Intention Modeling from Ordered and Unordered Facets for Sequential Recommendation

Xueliang Guo
Beijing Institute of Technology
School of Computer Science
Beijing, China
guoxueliang@bit.edu.cn

Chongyang Shi*
Beijing Institute of Technology
School of Computer Science
Beijing, China
cy_shi@bit.edu.cn

Chuanming Liu
National Taipei University of Technology Computer Science and Information Engineering
cmliu@csie.ntut.edu.tw

ABSTRACT

Recently, sequential recommendation has attracted substantial attention from researchers due to its status as an essential service for e-commerce. Accurately understanding user intention is an important factor to improve the performance of recommendation system. However, user intention is highly time-dependent and flexible, so it is very challenging to learn the latent dynamic intention of users for sequential recommendation. To this end, in this paper, we propose a novel intention modeling from ordered and unordered facets (IMfOU) for sequential recommendation. Specifically, the global and local item embedding (GLIE) we proposed can comprehensively capture the sequential context information in the sequences and highlight the important features that users care about. We further design ordered preference drift learning (OPDL) and unordered purchase motivation learning (UPML) to obtain user's the process of preference drift and purchase motivation respectively. With combining the users' dynamic preference and current motivation, it considers not only sequential dependencies between items but also flexible dependencies and models the user purchase intention more accurately from ordered and unordered facets respectively. Evaluation results on three real-world datasets demonstrate that our proposed approach achieves better performance than the state-of-the-art sequential recommendation methods achieving improvement of AUC by an average of 2.26%.

CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

KEYWORDS

sequential recommendation, user intention, preference drift, purchased motivation

ACM Reference Format:
Xueliang Guo, Chongyang Shi*, and Chuanming Liu. 2020. Intention Modeling from Ordered and Unordered Facets for Sequential Recommendation. In *Proceedings of The Web Conference 2020 (WWW '20), April 20–24, 2020, Taipei, Taiwan.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3366423.3380190

WWW '20, April 20–24, 2020, Taipei, Taiwan

ACM ISBN 978-1-4503-7023-3/20/04.
https://doi.org/10.1145/3366423.3380190

1 INTRODUCTION

As important online services, recommendation system are widely used in the field of electronic commerce. Due to the explosion in commodity information, recommendation systems can effectively help customers to choose the commodities that suit their demands. However, when browsing e-commerce websites, most users will not indicate their purchase intention. As a result, accurately learning users' latent dynamic intention has become an important task of online recommendation systems.

Traditional recommendation systems, including content-based recommendation systems and collaborative filtering-based recommendations systems, assume that user-item interactions are mutually independent and model the users' preferences in a static way[18]. Zhou et al.[28] introduced the Deep Interest Network (DIN) for next-click prediction, which uses an attention-based model to capture interests related to the target item and obtain adaptive interest representation for recommendation. However, in real life, users' purchase behaviors are usually related to the previous purchase sequence, and users' preferences change dynamically over time. Recently, many sequence mechanism-based models have been proposed with the aim of learning the users' latent preferences, such as the Markov-chain model[6, 16] and the recurrent neural networks (RNN) model[12]. These approaches have attracted a lot of attention due to their ability to capture sequential dependencies in user-item interaction sequences. Rendle et al.[16] proposed factorizing personalized Markov chains to capture long-term preferences and short-term transitions at the same time. Yu et al.[24] proposed the DREAM model, which uses RNN to investigate the dynamic representation of each user and the sequential behaviors of user-item interactions. Zhou et al.[27] introduced an improved GRU approach to model the evolution of users' interests with the aim of addressing the problem of user interest drift.

Although existing approaches have achieved good performance on sequential recommendation tasks, there are still several factors to be considered if users' latent dynamic intentions are to be modeled; these are illustrated in **Figure 1**. First, affected by commodity attributes, different users may have similar purchase intentions, while other users' sequences may have an influence on current user intention. As illustrated by the dotted line ②, two users usually buy milk after buying bread and buy a phone case after buying a new phone. If only the local sequential dependencies are considered, it may not be possible to accurately understand the users' latent intention. Second, users will generate different intentions not only for different items, but also for different features of a specific item. Treating item features differently may facilitate a more

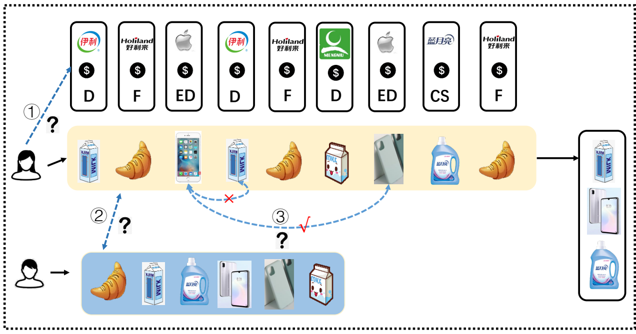


Figure 1: An illustrative example of sequential recommendation. Given the historical sequence, sequential recommendation predict whether a user will buy the next item. Dotted lines represent the possible relations.

accurate understanding of user intention. As shown by the dotted line ①, when choosing milk, users may have more intentions regarding the brand and do not care about the price, while when choosing a phone, the users may care more about the price. Third, user intention is flexible and changes over time; thus, it is important to consider users' dynamic preferences and flexible motivation simultaneously. Most existing methods for sequential recommendation only use a sequential mechanism-based model to capture the sequential dependencies, which results in the flexible dependencies between items being ignored. These methods overestimate the strength of the sequential dependencies between items, which may lead to noisy dependencies being generated and user intention being learned erroneously[18]. As illustrated by the dotted line ③, given the history sequence that the woman in the example recently purchased, it is obvious that there is a strong correlation between iphone and phone shell. If we pay too much attention to the sequential dependencies between the items, the correlation between phone and phone shell will be ignored.

