





# Pseudo Session-Based Recommendation with Hierarchical Embedding and Session Attributes

Yuta Sumiya<sup>1</sup> , Ryusei Numata<sup>2</sup>, and Satoshi Takahashi<sup>1</sup> 

<sup>1</sup> The University of Electro-Communications, Tokyo, Japan  
{sumiya, stakahashi}@uec.ac.jp

<sup>2</sup> The Japan Research Institute Limited, Tokyo, Japan  
numata.ryusei@jri.co.jp

**Abstract.** Recently, electronic commerce (EC) websites have been unable to provide an identification number (user ID) for each transaction data entry because of privacy issues. Because most recommendation methods assume that all data are assigned a user ID, they cannot be applied to the data without user IDs. Recently, session-based recommendation (SBR) based on session information, which is short-term behavioral information of users, has been studied. A general SBR uses only information about the item of interest to make a recommendation (e.g., item ID for an EC site). Particularly in the case of EC sites, the data recorded include the name of the item being purchased, the price of the item, the category hierarchy, and the gender and region of the user. In this study, we define a pseudo-session for the purchase history data of an EC site without user IDs and session IDs. Finally, we propose an SBR with a co-guided heterogeneous hypergraph and globalgraph network plus, called CoHHGN+. The results show that our CoHHGN+ can recommend items with higher performance than other methods.

**Keywords:** Session-Based Recommendation · Pseudo Session ID · Session information · Auxiliary information · Heterogeneous hypergraph network · Global Graph · Co-guided Learning

## 1 Introduction

In electronic commerce (EC) markets, the effective recommendation of items and services based on individual user preferences and interests is an important factor in improving customer satisfaction and sales, and several previous studies have focused on recommendation systems. A recommendation system is a technology that suggests items based on a user's past actions and online behavior. However, in recent years, user IDs are not assigned to users to protect their privacy. Under such circumstances, it is difficult to identify users; therefore, conventional effective recommendation systems that need user IDs cannot be used.

Session-based recommendation (SBR), which makes recommendations without focusing on user IDs, is currently attracting attention. SBR is a method

of providing recommendations based on session IDs assigned to short-term user actions. They are assigned when a user logs into an EC site, and are advantageous in that users cannot be uniquely identified as they are assigned different IDs depending on the time of day. However, even if session ID management is inadequate, there is a risk that the session ID of a logged-in user may be illegally obtained to gain access. To prevent this, we propose a new method for recommending items without using either user or session IDs. Specifically, for purchase history data to which user and session IDs are not assigned, records with consecutive user attributes, such as gender and place of residence, are defined as pseudo-sessions, and the next item to be purchased in the pseudo-session is predicted. In this manner, items that anonymous users place in their carts in chronological order can be recommended for their next purchase without using session IDs.

Generally, existing SBRs are often graph neural network (GNN)-based [5] methods that consider only item transactions within a session. However, in the case of purchase history, other features such as item price and category tend to be observed as well. The existing method CoHHN [10] shows that price information and categories are effective in recommending items. In this study, we propose a new GNN model called the co-guided heterogeneous hypergraph and globalgraph network plus (CoHHGN+), which consider not only the purchase transition and price of items, but also the category hierarchy of items and auxiliary information of sessions; our model also learns the co-occurrence relationships with other sessions within the same features, and takes into account the importance of embeddings between different features and same features. In summary, our key contributions are as follows:

1. A pseudo session-based high-accuracy recommendation system is proposed.
2. We exploit session information about users and time series sales.
3. Item hierarchies and co-occurrence relationships of the same features are considered.

## 2 Related Work

Rendle et al. proposed a Markov chain-based SBR model, called factorized personalized Markov chains (FPMC) [6]. FPMC is a hybrid method that combines Markov chains and matrix factorization to capture sequential patterns and long-term user preferences. The method is based on a Markov chain that focuses on two adjacent states between items and is adaptable to anonymous SBRs. However, a major problem with Markov chain-based models is that they combine past components independently, which restricts their predictive accuracy.

Hidasi et al. proposed a recurrent neural network (RNN)-based SBR model called GRU4Rec [4]. GRU4Rec models transition between items using gated recurrent units (GRUs) for inputs represented as graphs.

The purchase transitions of an EC site can be represented by a graph structure, which is a homogeneous or heterogeneous graph depending on whether the attributes of the nodes are singular or plural. Homogeneous graphs are graphs

that represent relationships by only one type of node and edge and are used to represent relationships in social networks. In contrast, heterogeneous graphs are graphs that contain multiple and diverse nodes and edges and are used to represent relationships between stores and customers.

Wu et al. proposed SR-GNN [9], which uses a GNN to predict the next item to be purchased in a session based on a homogeneous graph of items constructed across sessions. Using GNNs, we obtain item embeddings that are useful for predicting by introducing attention mechanism to the continuously observed item information. Currently, SBRs based on this GNN have shown more effective results than other methods, and several extended methods based on SR-GNN have been proposed. Wang et al. proposed GCE-GNN [8], which embeds not only the current session but also item transitions of other sessions in the graph.

Existing methods, such as SR-GNN and GCE-GNN are models that learn item-only transitions; however, sessions may also include item prices and categorical features. To construct a model that takes these into account, it is necessary to use heterogeneous graphs. However, when using graphs to represent the relationship between auxiliary information such as price and items, the graph becomes more complex as the number of items in a particular price range increases. Therefore, we apply an extended heterogeneous hypergraph to allow the edges to be connected to multiple nodes. This makes it possible to understand complex higher-order dependencies between nodes, especially in recommendation tasks [10]. Zhang et al. proposed CoHHN [10], which embeds not only item transitions, but also item prices and categories. While CoHHN can consider price and item dependencies, it does not consider the hierarchical features of categories or sales information and user attributes observed during the sessions. It also does not embed the global information that represents item purchase transitions in other sessions. Therefore, we propose a new GNN model that embeds global information as in GCE-GNN, and considers item category hierarchy, user attributes, and sale information.

