2019. A Dynamic Co-attention Network for Session-based Recommendation. In *CIKM*. 1461–1470.

[7] Minjin Choi, Jinhong Kim, Joonseok Lee, Hyunjung Shim, and Jongwuk Lee. 2021. Session-aware Linear Item-Item Models for Session-based Recommendation. In *WWW*. 1604–1614.

[8] Shaohua Fan, Junxiong Zhu, Xiaotian Han, Chuan Shi, Linmei Hu, Biyu Ma, and Yongliang Li. 2019. Metapath-guided Heterogeneous Graph Neural Network for Intent Recommendation. In *KDD*. 2478–2486.

[9] Asnat Greenstein-Messica and Lior Rokach. 2018. Personal price aware multi-seller recommender system: Evidence from eBay. *Knowl. Based Syst.* 150 (2018), 14–26.

[10] Sangman Han, Sunil Gupta, and Donald R Lehmann. 2001. Consumer price sensitivity and price thresholds. *Journal of retailing* (2001), 435–456.

[11] Ruining He and Julian J. McAuley. 2016. Ups and Downs: Modeling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering. In *WWW*. 507–517.

[12] Balázs Hidasi and Alexandros Karatzoglou. 2018. Recurrent Neural Networks with Top-k Gains for Session-based Recommendations. In *CIKM*. 843–852.

[13] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2016. Session-based Recommendations with Recurrent Neural Networks. In *ICLR*.

[14] Balázs Hidasi, Massimo Quadrana, Alexandros Karatzoglou, and Domonkos Tikk. 2016. Parallel Recurrent Neural Network Architectures for Feature-rich Session-based Recommendations. In *RecSys*. ACM, 241–248.

[15] Linmei Hu, Tianchi Yang, Chuan Shi, Houye Ji, and Xiaoli Li. 2019. Heterogeneous Graph Attention Networks for Semi-supervised Short Text Classification. In *EMNLP-IJCNLP*. 4820–4829.

[16] Chao Huang, Jiahui Chen, Lianghao Xia, Yong Xu, Peng Dai, Yanqing Chen, Liefeng Bo, Jiashu Zhao, and Jimmy Xiangji Huang. 2021. Graph-Enhanced Multi-Task Learning of Multi-Level Transition Dynamics for Session-based Recommendation. In *AAAI*. 4123–4130.

[17] Wang-Cheng Kang and Julian J. McAuley. 2018. Self-Attentive Sequential Recommendation. In *ICDM*. 197–206.

[18] Yehuda Koren, Robert M. Bell, and Chris Volinsky. 2009. Matrix Factorization Techniques for Recommender Systems. *IEEE Computer* (2009), 30–37.

[19] Jing Li, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tao Lian, and Jun Ma. 2017. Neural Attentive Session-based Recommendation. In *CIKM*. ACM, 1419–1428.

[20] Qiao Liu, Yifu Zeng, Refuoe Mokhosi, and Haibin Zhang. 2018. STAMP: Short-Term Attention/Memory Priority Model for Session-based Recommendation. In *KDD*. ACM, 1831–1839.

[21] Malte Ludewig, Noemi Mauro, Sara Latifi, and Dietmar Jannach. 2020. Empirical analysis of session-based recommendation algorithms. *UMUAI* (2020), 1–33.

[22] Zhiqiang Pan, Fei Cai, Wanyu Chen, Honghui Chen, and Maarten de Rijke. 2020. Star Graph Neural Networks for Session-based Recommendation. In *CIKM*. 1195–1204.

[23] Ruihong Qiu, Jingjing Li, Zi Huang, and Hongzhi Yin. 2019. Rethinking the Item Order in Session-based Recommendation with Graph Neural Networks. In *CIKM*. ACM, 579–588.

[24] Pengjie Ren, Zhumin Chen, Jing Li, Zhaochun Ren, Jun Ma, and Maarten de Rijke. 2019. RepeatNet: A Repeat Aware Neural Recommendation Machine for Session-Based Recommendation. In *AAAI*. 4806–4813.

[25] Badrul Munir Sarwar, George Karypis, Joseph A. Konstan, and John Riedl. 2001. Item-based collaborative filtering recommendation algorithms. In *WWW*. ACM, 285–295.

[26] J Ben Schafer, Joseph Konstan, and John Riedl. 1999. Recommender systems in e-commerce. In *COEC*. 158–166.

[27] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer. In *CIKM*. 1441–1450.

[28] Yizhou Sun and Jiawei Han. 2012. Mining heterogeneous information networks: a structural analysis approach. *SIGKDD Explor.* (2012), 20–28.

[29] Qiaoyu Tan, Jianwei Zhang, Jiangchao Yao, Ninghao Liu, Jingren Zhou, Hongxia Yang, and Xia Hu. 2021. Sparse-Interest Network for Sequential Recommendation. In *WSDM*. 598–606.

[30] Yong Kiam Tan, Xinxing Xu, and Yong Liu. 2016. Improved Recurrent Neural Networks for Session-based Recommendations. In *DLRS@RecSys*. ACM, 17–22.