Accordingly, inspired by the above three questions, we introduce an intention modeling from ordered and unordered facets (IMfOU) to learn the dynamic latent intentions of users, which also considers two kinds of context information: the sequential information and the periodic information regarding purchases. Specifically, we propose a novel item embedding method, named global and local item embedding (GLIE) using graph neural network and attention mechanism. This approach can capture the sequential context information comprehensively and highlight the items' features that users care about. Furthermore, we also design ordered preference drift learning (OPDL) and unordered purchase motivation learning (UPML) to obtain users' dynamic preferences and users' purchase motivations respectively. With combining the users' dynamic preference and current motivation, it considers not only sequential dependencies between items but also flexible dependencies and models the user purchase intention more accurately from ordered and unordered facets respectively. Finally, by matching users' purchase intention with the target item, the fusion layer is used to predict whether user will purchase the target item next.

We conduct extensive experiments on real-world datasets to evaluate our proposed approach. Evaluation results demonstrate that our proposed approach achieves better performance than state-of-the-art sequential recommendation methods, which improves AUC by an average of 2.26%. In summary, the main contributions of this paper are as follows:

- Proposing a novel item embedding method based on the sequence graph, i.e., global and local item embedding (GLIE). It not only considers the local sequential information, but also considers the global sequential dependencies of all sequences. In addition, with using user preference, it adaptively highlight the important features, allowing the users' different attention to be learned.
- Designing ordered preference drift learning (OPDL) and unordered purchase motivation learning (UPML) to model users' dynamic latent intentions. Specifically, OPDL performs PcGRU extended from GRU on the output of BiLSTM to model users' preference drift. UPML using attention mechanism obtains the flexible dependencies existing in user-item interactions, which effectively prevents the influence of noisy dependencies. With modeling users' latent intentions from ordered facet and unordered facet respectively, we can predict the items that user prefers effectively.
- Evaluating our proposed approach on three real-world datasets. The results suggest that our proposed IMfOU significantly outperforms than the state-of-the-art approaches. In addition, these results show that the two kinds of context information are highly beneficial for users' dynamic intention modeling.

2 RELATED WORK

2.1 Sequential Recommendation

In the real world, users' purchase behaviors usually happen in a sequence, and are often related to previously purchased items[18]. In recent years, researchers have focused on various sequential recommendation tasks, such as next-item recommendation[4], next-basket recommendation[16, 17] and session-based recommendation[10].

Early studies were typically based on the sequential pattern mining and the Markov chain models for sequential recommendation. The Markov chain-based approaches use a k-dimensional Markov chain to predict the next interaction based on k previous user-item interactions[2]. For example, Rendle et al.[16] proposed the factorizing personalized Markov chains for capturing long-term preferences and short-term transitions at in the same time. Moreover, He et al.[6] combined similarity-based models with high-order Markov chains to make personalized sequential recommendations.

Following the development of deep learning networks, many related approaches have been proposed for sequential recommendation. RNN-based approaches have been the most popular methods for sequential recommendation, due to RNN's excellent performance on sequential tasks. In most methods based on the RNN model[7, 15] the sequential dependencies of the given interactions are used to predict the next possible interaction[8]. Yu et al.[24] proposed the DREAM model, which uses RNN to investigate the dynamic representation of each user and the sequential behaviors of user-item interactions. In addition to the basic RNN, long short term memory (LSTM)[21] and gated recurrent unit (GRU)[8] also can capture the sequential dependencies effectively, as long-term and short-term user interests can be captured by their hidden layer states[20]. Zhu et al.[30] proposed an approach based on several time gates to model time intervals and capture the users' long-term and short-term interests. Zhou et al.[27] proposed AUGRU to model the evolution of users' interests. However, the overly strong order assumption employed in RNN limits RNN's application in sequences with a flexible order[18]. To avoid this problem, Yuan et al.[26] proposed a CNN-based approach to obtain the flexible sequential dependencies existing in user-item interactions. However, due to the limited sizes of the filters used in CNN, CNN-based methods can not capture long-term sequential dependencies effectively[18]. In this paper, we use period contextual information to model user dynamic intention from ordered and unordered facets, which captures the sequential dependencies and flexible dependencies in the same time.

2.2 Graph Neural Network

Due to the advantages of using a graph structure in information extraction, graph neural networks (GNNs) have attracted the attention of a large number of researchers. Many GNN approaches have been proposed and applied in many fields[9], such as social networking and[29] and knowledge graph[19]. Perozzi et al.[13] proposed DeepWalk, an unsupervised algorithm based on random walk, to obtain the united representations of graph nodes. Grover et al.[5] proposed another an other unsupervised model, node2vec, to represent the graph nodes.

Recently, graph neural networks have been used for recommendation systems. Binbin Hu et al.[9] proposed an attention-based graph neural network for top-N recommendation. It uses an attention mechanism to capture the context information that exists in the meth-path. Qitian Wu et al.[22] proposed a novel dual graph attention networks to collaboratively learn representations for two-fold social effects in recommend system. A session-based graph neural network was further proposed by Shu Wu et al.[23] and used to obtain the sequential information in a recommendation session. From the above as we can see, graph neural network have a good performance on sequential recommendation. In this paper, we use attention mechanism-based graph neural networks to capture the global sequential dependencies existing in sequences for sequential recommendation.

3 PRELIMINARIES

In this section, we first formulate the sequence recommendation task, then we introduce the process of constructing a global sequence graph. Finally, we formally define the user purchase cycle, which is used to represent the periodic context information when the user purchases an item.

3.1 Problem Definition

Sequential recommendation is designed to predict whether a given user will purchase the target products based on his historical interaction sequence. User-item interactions usually happen successively in a sequence, and each user has a corresponding historical sequence of interactions. The historical sequence S_u of user $u \in U$ can be represented as $S_u = \{(i_1, t_1, c_1), (i_2, t_2, c_2), ..., (i_t, t_t, c_t)\}$, where $i_j \in I$ denotes the j-th item purchased by current user, t_j represents the time at which user u purchases item i_j, and c_j is the category of item i_j. U and I represents the user set composed by all users and the item set composed by total items respectively. The global interaction sequence of all users is denoted as $H = \{S_1, S_2, ..., S_n\}$, where n denotes the number of current users.