### 3 Preliminaries

Let  $\tau$  be a feature type that changes within a given session. Let  $\mathcal{V}^\tau = \{v_1^\tau, v_2^\tau, \dots, v_{n^\tau}^\tau\}$  be a unique set of feature  $\tau$  and  $n^\tau$  be their size. We consider four items: item ID, price, and hierarchical category of item (large and middle); we subsequently denote its item set as  $\mathcal{V}^{\text{id}}$ ,  $\mathcal{V}^{\text{pri}}$ ,  $\mathcal{V}^{\text{lr}}$ , and  $\mathcal{V}^{\text{mid}}$ , respectively. Note that the prices are discretized into several price ranges according to a logistic distribution [2, 10], taking into account the market price of each item.

Let  $S_a^\tau = [v_1^{a,\tau}, v_2^{a,\tau}, \dots, v_s^{a,\tau}]$  be a sequence of the feature  $\tau$  for a pseudo-session and  $s$  be its length. Note that each element  $v_i^{a,\tau}$  of  $S_a^\tau$  is belongs to  $\mathcal{V}^\tau$ . The objective of SBR is to recommend the top  $k$  items from  $\mathcal{V}^{\text{id}}$  that are most likely to be purchased or clicked next by the user in the current session  $a$ .

### 3.1 Heterogeneous Hypergraph and Global Graph

To learn the transitions of items in a pseudo-session, two different graphs are constructed from all available sessions.

We construct heterogeneous hypergraphs  $\mathcal{G}^{\tau_1, \tau_2} = (\mathcal{V}^{\tau_1}, \mathcal{E}_h^{\tau_2})$  to consider the relationships between different features. Let  $\mathcal{E}_h^{\tau_2}$  be a set of hyperedges for feature  $\tau_2$ . Each hyperedge  $e_h^{\tau_2} \in \mathcal{E}_h^{\tau_2}$  can be connected to multiple nodes  $v_i^{\tau_1} \in \mathcal{V}^{\tau_1}$  in the graph. This means that a node  $v_i^{\tau_1}$  is connected to a hyperedge  $e^{\tau_2}$  when the features  $\tau_1$  and  $\tau_2$  are observed in the same record. If several nodes are contained in the same hyperedge, they are considered to be adjacent.

Heterogeneous hypergraphs are a method of constructing graphs with reference to different features; however, transition regarding information about features of the same type is not considered. Additionally, item purchase transitions may include items that are not relevant to prediction. Thus, we construct the global graph shown below.

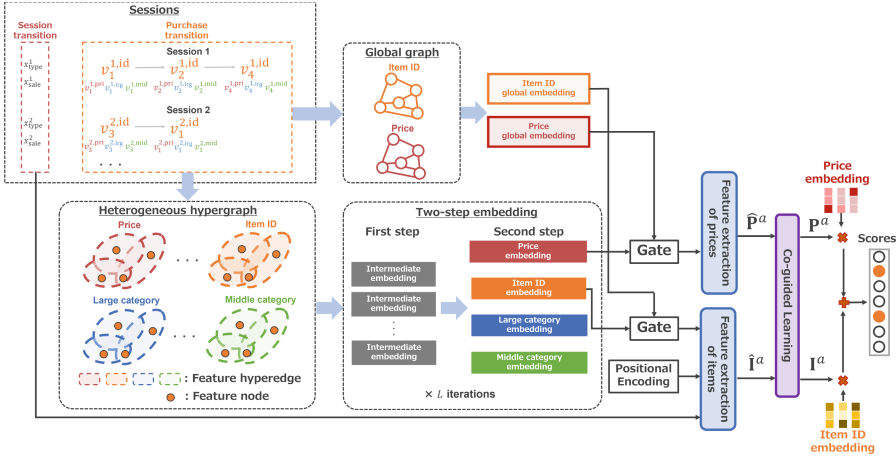
The global graph captures the relationship between items of the same type that co-occur with an item for all sessions. According to [8], the global graph is constructed based on  $\varepsilon$ -neighborhood set of an item for all sessions. Assuming that  $a$  and  $b$  are different arbitrary session, we define the  $\varepsilon$ -neighborhood set as follows.

$$\mathcal{N}_\varepsilon(v_i^{a, \tau}) = \left\{ v_j^{b, \tau} \mid v_i^{a, \tau} = v_j^{b, \tau} \in S_a^\tau \cap S_b^\tau; v_j^{a, \tau} \in S_b^\tau; j \in [i' - \varepsilon, i' + \varepsilon]; a \neq b \right\}, \quad (1)$$

where  $i'$  is an index of  $v_i^{a, \tau}$  in  $S_b^\tau$  and  $\varepsilon$  is a parameter that controls how close items are considered from the position of  $i'$  in session  $B$ . Consider that  $\mathcal{G}_g = (\mathcal{V}^\tau, \mathcal{E}_g^\tau)$  is a global graph where  $\mathcal{E}_g^\tau$  is an edge set and  $e_g^\tau \in \mathcal{E}_g^\tau$  connects two vertices  $v_i^\tau \in \mathcal{V}^\tau$  and  $v_j^\tau \in \mathcal{N}_\varepsilon(v_i^\tau)$ . Notably, the global graph only shows the relationship between identical features, and the adjacency conditions between nodes are not affected by other features.