[31] Xiaohai Tong, Pengfei Wang, Chenliang Li, Long Xia, and ShaoZhang Niu. 2021. Pattern-enhanced Contrastive Policy Learning Network for Sequential Recommendation. In *IJCAI*. 1593–1599.

[32] Panniello Umberto. 2015. Developing a price-sensitive recommender system to improve accuracy and business performance of ecommerce applications. *International Journal of Electronic Commerce Studies* 6, 1 (2015), 1–18.

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *NeurIPS*. 5998–6008.

[34] Soumya Wadhwa, Ashish Ranjan, Selene Xu, Jason H. D. Cho, Sushant Kumar, and Kannan Achan. 2020. Personalizing Item Recommendation via Price Understanding. In *CSIRS@RecSys*.

[35] Jianling Wang, Kaize Ding, Liangjie Hong, Huan Liu, and James Caverlee. 2020. Next-item Recommendation with Sequential Hypergraphs. In *SIGIR*. 1101–1110.

[36] Jianling Wang, Kaize Ding, Ziwei Zhu, and James Caverlee. 2021. Session-based Recommendation with Hypergraph Attention Networks. In *SDM*. 82–90.

[37] Meirui Wang, Pengjie Ren, Lei Mei, Zhumin Chen, Jun Ma, and Maarten de Rijke. 2019. A Collaborative Session-based Recommendation Approach with Parallel Memory Modules. In *SIGIR*. 345–354.

[38] Shoujin Wang, Longbing Cao, Yan Wang, Quan Z. Sheng, Mehmet A. Orgun, and Defu Lian. 2022. A Survey on Session-based Recommender Systems. *ACM Comput. Surv.* (2022), 154:1–154:38.

[39] Shoujin Wang, Liang Hu, Longbing Cao, Xiaoshui Huang, Defu Lian, and Wei Liu. 2018. Attention-Based Transactional Context Embedding for Next-Item Recommendation. In *AAAI*. 2532–2539.

[40] Shoujin Wang, Liang Hu, Yan Wang, Quan Z. Sheng, Mehmet A. Orgun, and Longbing Cao. 2019. Modeling Multi-Purpose Sessions for Next-Item Recommendations via Mixture-Channel Purpose Routing Networks. In *IJCAI*. 3771–3777.

[41] Ziyang Wang, Wei Wei, Gao Cong, Xiao-Li Li, Xianling Mao, and Minghui Qiu. 2020. Global Context Enhanced Graph Neural Networks for Session-based Recommendation. In *SIGIR*. 169–178.

[42] Zhenlei Wang, Jingsen Zhang, Hongteng Xu, Xu Chen, Yongfeng Zhang, Wayne Xin Zhao, and Ji-Rong Wen. 2021. Counterfactual Data-Augmented Sequential Recommendation. In *SIGIR*. 347–356.

[43] Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. 2019. Session-Based Recommendation with Graph Neural Networks. In *AAAI*. 346–353.

[44] Xin Xia, Hongzhi Yin, Junliang Yu, Yingxia Shao, and Lizhen Cui. 2021. Self-Supervised Graph Co-Training for Session-based Recommendation. In *CIKM*. 2180–2190.

[45] Xin Xia, Hongzhi Yin, Junliang Yu, Qinyong Wang, Lizhen Cui, and Xiangliang Zhang. 2021. Self-Supervised Hypergraph Convolutional Networks for Session-based Recommendation. In *AAAI*. 4503–4511.

[46] Chengfeng Xu, Pengpeng Zhao, Yanchi Liu, Victor S. Sheng, Jiajie Xu, Fuzhen Zhuang, Junhua Fang, and Xiaofang Zhou. 2019. Graph Contextualized Self-Attention Network for Session-based Recommendation. In *IJCAI*. 3940–3946.

[47] Naganand Yadati, Madhav Nimishakavi, Prateek Yadav, Vikram Nitin, Anand Louis, and Partha P. Talukdar. 2019. HyperGCN: A New Method For Training Graph Convolutional Networks on Hypergraphs. In *NeurIPS*. 1509–1520.

[48] Xiaokun Zhang, Hongfei Lin, Bo Xu, Chenliang Li, Yuan Lin, Haifeng Liu, and Fenlong Ma. 2022. Dynamic Intent-aware Iterative Denoising Network for Session-based Recommendation. *Information Processing & Management* (2022), 102936.

[49] Xiaokun Zhang, Hongfei Lin, Liang Yang, Bo Xu, Yufeng Diao, and Lu Ren. 2021. Dual Part-pooling Attentive Networks for Session-based Recommendation. *Neurocomputing* (2021), 89–100.

[50] Yu Zheng, Chen Gao, Xiangnan He, Yong Li, and Depeng Jin. 2020. Price-aware Recommendation with Graph Convolutional Networks. In *ICDE*. 133–144.

[51] Dengyong Zhou, Jiayuan Huang, and Bernhard Schölkopf. 2006. Learning with Hypergraphs: Clustering, Classification, and Embedding. In *NeurIPS*. 1601–1608.

[52] Huachi Zhou, Qiaoyu Tan, Xiao Huang, Kaixiong Zhou, and Xiaoling Wang. 2021. Temporal Augmented Graph Neural Networks for Session-Based Recommendations. In *SIGIR*. 1798–1802.