Given user u and his historical interaction sequence S_u, the sequential recommendation task involves predicting whether the user will purchase item i_{t+1} next; this is also referred to as next-item recommendation.

3.2 Global Sequence Graph Construction

In the real world, user-item interactions usually do not happen in isolation, meaning that there are sequential relationships between user-item interactions. Accordingly, in this paper, we construct a global sequence graph G for all items in interaction sequence H; this can be denoted as $G = (N, E)$, where N represents the set of all nodes and E denotes the set of all edges between nodes. Each node represents an item, and the edges represent item-item interactions.

In addition, the frequency of interaction between items is different, and the implied sequential context is also different. If the interaction frequency of two items is high, this indicates that these two items usually have higher similarity and relevance; by contrast, if the interaction frequency between these items is low, it indicates that the relevance of these different items is weak, which will introduce noise into the sequential recommendation. Therefore, we further use the interaction times between items as the weight of the edge in the graph in order to represent the strength of the correlations between items.

The i-th node in the global sequence graph can be denoted as n_i, while the edge between node n_i and n_j can be represented as e_{ij}, where the weight of edge e_{ij} means the interaction times between node n_i and n_j.

Specifically, given a historical sequence $S_u = \{i_1, i_2, ..., i_t\}$ of user u, we can obtain the node set $N_{S_u} = \{n_1, n_2, ..., n_t\}$ and its adjacent node set $N^a(S_u) = \{N_1^a, N_2^a, ..., N_t^a\}$. The adjacent node set of node n_i and the adjacent edge set of node n_i can be denoted as $N_i^a = \{n_{i1}, n_{i2}, ..., n_{ik}\}$ and $E_i^a = \{e_{i1}^a, e_{i2}^a, ..., e_{ik}^a\}$ respectively. n_{ij} represents the j-th adjacent node of node n_i and e_{ij}^a denotes the edge between nodes n_i and n_{ij}. For example, given a purchase sequence $S_u = \{i_1, i_3, i_s, i_3, i_4\} \in \mathbb{R}^{5 \times L}$ of user $u \in U$, the graph of this session is shown in **Figure 2**. In this graph, n_1, n_s, n_3, and n_4 correspond to items i_1, i_s, i_3, and i_4 respectively, while nodes n_{sj}, n_{s1}, n_{s2} are the adjacent nodes of node n_s. The direction of the edge between two nodes represents the order of item purchase, while the number of edges between two nodes denotes the weight of this edge.

3.3 Purchase Cycle Modeling

In the real world, the same user will often buy different items with different frequency, while different users will buy the same item with different frequency. Accordingly, it is vital to consider the purchase cycle when a user purchases some item. Given a user and his historical interaction sequence S_u, the time interaction sequence TI_u of user u for different items can be obtained, which can be represented as $TI_u = \{TI(i_1), TI(i_2), ..., TI(i_t)\}$, where t is the number of items that user u has purchased. Moreover, the time interaction sequence of user u for item i_j can be expressed as $TI(i_j) = \{t_1, t_2, ..., t_q\}$, where q denotes the number of times that the user repeatedly purchases item i_j. Subsequently, the average purchase time $ap_u(i_j)$ interval between user u and item i_j can be calculated using Equation (1):

$$ap_j^u = \frac{t_q - t_1}{q - 1}. \tag{1}$$

In addition to the average time interval, the purchase cycle of items also depends on the category of the item. For example, the purchase cycle of electronics items is longer than than that of food items. In this paper, the purchase cycle of user u regarding item i_j is defined as follows:

$$\mathbf{P}_j = \mathbf{W}_{p1} ap_j^u + \mathbf{W}_{p2}\mathbf{C}_j^\top + \mathbf{b}_p. \tag{2}$$

Where $\mathbf{W}_{p1} \in \mathbb{R}^{d\times 1}, \mathbf{W}_{p2} \in \mathbb{R}^{d\times d}$ are vectors for transform matrices that can transfer three different properties to the same space, while $\mathbf{b}_p \in \mathbb{R}^d$ denotes the bias term and $\mathbf{C}_j \in \mathbb{R}^d$ represents the embedding of the category of item i_j. Based on the user's purchase cycle, the user's purchase frequency can be obtained to capture the change process of user preference and calculate the users' attention score for different items.

4 METHODS

In this section, we first provide an overview of our proposed IMfOU model for sequential recommendation. We then describe the three main components of IMfOU in detail: namely, global and local item embedding, latent user intention modeling and MLP based fusion layer. Finally, we analyze the loss function and training process of the model.

4.1 Overview

In this paper, we propose a novel model for sequential recommendation: intention modeling from ordered and unordered facets (IMfOU). **Figure 2** presents the structure of our proposed model. As shown in **Figure 2**, IMfOU contains three main layers: global and local item embedding (GLIE) layer, latent user intention modeling layer and fusion layer. Here, latent user intention modeling mainly consists of OPDL and UPML. With the GLIE, moreover, each item can be represented as a unique vector which can capture the global sequential contextual information of all users and highlight the important features that users have intention to purchase. Furthermore, at the latent user intention modeling layer, ordered preference drift learning (OPDL) and unordered purchase motivation learning (UPML) obtain the user's dynamic preferences and current purchase motivations respectively. With combining the ordered-based mechanism and unordered-based mechanism, it not only consider the sequential dependencies but also the flexible dependencies. Finally, the MLP-based fusion model combines users' dynamic preferences with the unordered purchase motivation to get the purchase intention of user, and predicts whether the current user will purchase the target project. The important notations we will use throughout the article are summarized in **Table 1**.