## 4 Proposed Method

From the perspective of privacy protection, we propose a pseudo session-based recommendation method using a heterogeneous hypergraph constructed from a set of features including a categorical hierarchy, a global graph for item and price features, and additional session attribute information. Figure 1 shows an overview of our proposed method. To consider the interactions and importance between features, our model learns feature embeddings in two steps. In the first step of aggregation, the intermediate embedding of each feature is learned from a heterogeneous hypergraph which consider the interrelationships among different features. In the second step, the final feature embedding vector obtained by aggregating the intermediate embedding in accordance with their respective importance. To address the problem of the heterogeneous hypergraph not being able to learn purchase transitions within the same feature, a global graph is used to incorporate co-occurrence relationships within the same feature into learning. Finally, we propose learning of purchase transitions within a session by considering the features of the session itself, in addition to existing methods.



**Fig. 1.** Overview of the proposed system. First, heterogeneous hypergraphs and global graphs are constructed for all training sessions. In two-step embedding training, embeddings within and between graphs are iteratively trained to obtain multiple feature embeddings, including categorical hierarchies. Then, using the item and price embeddings, we apply co-guided Learning [10] to predict the next item to be purchased by extracting features that account for transitions within the session and the interaction between the two.

**4.1 Two-Step Embedding with Category Hierarchy**

Based on intra-type and inter-type aggregating method in CoHHN [10], we extend it to multiple categorical hierarchies. We obtain the item ID, price, large category, and middle category embedding vectors from the two-step learning method. In the first step of embedding, the embedding of a feature is learned from a heterogeneous hypergraph in which the feature is a node and others are hyperedges. For example, if the item ID is a node, price, large category, and middle category correspond to the hyperedges. In this case, multiple intermediate embeddings are obtained depending on the type of feature, i.e., the hyperedge. In the second step, these embeddings are used to learn the final node embeddings by aggregating them based on their importance. Each learning step is repeated for all  $L$  iterations.

**First Step.** We learn a first-step embedding for a feature  $t$  from a heterogeneous hypergraph, where the target feature  $t$  is a node and another feature  $\tau$  is a hyperedge. First, we define the embedding of a node  $v_i^t \in \mathcal{V}^t$  as  $\mathbf{h}_{l,i}^{\text{hyper},t} \in \mathbb{R}^d$ . Here,  $l$  denotes the location of the training iteration. In the initial state  $l = 0$ , the parameters are initialized using He’s method [3]. Let  $\mathcal{N}_\tau^t(v_i^t)$  be the adjacent node set of  $v_i^t$ . Then, the intermediate embedding of  $v_i^t$  in the  $l$ -iteration is given by

$$\mathbf{m}_{\tau,i}^t = \sum_{v_j^t \in \mathcal{N}_\tau^t(v_i^t)} \alpha_j \mathbf{h}_{l-1,j}^{\text{hyper},t}, \tag{2}$$

$$\alpha_j = \text{Softmax}_j \left( \left[ \mathbf{u}_t^\top \mathbf{h}_{l-1,k}^{\text{hyper},t} \mid v_k^t \in \mathcal{N}_\tau^t(v_i^t) \right] \right), \tag{3}$$

where  $\mathbf{u}_t^\top$  is an attention vector that determines the importance of  $\mathbf{h}_{l-1,j}^{\text{hyper},t}$ . The function  $\text{Softmax}_i$  is defined as

$$\text{Softmax}_i([\mathbf{a}_1, \dots, \mathbf{a}_s]) = \frac{\exp(\mathbf{a}_i)}{\sum_{j=1}^s \exp(\mathbf{a}_j)}. \tag{4}$$

Here,  $\mathbf{m}_{\tau,i}^t \in \mathbb{R}^d$  represents an intermediate embedding of the feature  $t$  when  $\tau$  is a type of hyperedge. In the first step of embedding, we learn the features to focus on when embedding  $t$ .

**Second Step.** Let us assume that  $\mathbf{m}_{\tau_1,i}^t$ ,  $\mathbf{m}_{\tau_2,i}^t$ , and  $\mathbf{m}_{\tau_3,i}^t$  are intermediate embeddings for a feature  $t$  when  $\tau_1, \tau_2, \tau_3$  are types of hyperedge, respectively. By aggregating the embeddings of the first step, we obtain the embedding of  $v_i^t$  shown in the following equation.

$$\mathbf{h}_{l,i}^{\text{hyper},t} = \beta_1 * \mathbf{h}_{l-1,i}^{\text{hyper},t} + \sum_{j=2}^4 \beta_j * \mathbf{m}_{\tau_{j-1},i}^t, \tag{5}$$

$$\beta_j = \text{Softmax}_j \left( \left[ \mathbf{W}^t \mathbf{h}_{l-1,i}^{\text{hyper},t}, \mathbf{W}_{\tau_1}^t \mathbf{m}_{\tau_1,i}^t, \mathbf{W}_{\tau_2}^t \mathbf{m}_{\tau_2,i}^t, \mathbf{W}_{\tau_3}^t \mathbf{m}_{\tau_3,i}^t \right] \right), \tag{6}$$

where  $\mathbf{W}^t, \mathbf{W}_{\tau_1}^t, \mathbf{W}_{\tau_2}^t, \mathbf{W}_{\tau_3}^t \in \mathbb{R}^{d \times d}$  are learnable parameters, and  $*$  denotes the element-wise items of the vectors. Further,  $\beta_j$  is a parameter that computes the importance between the embedding vectors and aggregates the previous and intermediate iteration embeddings.