4.2 Global and Local Item Embedding

The purpose of the GLIE model is to generate a unified vector space for exploiting the global sequential context information between

Table 1: Notations and Explanations

Nitation	Explanation
u	The u-th user
v	The embedding of user u
i_j	The j-th item
c_j	The category of the j-th item
x_j	The embedding of item i_j
C_j	The embedding of category c_j
e_j	Node embedding of item i_j
e_j^F	The final embedding of item i_j
O_j	Weight matrix of item i_j's neighbor edges

items and highlighting the important features that users care about. Previous studies have generally used one-hot encoding to denote the user-item interaction, while the sequential relationship between items is extracted using subsequent neural network models such as RNN. However, one-hot encoding requires a lot of memory to operate and cannot adequately represent the latent sequential dependencies between items. On the other hand, some deep learning models like RNN only consider the sequential information existing in the local sequence.

In recent years, graph neural networks have been widely used in recommendation systems, such as social network analysis and Point of Interest (POI) recommendation, due to their excellent performance. In this paper, we propose a novel user intention-based global and local item embedding (GLIE) approach for sequential recommendation, which generates unified vectors to represent items by capturing the global sequential context information and the user intention of different features. The structure of GLIE is shown in **Figure 2**. In addition to the sequential context relationship that exists in the local user history sequence, GLIE also constructs the sequence graph by using the historical sequences of all users, from which the sequential relationships can be extracted more comprehensively for sequential recommendation.

As people have different intentions for different neighbors, each adjacency node has different importance to the representation of the target node. Here, we introduce the node-level attention mechanism designed to capture users' intention regarding different neighbors, as well as to aggregate the information of these neighbors to generate embedding vectors for target nodes.

Firstly, we leverage user information to learn the importance score, which represents the importance of different neighbors. Given an adjacent node n_{sj} included in the set of neighbors (N_s^a), the importance score $\mathbf{A}_s^{(im)} \in \mathbb{R}^{K\times 1}$ can be learned. Where K is the number of node n_s's neighbor nodes. This denotes how important node n_{sj} will be for the representation of node n_s. The importance score of the node pair(n_s,n_{sj}) is represented as follows:

$$\mathbf{X}_n = [\mathbf{x}_1; \mathbf{x}_2; ...; \mathbf{x}_K] \tag{3}$$

$$\mathbf{A}_s^{(im)} = \mathbf{W}_u^N \mathbf{v}^\top + \mathbf{W}_O^N O_s + \sigma(\mathbf{X}_n(\mathbf{X}_n)^\top)\mathbf{W}_s^N \tag{4}$$

Where $\mathbf{X}_n \in \mathbb{R}^{K\times d}$ represents the matrix of all neighbor nodes and $\mathbf{x}_j \in \mathbb{R}^d$ is the embedding of item i_j, which correspond to

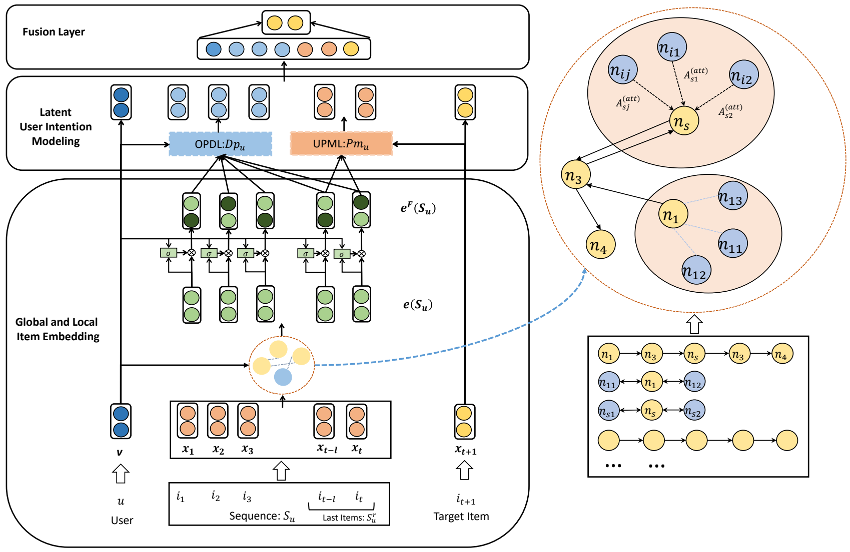


Figure 2: The structure of IMfOU. IMfOU consists of three layers: global and local item embedding, latent user intention modeling, and fusion layer. The schematic figure of the global sequence graph is shown at the right of this figure. The latent user intention modeling primarily consists of the OPDL and UPML modules.

the node n_j. $\mathbf{v} \in \mathbb{R}^d$ is the embedding of the given user u, which represents the user information we add. d is the dimension of input embedding. O_s is the weight matrix of neighbor edges, which means the times that sequential interaction has occurred between items i_s and it's neighbor nodes. $\mathbf{X}_n(\mathbf{X}_n)^\top$ is the interactive relationship of different items in one item sequence. $\mathbf{W}_u^N \in \mathbb{R}^{K\times d}$, $\mathbf{W}_s^N \in \mathbb{R}^{K\times 1}$ and, $\mathbf{W}_{co}^N \in \mathbb{R}^{K\times 1}$ denote spatial transfer matrix, which turn the embedding of user, the embedding of the weight of edge and the interactive relationship of different items into one space.

Next, the attention score $\mathbf{A}_s^{(att)} \in \mathbb{R}^{K\times 1}$ can be calculated via the $softmax$ function as follows:

$$\mathbf{A}_s^{(att)} = softmax(\mathbf{A}_s^{im}) = \frac{\exp(\mathbf{A}_s^{(im)})}{\sum_{k=1}^{K}\exp(\mathbf{A}_{sk}^{(im)})}, \tag{5}$$

Then, we obtain the node representation $\mathbf{e}_s \in \mathbb{R}^{3d}$ of the s-th node n_s with concatenating the local information $\mathbf{x}_s$ and the global information $\sum_{j=1}^{K} A_{sj}^{(att)}\mathbf{x}_j$.

$$\mathbf{e}_s = \mathbf{C}_s \oplus \mathbf{x}_s \oplus \sum_{j=1}^{K} A_{sj}^{(att)}\mathbf{x}_j. \tag{6}$$

Where K represents the number of neighbors of node n_s, $\mathbf{C}_s \in \mathbb{R}^d$ is the embedding of item i_s's category c_s, $A_{sj}^{(att)}$ represents the attention score corresponding to item i_j, and $\oplus$ denotes the connect operation.