### 4.2 Embedding of Global Graph

Since heterogeneous hypergraph does not consider the co-occurrence relationships or counts between sessions related to the same feature, we use the learning of embedding global graphs in a GCE-GNN [8] with two configurations: propagation and aggregation of information.

**Information Propagation.** The  $\varepsilon$ -neighborhood of each feature from the global graph for feature  $t$  are embedded. Because the number of features of interest within a neighborhood is considered to be different for each user, based on the attention score shown in the following equation, the neighborhood embedding  $\mathbf{h}_{\mathcal{N}_\varepsilon(v_i^t)}$  is first learned.

$$\mathbf{h}_{\mathcal{N}_\varepsilon(v_i^t)} = \sum_{v_j^t \in \mathcal{N}_\varepsilon(v_i^t)} \pi(v_i^t, v_j^t) \mathbf{h}_{l-1,j}^{\text{global},t}, \tag{7}$$

$$\pi(v_i^t, v_j^t) = \text{Softmax}_j \left( [a(v_i^t, v_k^t) \mid v_k^t \in \mathcal{N}_\varepsilon(v_i^t)] \right), \tag{8}$$

$$a(v_i^t, v_j^t) = \mathbf{q}^\top \text{LeakyRelu} \left( \mathbf{W}_1 \left[ \mathbf{s} * \mathbf{h}_{l-1,j}^{\text{global},t} \right]; w_{ij} \right), \tag{9}$$

where  $\mathbf{h}_{l-1}^{\text{global},t}$  is an embedding of the global graph for the feature  $j$  on the  $l-1$ -th learning iteration, and  $\pi(v_i^t, v_j^t)$  is an attention weight that considers the importance of neighborhood node embedding. The attention score  $a(v_i^t, v_j^t)$  employs LeakyRelu. In LeakyRelu,  $w_{ij} \in \mathbb{R}$  is the weight of an edge  $(v_i^t, v_j^t)$  in the global graph that represents the number of co-occurrences with features  $v_j^t$ , and  $\parallel$  is a concatenation operator. Further,  $\mathbf{W}_1 \in \mathbb{R}^{(d+1) \times (d+1)}$  and  $\mathbf{q} \in \mathbb{R}^{d+1}$  are learnable parameters, and  $\mathbf{s}$  is the average embedding of the session to which  $v_i^t$  belongs, defined as

$$\mathbf{s} = \frac{1}{s} \sum_{v_i^t \in S_a^t} \mathbf{h}_{l-1,i}^{\text{global},t}. \quad (10)$$

**Information Aggregation.** For a feature  $v^t$  to be learned, the  $l$ -iteration embedding  $\mathbf{h}_l^{\text{global},t}$  is obtained by aggregating the  $(l-1)$ -iteration embedding and the neighborhood embeddings using the following formula:

$$\mathbf{h}_{l,i}^{\text{global},t} = \text{ReLU} \left( \mathbf{W}_2 \left[ \mathbf{h}_{l-1,i}^{\text{global},t}; \mathbf{h}_{\mathcal{N}_\varepsilon(v_i^t)} \right] \right), \quad (11)$$

where  $\mathbf{W}_2 \in \mathbb{R}^{d \times 2d}$  denotes a learnable parameter. In global graph embedding, highly relevant item information can be incorporated throughout the session by aggregating the reference features and their  $\varepsilon$ -neighborhoods.

### 4.3 Embedding Feature Nodes

For the feature node  $v_i^t$ , the final embedding is obtained from the embedding of heterogeneous hypergraphs considering the category hierarchy and the embedding of global graphs by the following gate mechanism:

$$\mathbf{g}_i^t = \sigma(\mathbf{W}_3 \mathbf{h}_{L,i}^{\text{hyper},t} + \mathbf{W}_4 \mathbf{h}_{L,i}^{\text{global},t}), \quad (12)$$

$$\mathbf{h}_i^t = \mathbf{g}_i^t * \mathbf{h}_{L,i}^{\text{hyper},t} + (\mathbf{1} - \mathbf{g}_i^t) * \mathbf{h}_{L,i}^{\text{global},t}, \quad (13)$$

where  $\sigma$  is a sigmoid function,  $\mathbf{W}_3 \in \mathbb{R}^{d \times d}$  and  $\mathbf{W}_4 \in \mathbb{R}^{d \times d}$  are learnable parameters, and  $L$  is the final iteration of graph embedding.  $\mathbf{g}_i^t$  is learned to consider the importance of embedding heterogeneous hypergraphs and embedding global graphs. The final feature node embedding is required only for the item ID and price based on the training of the next item.

### 4.4 Feature Extraction Considering Session Attributes

To enhance the recommendation accuracy in pseudo-sessions based on the learned node embeddings, we propose an extraction method of features related to the user's items and prices in each session.

**Feature Extraction of Items.** The embedding of an item node in session  $a$  is given by the sequence  $[\mathbf{h}_1^{a,\text{id}}, \dots, \mathbf{h}_s^{a,\text{id}}]$ . In addition to items, user attribute information, time-series information, and EC site sale information, among others, may be observed in each session. Therefore, we considered this information and learned to capture the session-by-session characteristics associated with the items. Let  $d_{\text{sale}}$  be the number of types of sale information and  $\mathbf{x}_{\text{sale}}^a \in \{0, 1\}^{d_{\text{sale}}}$  items be given per session. Each dimension of this vector represents the type of sale, with a value of 1 if it is during a particular sale period and a value of 0 if it is outside that period. Similarly, if the number of types of attribute information is  $d_{\text{type}}$ , then  $\mathbf{x}_{\text{type}}^a \in \{0, 1\}^{d_{\text{type}}}$  is a vector representing user attributes.

For items and sales, we also consider time-series location information. The item location information defines a location encoding  $\mathit{pos\_item}_i \in \mathbb{R}^d$  as in [7]. Furthermore, for the location information of the sale, the week information to which the current session belongs is encoded by the following formula:

$$\mathit{pos\_time}_{2k-1}^a = \sin\left(\frac{2m\pi}{52k}\right), \tag{14}$$

$$\mathit{pos\_time}_{2k}^a = \cos\left(\frac{2m\pi}{52k}\right), \tag{15}$$

where  $\mathit{pos\_time}^a \in \mathbb{R}^c$  is the location encoding associated with the week information of the session  $a$ ,  $m \in \mathbb{Z}$  represents the week, and  $k$  is the embedding dimension. Because a year comprise 52 weeks, the trigonometric function argument is divided by 52. Based on the above, item embedding in a session is defined as follows:

$$\mathbf{v}_i^{a,\text{id}} = \tanh\left(\mathbf{W}_5 \left[\mathbf{h}_i^{a,\text{id}}; \mathit{pos\_item}_i\right] + \mathbf{W}_6 \left[\mathbf{x}_{\text{sale}}^a; \mathit{pos\_time}^a\right] + \mathbf{W}_7 \mathbf{x}_{\text{type}}^a + \mathbf{b}_1\right), \tag{16}$$

where  $\mathbf{W}_5 \in \mathbb{R}^{d \times 2d}$ ,  $\mathbf{W}_6 \in \mathbb{R}^{d \times (d_{\text{sale}}+c)}$ ,  $\mathbf{W}_7 \in \mathbb{R}^{d \times d_{\text{type}}}$ ,  $\mathbf{b}_1 \in \mathbb{R}^d$  are trainable parameters,  $\mathbf{v}_i^{a,\text{id}}$  is the  $i$ -th item embedding in session  $a$ . The item preferences  $\widehat{\mathbf{I}}^a$  of a user in a session are determined according to [10] as follows:

$$\widehat{\mathbf{I}}^a = \sum_{i=1}^s \beta_i \mathbf{h}_i^{a,\text{id}}, \tag{17}$$

$$\beta_i = \mathbf{u}^\top \sigma(\mathbf{W}_8 \mathbf{v}_i^{a,\text{id}} + \mathbf{W}_9 \bar{\mathbf{v}}^{a,\text{id}} + \mathbf{b}_2), \tag{18}$$

where  $\mathbf{W}_8, \mathbf{W}_9 \in \mathbb{R}^{d \times d}$ ,  $\mathbf{b}_2 \in \mathbb{R}^d$  are learnable parameters,  $\mathbf{u}^\top \in \mathbb{R}^d$  is the attention vector. Additionally,  $\bar{\mathbf{v}}^{a,\text{id}} = \frac{1}{s} \sum_{i=1}^s \mathbf{v}_i^{a,\text{id}}$ .

**Feature Extraction of Prices.** The price hyperedge in session  $a$  is given by  $[\mathbf{h}_1^{a,\text{p}}, \dots, \mathbf{h}_s^{a,\text{p}}]$ . To estimate price preferences with respect to users, we follow [10] and learn the features of the price series using multi-head attention as shown



in the following equation:

$$\mathbf{E}^{a,p} = [\mathbf{h}_1^{a,p}; \dots; \mathbf{h}_s^{a,p}], \quad (19)$$

$$\mathbf{M}_i^{a,p} = [\mathbf{head}_1^a; \dots; \mathbf{head}_h^a], \quad (20)$$

$$\mathbf{head}_i^a = \text{Attention}(\mathbf{W}_i^Q \mathbf{E}^{a,p}, \mathbf{W}_i^K \mathbf{E}^{a,p}, \mathbf{W}_i^V \mathbf{E}^{a,p}), \quad (21)$$

where  $h$  is the number of blocks of self-attention,  $\mathbf{W}_i^Q, \mathbf{W}_i^K, \mathbf{W}_i^V \in \mathbb{R}^{\frac{d}{h} \times d}$  are parameters that map item  $i$  in session  $a$  to query and key, value, and  $\mathbf{head}_i^a \in \mathbb{R}^{\frac{d}{h}}$  is the embedding vector of each block of multi-head-attention for item  $i$ . Further,  $\mathbf{E}^{a,p} \in \mathbb{R}^{dm}$ ,  $\mathbf{M}_i^{a,p} \in \mathbb{R}^d$  and the embedded price series is  $[\mathbf{M}_1^{a,p}, \dots, \mathbf{M}_s^{a,p}]$ .

Because the last price embedding is considered to be the most relevant to the next item price in the price series, we determine the user's price preference  $\widehat{\mathbf{P}}^a = \mathbf{M}_s^{a,p}$  in the session.

#### 4.5 Predicting and Learning About the Next Item

The user's item preferences  $\widehat{\mathbf{I}}^a$  and price preferences  $\widehat{\mathbf{P}}^a$  are transformed into  $\mathbf{I}^a$  and  $\mathbf{P}^a$  respectively by co-guided learning [10], considering mutual dependency relations. When an item  $v_i^{a,\text{id}} \in \mathcal{V}^{\text{id}}$  and a price range  $v_i^{a,p} \in \mathcal{V}^p$  are observed in session  $a$ , the next item to view and purchase is given by the score of the following Softmax function:

$$\widehat{y}_i = \text{Softmax}_i([q_1, \dots, q_{n^{\text{id}}}], \quad (22)$$

$$q_i = \mathbf{P}^a \top \mathbf{h}_i^{a,p} + \mathbf{I}^a \top \mathbf{h}_i^{a,\text{id}}. \quad (23)$$

At the training time, this score is used to compute the cross-entropy loss.

$$\mathcal{L}(\mathbf{y}, \widehat{\mathbf{y}}) = - \sum_{j=1}^{n^{\text{id}}} (y_j \log(\widehat{y}_j) + (1 - y_j) \log(1 - \widehat{y}_j)), \quad (24)$$

where  $\mathbf{y} \in \{0, 1\}^{n^{\text{id}}}$  is the objective variable that indicates whether the user has viewed and purchased item  $v_i^{\text{id}}$ .  $\widehat{\mathbf{y}} \in \mathbb{R}^{n^{\text{id}}}$  is the score for all items.