For a specific item, users generate different attention for different features; they may only focus on specific features and ignore other features. In order to more accurately understand users' different attention for features, we use the purchase cycle and user information to differentiate the features of specific items, which highlights the features that users have intention to and ignores the features that users do not care about. Item i_s's feature attention score $\mathbf{score}_s^F \in \mathbb{R}^d$ is expressed as follows:

$$\mathbf{score}_s^F = \sigma(\mathbf{W}_u\mathbf{v}^\top + \mathbf{W}_P\mathbf{P}_s^\top + \mathbf{W}_I\mathbf{e}_s^\top + \mathbf{b}_F) \tag{7}$$

where $\mathbf{v}$ denotes the embedding of the user u, while $\mathbf{P}_i \in \mathbb{R}^d$ represents the purchase cycle for user u to item i_s. $\mathbf{e}_s \in \mathbb{R}^{3d}$ is the node representation of item i_s, σ is the *sigmoid* function, and $\mathbf{W}_u, \mathbf{W}_P \in \mathbb{R}^{d\times d}$, $\mathbf{W}_I \in \mathbb{R}^{d\times 3d}$, and $\mathbf{b}_F \in \mathbb{R}^d$ donate the learnable parameters.

Next, the the item embedding vectors through feature screening can be represented by:

$$\mathbf{e}_s^F = \mathbf{e}_s \otimes \mathbf{score}_s^F. \tag{8}$$

Where $\otimes$ denotes the element-wise product between matrices.

By using GLIE, the united embedding vectors of items can be obtained. Given a user u and his history sequence S_u, we can generate an interaction sequence after GLIE which can be represented as $\mathbf{e}(S_u) = \{\mathbf{e}_1^F, \mathbf{e}_2^F, ..., \mathbf{e}_t^F\} \in \mathbb{R}^{t\times d}$, where $\mathbf{e}_s^F \in \mathbb{R}^d$ denotes the d-dimensional embedding vector of item i_s and t represents the length of S_u.

4.3 Latent Intention Modeling

In the real world, users' purchase intention is mainly affected by their preferences and current consumption motivation. Based on the purchase cycle of users, we propose the ordered preference drift learning (OPDL) and unordered purchase motivation learning (UPML) to model user's latent intention. Users' latent purchase intentions are related with the purchase cycle; the same user will often buy different items with different frequency, while different users will buy the same item with different frequency. For example, the purchase frequency of food is significant higher than the purchase frequency of phone.

Users' preferences are dynamic and change over time; this change is greatly affected by the relationships of sequential interactions. OPDL is mainly based on the RNN model to obtain users' dynamic preferences. By using purchase cycle context information to obtain the user intention score for each item and update the hidden state in RNN, OPDL can comprehensively capture the sequential dependencies and resolve the problem of preference drift effectively. On the other hand, the current consumption motivation is generally transient and flexible, as well as less affected by the sequential dependencies. UPML using attention mechanism captures the flexible dependencies and effectively prevents the generation of nosiey dependencies between items.

4.3.1 Order Preference Drift Learning. Through capturing the sequential context information and the purchase cycle context information that exists in user-item interactions, Order Preference Drift Learning (OPDL) can be used to obtain the modeling of user dynamic preferences and capture the process of preference drift, the structure of which is presented in **Figure 3**. Given the purchase sequence S_u of user u, we can obtain the embedding of the user-item interaction sequence $\mathbf{e}(S_u) = \{\mathbf{e}_1^F, \mathbf{e}_2^F, ..., \mathbf{e}_t^F\} \in \mathbb{R}^{t\times d}$ and the cycle sequence $\mathbf{P}_u = \{\mathbf{P}_1, \mathbf{P}_2, ..., \mathbf{P}_t\} \in \mathbb{R}^{t\times d}$.

First, we can obtain a series of users' interest states for items by using BiLSTM to capture the sequential dependencies. Because BiLSTM can engage in forward learning and backward learning at the same time, it can make full use of long-term sequential dependencies and high-dimensional sequential dependencies. Specifically, the cell of BiLSTM is the same as the cell of LSTM, and the formulations of LSTM are as follows:

$$\begin{bmatrix} \mathbf{i}_j \\ \mathbf{f}_j \\ \mathbf{o}_j \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \end{bmatrix} (\mathbf{W}[\mathbf{h}_{j-1}; \mathbf{e}_j^F + \mathbf{b}]), \tag{9}$$

$$\hat{\mathbf{l}}_j = \tanh(\mathbf{W}_l[\mathbf{h}_{j-1}; \mathbf{e}_j^F] + \mathbf{b}_l), \tag{10}$$

$$\mathbf{l}_j = \mathbf{f}_j \otimes \hat{\mathbf{l}_{j-1}} + \mathbf{i}_j \otimes \hat{\mathbf{l}}_j, \tag{11}$$

$$\mathbf{h}_j = \mathbf{o}_j \otimes \tanh(\mathbf{l}_j). \tag{12}$$

where $\mathbf{i}_j, \mathbf{f}_j$ and $\mathbf{o}_j$ are gate activation, σ is *logistic sigmoid* function in [0, 1], and $\otimes$ means element-wise multiplication. BiLSTM uses three gates to control the state flow in each unit. At each interaction step j, the forward layer with current cell state $\overrightarrow{\mathbf{l}_j} \in \mathbb{R}^d$ and hidden state $\overrightarrow{\mathbf{h}_j} \in \mathbb{R}^d$ can be updated with the current item representation $\mathbf{e}_j^F$ and previous hidden state $\overrightarrow{\mathbf{h}_{j-1}} \in \mathbb{R}^d$. In the backward layer, moreover, the current cell state $\overleftarrow{\mathbf{l}_j} \in \mathbb{R}^d$ and hidden state $\overleftarrow{\mathbf{h}_j} \in \mathbb{R}^d$ are updated by the previous hidden state $\overleftarrow{\mathbf{h}_{j+1}} \in \mathbb{R}^d$ and item representation $\mathbf{e}_j^F$. Then we concatenate the forward state and backward state to calculate the hidden state $\mathbf{h}_j \in \mathbb{R}^{2d}$ of BiLSTM, which can capture the sequential dependencies to represent the preference of the user:

$$\mathbf{h}_j = concatenate(\overrightarrow{\mathbf{h}_{j-1}}, \overleftarrow{\mathbf{h}_{j+1}}). \tag{13}$$