## 5 Experiments

We evaluate our proposed method using purchasing history data of an EC market. The dataset comprises the purchasing history of 100,000 people randomly selected by age group which are obtained from the users registered in 2019–20 in the Rakuten [1] market, which is a portal site for multiple EC sites. We consider four age groups: 21–35, 36–50, 51–65, and 66–80. Each purchasing history comprises the category name of the purchased item (large, middle, small), week (week 1–105), gender (male or female), residence (nine provinces in Japan), and price segment (separated by thousands of JPY). The user ID and session information are not recorded. Note that this dataset is provided at the 2022 Data Analysis Competition organized by Joint Association Study Group of Management Science and is not open to the public.

**Table 1.** Statistical information of data set.

Age group	21–35	36–50	51–65	66–80
# of price range	10	10	10	10
# of large categories	36	36	36	35
# of middle categories	342	354	340	322
# of small categories	2,800	2,975	2,763	2,327
# of interaction	727,655	1,033,405	712,894	452,496
# of sessions	326,110	462,290	323,184	203,725
Avg. session length	2.24	2.24	2.21	2.24

## 5.1 Preprocessing

Our method recommends a small category name as the item ID. Additionally, the proposed model also considers session attributes, such as purchaser gender, region of residence, and EC site sales. As specific sale information, we include two types of sales that are regularly held at the Rakuten market. Sale 1 is held once every three months for one week, during which many item prices are reduced by up to half or less. Sale 2 is held for a period of one week each month, and more points are awarded for shopping for items on the EC site. Each session attribute is represented by a discrete label. When learning, we treat each gender, region, and sale as a vector with the observed value as 1 and all other values as 0. The price intervals are converted to price range labels by applying a logistic distribution [2].

In each transformed dataset, consecutive purchase intervals with the same gender and residential area are labeled as pseudo-sessions. Based on the assigned pseudo session ID, records with a session length of less than 2 or frequency of occurrence of less than 10 are deleted, according to [10]. Within each session, the last observed item ID is used as the prediction target, and the other series are used for training. In dividing the data, weeks 1 through 101 are used as training data, and the remaining weeks 102 through 105 are used as test data. Additionally, 10% of the training data re used as validation data for hyperparameter tuning of the model. The statistical details of the four datasets are listed in Table 1.

## 5.2 Evaluation Criteria

We employ the following criteria to evaluate the recommendation accuracy:

- **P@ $k$  (Precision)** : The percentage of the top  $k$  recommended items that are actually purchased.
- **M@ $k$  (Mean Reciprocal Rank)** : The mean value for the inverse of the rank of the items actually recommended for purchase. If the rank exceeds  $k$ , it is 0.

The precision does not consider the ranking of recommended items; however, the mean reciprocal rank is a criterion that considers ranking, implying that the

higher the value, the higher the item actually purchased in the ranking. In our experiment, we set  $k = 10, 20$ .

### 5.3 Comparative Model

To verify the effectiveness of the proposed method, we compare it with the following five models.

- **FPMC** [6]: By combining matrix factorization and Markov chains, this method can capture both time-series effects and user preferences. As the dataset is not assigned an ID to identify the user, the observations for each session are estimated as if they were separate users.
- **GRU4Rec** [4]: An SBR based on RNN with GRU when recommending items for each session.
- **SR-GNN** [9]: An SBR that constructs a session graph and captures transitions between items using a GNN.
- **GCE-GNN** [8]: An SBR that builds a session graph and global graph, and captures transitions between items by a GNN while considering their importance.
- **CoHHN** [10]: An SBR that constructs a heterogeneous hypergraph regarding sessions that considers information other than items and captures transitions between items with a GNN.

### 5.4 Parameter Setting

To fairly evaluate the performance of the model, we use many of the same parameters for each model. For all models, the size of the embedding vector is set to 128, the number of epochs to 10, and the batch size to 100. For the optimization method, GRU4Rec uses Adagrad (learning rate 0.01) based on the results of previous studies, while the GNN method uses Adam (learning rate 0.001) with a weight decay of 0.1 applied every three epochs. The coefficients of the L2-norm regularity are set to  $10^{-5}$ . Additionally, in GCE-GNN and our model CoHHGN+, the size of the neighborhood item-set  $\varepsilon$  in the global graph is set to 12. Furthermore, in CoHHN and our model, the number of self-attention heads is set to 4 ( $h = 4$ ), and the number of price ranges to 10. Finally, the number of GNN iterations and percentage of dropouts used in the architecture are determined by grid search for each model using the validation data. We have released the source code of our model online<sup>1</sup>.

## 6 Results and Discussion

### 6.1 Performance Comparison

Tables 2 and 3 show the results of evaluating the five existing methods and the proposed method CoHHGN+ on the four selected datasets. CoHHGN+ obtains

<sup>1</sup> [https://github.com/sumugit/CoHHGN\\_plus](https://github.com/sumugit/CoHHGN_plus).