However, the hidden state of BiLSTM only captures the sequential dependencies between items, and cannot learn the process of user preference drift and capture the dynamic preference accurately[27]. Under the influence of their environmental context, users have different preferences for items, which generally change over time. Therefore, with the help of user information and the user purchase cycle, we propose an improved GRU model named PcGRU to capture the process of preference drift and model the dynamic preferences of users. First, we calculate user u's intention score for different BiLSTM's hidden layer states, which is used to represent the user's intention for different items. Given the j-th BiLSTM's hidden layer state $\mathbf{h}_j$, its intention score $\mathbf{score}_j^h$ can be denoted as follows:

$$\mathbf{score}_j^h = \sigma(\mathbf{W}_u^h \mathbf{v}^\top + \mathbf{W}_1^h \tanh(\mathbf{W}_2^h (\mathbf{h}_j)^\top)). \tag{14}$$

where $\mathbf{W}_u^h, \mathbf{W}_1^h \in \mathbb{R}^{d\times d}$, and $\mathbf{W}_2^h \in \mathbb{R}^{d\times 2d}$ represents the learnable parameters.

Then, we take the hidden state of BiLSTM as the input of PcGRU. In a similar way to LSTM, PcGRU uses two gates to control the state flow in the PcGRU unit. At each time step j of PcGRU, the hidden state $\mathbf{g}_j \in \mathbb{R}^d$ can be updated with the previous hidden state $\mathbf{g}_{j-1} \in \mathbb{R}^d$, current input $\mathbf{h}_j \in \mathbb{R}^{2d}$, the intention score $\mathbf{score}_j^h \in \mathbb{R}^d$ and the user's purchase cycle $\mathbf{P}_j \in \mathbb{R}^d$ for item i_j as follows:

$$\mathbf{z}_j = \mathbf{score}_j^h \otimes \sigma(\mathbf{W}_1^z \mathbf{h}_j + \mathbf{W}_2^z \mathbf{P}_j + \mathbf{U}^z \mathbf{g}_{j-1} + \mathbf{b}^z), \tag{15}$$

$$\mathbf{r}_j = \sigma(\mathbf{W}_1^r \mathbf{h}_j + \mathbf{U}^r \mathbf{g}_{j-1} + \mathbf{b}^r), \tag{16}$$

$$\hat{\mathbf{g}_j} = \tanh(\mathbf{W}_1^g \mathbf{h}_j + \mathbf{r}_j \otimes \mathbf{U}^g \mathbf{g}_{j-1} + \mathbf{b}^g), \tag{17}$$

$$\mathbf{g}_j = (1 - \mathbf{z}_j) \otimes \mathbf{g}_{j-1} + \mathbf{z}_j \otimes \hat{\mathbf{g}_j}. \tag{18}$$

Where the $\mathbf{P}_j$ means the embedding of the item i_j's purchase cycle, $\otimes$ denotes the element-wise multiplication, $\mathbf{h}_j \in \mathbb{R}^{2d}$ is the hidden state of BiLSTM, and $\mathbf{g}_j \in \mathbb{R}^d$ is the j-th hidden state of PcGRU.

At this point, we can obtain the feature representation of user u's dynamic preference $\mathbf{D}_u \in \mathbb{R}^d$ by averaging the hidden states of PcGRU:

$$\mathbf{D}_u = average(\mathbf{g}_1, \mathbf{g}_2, ..., \mathbf{g}_t). \tag{19}$$

Where *average* means the average function, and t denotes the number of items user u has purchased.

4.3.2 Unordered Purchase Motivation Learning. Consumption motivation is generally transient and flexible, and is less affected by the sequential dependencies. Unordered purchase motivation learning(UPML) captures the implied flexible dependencies from the user's recent purchases, and thereby aims to obtain the representation of a user's current consumption motivation. The structure of UPML is shown as **Figure 4**.

We use an attention layer based on user information and the user purchase cycle to highlight those items that users care about. Given the set of items that the user has purchased recently, $S_u^r = \{(i_{t-l}, t_{t-l}), (i_{t-l+1}, t_{t-l+1}), ..., (i_t, t_t)\}$, we can obtain the embedding set of items $\mathbf{e}(S_u^r) = \{\mathbf{e}_{t-l}^F, \mathbf{e}_{t-l+1}^F, ..., \mathbf{e}_t^F\} \in \mathbb{R}^{l\times d}$ and the purchase cycle sequence $\mathbf{P}_u^r = \{\mathbf{P}_{t-l}, \mathbf{P}_{t-l+1}, ..., \mathbf{P}_t\} \in \mathbb{R}^{l*d}$. Where l is the number recent items we select to capture the purchase motivation. In UPML, the attention score $\mathbf{A}^I \in \mathbb{R}^l$ is calculated as