**Table 2.** Precision of CoHHGN+ and comparative methods. The most accurate value for each dataset is shown in bold, and the second most accurate value is underlined. Each value is the average of three experiments conducted to account for variations due to random numbers. For CoHHGN+ and the other most accurate models, a t-test is performed to confirm statistical significance, and a p-value of less than 0.01 is marked with an asterisk (\*).

Dataset	age 21–35		age 36–50		age 51–65		age 66–80	
Method	P@10	P@20	P@10	P@20	P@10	P@20	P@10	P@20
FPMC	3.84	6.22	4.00	6.56	0.66	2.83	1.13	3.46
GRU4Rec	1.72	2.57	1.73	2.60	1.81	2.82	1.63	2.73
SR-GNN	15.06	20.92	13.71	20.11	13.78	20.34	14.30	22.55
GCE-GNN	15.16	20.88	13.72	20.11	13.87	20.46	<u>14.42</u>	22.46
CoHHN	<u>15.19</u>	<u>21.06</u>	<u>13.96</u>	<u>20.22</u>	<u>13.93</u>	<u>20.73</u>	14.36	<u>22.49</u>
CoHHGN+	<b>15.92*</b>	<b>22.28*</b>	<b>14.75*</b>	<b>22.01*</b>	<b>15.16*</b>	<b>22.57*</b>	<b>15.55*</b>	<b>23.84*</b>

**Table 3.** Mean reciprocal rank of CoHHGN+ and comparative methods. The symbols attached to the values are the same as those in the table 2.

Dataset	age 21–35		age 36–50		age 51–65		age 66–80	
Method	M@10	M@20	M@10	M@20	M@10	M@20	M@10	M@20
FPMC	0.88	1.04	1.14	1.31	0.15	0.28	0.38	0.54
GRU4Rec	0.78	0.84	0.75	0.81	0.71	0.78	0.59	0.66
SR-GNN	6.56	6.95	5.95	6.38	5.51	5.97	5.24	5.80
GCE-GNN	6.65	7.04	<b>6.02</b>	<b>6.45</b>	5.58	6.04	5.21	5.75
CoHHN	<u>6.67</u>	<u>7.06</u>	<u>6.01</u>	<u>6.44</u>	<u>5.62</u>	<u>6.08</u>	<u>5.27</u>	<u>5.81</u>
CoHHGN+	<b>6.89*</b>	<b>7.32*</b>	5.93	6.42	<b>5.83*</b>	<b>6.34*</b>	<b>5.81*</b>	<b>6.37*</b>

the most accurate results for all datasets with precision for  $k = 10, 20$ . The mean reciprocal rank is also the most accurate, except for the data for the 36–50 age group. For the 36–50 year age group dataset, the precision is higher than that for the other models, while the mean reciprocal rank shows the highest accuracy for GCE-GNN. However, there is no statistically significant difference in the prediction accuracy between CoHHGN+ and GCE-GNN in this dataset. Thus, it can be inferred that there is no clear difference in prediction accuracy. This confirms the effectiveness of the proposed method for all the data.

In the comparison method, a large discrepancy in accuracy between the GNN-based method, which introduces an attention mechanism in the purchase series, and the other methods is noted. Overall, the GRU4Rec without attention mechanism results in the lowest accuracy, suggesting that the results were not sufficiently accurate for data with a small number of sessions. This is because the model focuses only on purchase transitions between adjacent items. Similarly, for FPMC, although the accuracy is improved compared to GRU4Rec, modeling

with Markov chains and matrix factorization is not effective for purchase data with pseudo-sessions. Moreover, SR-GNN, GCE-GNN, CoHHN, and CoHHGN+ using graphs of purchase transitions between sessions show a significant improvement in accuracy and are able to learn the purchase trends of non-adjacent items as well.

Among the compared methods, CoHHN, which considered price and large category information in addition to item ID information, tends to have a higher prediction accuracy overall. The number of series per session is generally small for purchase history data, and it can be said that higher accuracy can be obtained by learning data involving multiple features, including items. GCE-GNN, which also considers the features of other sessions, shows the second highest prediction accuracy after CoHHN. When using purchase history data with short session lengths, it is more accurate to learn embedding vectors by considering items that have co-occurrence relationships with other sessions, in addition to series within sessions. The SR-GNN that has learned only from item ID transitions is inferior to the GCE-GNN in terms of overall accuracy among GNN-based systems, although it is more accurate than the GCE-GNN for some datasets. Therefore, it can be considered that adopting features other than the item ID and other session information will lead to improved recommendation accuracy.

We confirm that the proposed method improves accuracy not only by considering auxiliary information in the purchase transition of items, but also by learning methods for its embedding vectors and including additional features that change from session to session. Furthermore, the embedding vector obtained from the global graph of the item of interest works well for a series with short session lengths.

## 6.2 Impact of Each Model Extension

Next, we conduct additional experiments on four datasets to evaluate the effectiveness of embedding item category hierarchies and accounting for session attributes, as well as global-level features. Particularly, we design the following two comparative models:

- CoHHGN (H): A model that incorporates hierarchical embedding of three or more features that vary within a session.
- CoHHGN (HS): A model that considers the hierarchical embedding of three or more features and session attributes in the proposed method.