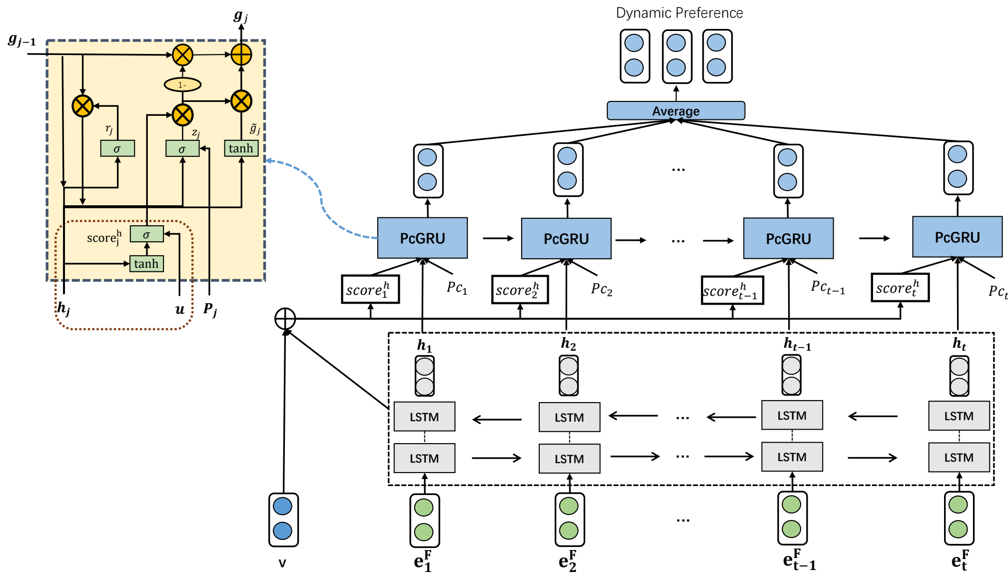


Figure 3: The structure of Order-Preference Drift Learning. OPDL mainly consists of the BiLSTM layer, the PcGRU layer, and the average layer. The structure of PcGRU's units is shown in detail on the left.

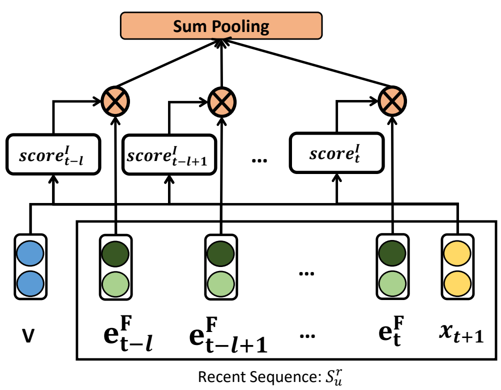


Figure 4: The structure of UPML.

follows:

$$\mathbf{E}^I = [\mathbf{e}^F_{t-l}; \mathbf{e}^F_{t-l+1}; \ldots; \mathbf{e}^F_t] \tag{20}$$

$$\mathbf{A}^I = softmax(\mathbf{W}^I_u \mathbf{v}^\top + \mathbf{W}^I_P \mathbf{P}^\top_s + \sigma(\mathbf{x}_{t+1}(\mathbf{E}^I)^\top)\mathbf{W}^I_s). \tag{21}$$

where $\mathbf{E}^I \in \mathbb{R}^{l \times d}$ represents the matrix of all recent items, $\mathbf{x}_{t+1} \in \mathbb{R}^d$ indicates the embedding of a target item that predicts whether user u will buy it, $\mathbf{e}^F_s \in \mathbb{R}^d$ is the embedding of item i_s through GLIE, and σ is the *sigmoid* function. $\mathbf{W}^I_u \in \mathbb{R}^{l \times d}, \mathbf{W}^I_P \in \mathbb{R}^{l \times d}, \mathbf{W}_s \in \mathbb{R}^{l \times l}$ denotes the learnable parameters. Then we can obtain the representation of the user's current purchase motivation $\mathbf{M}_u \in \mathbb{R}^d$ as follows:

$$\mathbf{M}_u = \sum_{s=l}^{t} A^I_s \mathbf{e}^F_s. \tag{22}$$

Where t is the number of items that user u has purchased, A^I_s is the attention score corresponding to item i_s.

4.4 Fusion Layer

After getting the 's dynamic preferences and purchase motivations, we can obtain the users' purchase intention $\mathbf{intent}_u \in \mathbb{R}^{2d}$ by concatenating the the user dynamic preference $\mathbf{D}_u \in \mathbb{R}^d$ and user purchase motivation $\mathbf{M}_u \in \mathbb{R}^d$, which is represented as $\mathbf{intent}_u = \mathbf{D}_u \oplus \mathbf{M}_u$. Then we use MLP to predict whether user u will purchase the target item i_{t+1}. In this paper, we turn the recommendation task into a binary classification problem, with the prediction results $\hat{y}_u$ being represented as follows:

$$\hat{y}_u = MLP(\mathbf{intent}_u \oplus \mathbf{x}_{t+1}) \tag{23}$$

where $\oplus$ is the connect operation and the $\mathbf{x}_{t+1}$ is the embedding of target item i_{t+1}. Then we select negative log-likelihood function as the loss function, which is defined as follows:

$$L = -\frac{1}{N} \sum_{i=1}^{N} y_i \log \hat{y}_u + (1 - y_i) \log(1 - \hat{y}_u). \tag{24}$$

where N is the total number of training instances, y_i is the label of instances. $y_i = 1$ means the user u will purchase item i_{t+1}, and $y_i = 0$ indicates the user u will not purchase item i_{t+1}.

5 EXPERIMENTS

In this section, we evaluate our proposed approach on three datasets for sequential recommendation.

5.1 Datasets

In this paper, we evaluate the proposed approach using three datasets derived from real-world applications, the statistics of which are

Table 2: Dataset statistics

Dataset	Users	Items	Category	Instance
Electronics	192k	63k	704	2993k
Movies	123k	50k	163	3147k
Clothes	39k	23k	484	279k

shown in **Table 2**. These datasets all come from Amazon, and thus vary significantly in terms of their domains and sparsity.