To compare the performance with existing methods, we use the most accurate values of the existing methods shown in Tables 2 and 3 as the baselines. Tables 4 and 5 show the prediction results of the comparison model. For both precision and Mean Reciprocal Rank, CoHHGN+, which incorporates all the proposed methods, performs better overall than the other two models. For Precision, the accuracy of CoHHGN (HS) is higher for P@10 in the 21–35 year age group dataset. However, because the accuracy of CoHHGN+ is higher than that of other methods in P@20, we believe that considering the embedding of global

**Table 4.** Comparison of the precision accuracy for each model extension. The most accurate values for each dataset are shown in bold. Each value is the average of three experiments conducted to account for random number variation. A t-test was conducted to confirm the statistical significance of the accuracy between the baseline and the proposed method, and an asterisk (\*) is added if the p-value is less than 0.01.

Dataset	age 21–35		age 36–50		age 51–65		age 66–80	
Method	P@10	P@20	P@10	P@20	P@10	P@20	P@10	P@20
Baseline	15.19	21.06	13.96	20.22	13.93	20.73	14.42	22.49
CoHHGN (H)	15.24	21.13	14.10	21.13	13.98	20.69	14.24	22.56
CoHHGN (HS)	<b>15.95*</b>	<b>22.11*</b>	<b>14.66*</b>	<b>21.97*</b>	13.97	20.57	15.13*	<b>23.51*</b>
CoHHGN+	15.92*	<b>22.28*</b>	<b>14.75*</b>	<b>22.01*</b>	<b>15.16*</b>	<b>22.57*</b>	<b>15.55*</b>	<b>23.84*</b>

**Table 5.** Comparison of mean reciprocal rank accuracy for each model extension. The symbols attached to the values are the same as those in the table 4.

Dataset	age 21–35		age 36–50		age 51–65		age 66–80	
Method	M@10	M@20	M@10	M@20	M@10	M@20	M@10	M@20
Baseline	6.67	7.06	6.02	6.45	5.62	6.08	5.27	5.81
CoHHGN (H)	6.63	7.02	<b>6.02</b>	6.45	5.66	6.12	5.32	5.88
CoHHGN (HS)	6.77	7.19	5.94	<b>6.45</b>	5.65	6.11	5.69*	6.24*
CoHHGN+	<b>6.89*</b>	<b>7.32*</b>	5.93	6.42	<b>5.83*</b>	<b>6.34*</b>	<b>5.81*</b>	<b>6.37*</b>

graph features will improve the accuracy in a stable manner. For CoHHGN (H), although the accuracy is improved over the baseline in several datasets, no statistically significant differences are identified. However, extending the model to CoHHGN (HS), which also considers session attributes, results in a significant difference in precision in all datasets, except for the age group 51–65.

Further, considering the mean reciprocal rank, although the recommendation accuracy tends to improve as the model is extended to CoHHGN (H) and CoHHGN (HS), the only dataset in which statistically significant differences can be confirmed is that for the 66–80 age group. However, when extended to CoHHGN+, which incorporates all the proposed methods, the overall prediction accuracy is higher and significant differences are confirmed. This confirms that the recommendation accuracy of the item ID can be improved by simultaneously considering features that vary between sessions and attributes of other sessions, in addition to features that vary within sessions.

## 7 Conclusion

In this study, we developed CoHHGN+ based on CoHHN, which is an SBR considering various features, and GCE-GNN considering global graphs, for purchase history data of EC sites. Moreover, we considered global time-series information, sale information, and user information. The application of the proposed model

to pseudo-session data with no user IDs shows that the GNN-based method exhibits significantly higher accuracy than those for the other methods, and that our proposed CoHHGN+ is the most accurate method on the dataset.

Although incorporating several types of data improves the prediction accuracy, there are still issues from the viewpoint of feature selection for data with more types of information recorded. If there are  $n$  types of heterogeneous information, the number of heterogeneous hypergraphs used to embed heterogeneous information is  $2^n$ . Therefore, selecting and integrating heterogeneous information remains an issue.

Future work on issues related to more efficient feature selection and methods for integrating heterogeneous information will lead to the development of models with even higher accuracy. We would also like to expand the scope of application of CoHHGN+ proposed in this study and attempt to provide useful recommendations in other domains as well.

**Acknowledgments.** We would like to thank the sponsor of the Data Analysis Competition, Joint Association Study Group of Management sScience (JASMAC), and Rakuten Group, Inc. for providing us with the data. This work was also supported by JSPS KAKENHI Grant Number JP20H04146.

## References

1. Rakuten ichiba homepage. <https://www.rakuten.co.jp/>. Accessed 1 May 2023
2. Greenstein-Messica, A., Rokach, L.: Personal price aware multi-seller recommender system: evidence from eBay **150**, 14–26 (2018)
3. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: surpassing human-level performance on ImageNet classification. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1026–1034 (2015)
4. Hidasi, B., Karatzoglou, A., Baltrunas, L., Tikk, D.: Session-based recommendations with recurrent neural networks. In: 4th International Conference on Learning Representations (2016)
5. Li, Y., Tarlow, D., Brockschmidt, M., Zemel, R.S.: Gated graph sequence neural networks. In: 4th International Conference on Learning Representations (2016)
6. Rendle, S., Freudenthaler, C., Schmidt-Thieme, L.: Factorizing personalized Markov chains for next-basket recommendation. In: Proceedings of the 19th International Conference on World Wide Web, pp. 811–820 (2010)
7. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, vol. 30 (2017)
8. Wang, Z., Wei, W., Cong, G., Li, X.L., Mao, X.L., Qiu, M.: Global context enhanced graph neural networks for session-based recommendation. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 169–178 (2020)
9. Wu, S., Tang, Y., Zhu, Y., Wang, L., Xie, X., Tan, T.: Session-based recommendation with graph neural networks. AAAI **33**(1), 346–353 (2019)
10. Zhang, X., et al.: Price does matter! modeling price and interest preferences in session-based recommendation. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1684–1693 (2022)