Amazon Datasets Amazon Dataset is a public dataset containing item reviews and metadata from Amazon. It is regularly used as a benchmark dataset for recommendation system. The metadata describe the information of items, i.e., category, price and item id, while the reviews describes the interaction information, i.e., user, item and time. We choose three subcategories to verify the performance of IMfOU, namely Electronics, Movies and Clothes, and sort the the user reviews by time to get the interaction sequence.

Given the interaction sequence, including t items($i_1, i_2, ..., i_k, ..., i_t$) and the $(t + 1)$-th item, the task is to predict whether user will purchase the $(k + 1)$-th item. Training dataset is generated with $k = 1, 2, ..., n - 2$ for each user. In the test set, given the item sequence $(i_1, i_2, ..., i_t)$ of user u, we aim to predict whether u will purchase the $(t + 1)$-th item.

5.2 Comparison Methods

We compare our proposed IMfOU with seven mainstream sequential recommendation methods, along with four variants of the IMfOU model. These approaches are briefly described below:

5.2.1 Baseline.

- ***Wide&Deep*** Wide&Deep[3] includes part of the Wide model and part of the Deep model. The Deep model is based on the MLP model for feature matching, while the Wide model is a linear model that uses manually designed cross-product features to better represent user-item interactions.
- ***PNN*** PNN[14] uses an embedding layer to learn a distributed representation of the categorical data, and a product layer to capture interactive patterns between inter field categories.
- ***DIN*** DIN[28] uses an attention mechanism to capture the importance of user behaviors for sequential recommendation and obtains an adaptive representation vector for user interests.
- ***CA-RNN*** CA-RNN[11] is based on a context-aware model and employs both adaptive context-specific input matrices and transition matrices to better capture the context information.
- ***NARM*** NARM[10] is an RNN-based model that employs an attention mechanism to capture the main purpose from hidden states and combines it with the sequential behavior as a final representation for sequential recommendations.
- ***DIEN*** DIEN[27] uses two GRU layers as the interest extracting layer and interest evolving layer respectively to model the user's sequential behaviors.
- ***BST*** BST[1] uses transformer model to capture the sequential relationship existing in users' behavior sequences.
- ***SLi-Rec*** SLi-Rec[25] uses a time-aware controller and the context-aware controller to control the state transition, and further proposes an attention-based framework to fuse users' long-term and short-term preferences.

5.2.2 Variants of the IMfOU model.

- ***IMfOU without GLIE*** IMfOU without GLIE is the IMfOU with the GLIE removed, which only use the embedding layer, and is used to prove that GLIE is beneficial to the sequential recommendation.
- ***GLIE+BiLSTM*** The GLIE+BiLSTM model is the approach that uses the BiLSTM to replace the OPDL and UMPL. It aims to prove that the latent user intention modeling is beneficial to the sequential recommendation.
- ***GLIE+UPML*** The GLIE+UPML uses global sequential item embedding to get the representation vectors of items and only capture the current purchase motivation by UPML.
- ***GLIE+OPDL*** The GLIE+OPDL model is the IMfOU with that the UPML removed, and is used to prove the UPML is beneficial to the sequential recommendation.

5.3 Experimental Settings

In the experiments, we use Tensorflow to implement the models. The experimental environment is a Linux server with Intel i9-9900k CPU and GTX2080Ti 11G GPUs. The latent dimension of all embedding vectors is set to 64 and the Optimizer is Adam. In our approach, we search the best performance in the following parameters. The learning rate is selected from $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$, and the L_2 regularization is selected from $\{1e^{-5}, 1e^{-4}, ..., 1e^{-2}, 1e^{-1}\}$. The batch size is 128. The max number of neighboring nodes is $\{3, 4, 5, 6, 7, 8\}$ and the number of recently purchased items is $\{0, 5, 10, 15, 20, 25\}$. The maximum length for user-item intention modeling is set to 450. The user embedding dimension is set to 64, and the other embedding dimension about item information also is 64.

5.4 Evaluation Metrics

In this paper, we turn the recommendation task into a binary classification problem to predict whether that user will buy the next item, and the output of our model has two states (buy or not buy). Therefore, we use AUC as an evaluation metric, which measures the probability that a positive instance will be ranked higher than a randomly chosen negative instance. In addition, we also use F1-score as an evaluation metric. F1-score is the harmonic mean of precision and recall, which is calculated as follows:

$$precision = \frac{TP}{TP + FP} \tag{25}$$

$$recall = \frac{TP}{TP + FN} \tag{26}$$

$$F_1 = 2 * \frac{precision * recall}{precision + recall} \tag{27}$$

Where TP represents the number of true positive predicts, FP represents the number of false positive predicts, and F_1 represents the number of false negative predicts.

Table 3: The results (AUC) of comparison with baselines

Model	Electronics	Movies	Clothes
Wide&Deep	0.7726	0.8412	0.7514
PNN	0.7741	0.8375	0.7492
DIN	0.7842	0.8383	0.7592
NARM	0.7849	0.8429	0.7572
DIEN	0.7876	0.8461	0.7568
CaRNN	0.8070	0.8540	0.7592
BST	0.8137	0.8442	0.6923
SLi-Rec	0.8243	0.8659	0.7675
IMfOU	**0.8560**	**0.8859**	**0.7830**

Table 4: The results (F_1) of comparison with baselines

Model	Electronics	Movies	Clothes
Wide&Deep	0.7103	0.7622	0.6724
PNN	0.7068	0.7531	0.6768
DIN	0.7096	0.7519	0.6835
NARM	0.7071	0.7552	0.6824
DIEN	0.7079	0.7631	0.6826
CaRNN	0.7112	0.7709	0.6808
BST	0.7339	0.7667	0.6403
SLi-Rec	0.7289	0.7865	0.6893
IMfOU	**0.7639**	**0.8022**	**0.6985**

5.5 Results and Discussion

We first present the results of comparison our proposed approach with the compared methods, which include the baseline methods and the IMfOU variant methods. We then discuss the number of recently purchased items. Finally, we visualize the intention score based on the purchase cycle in UPML, which is used to prove that the intention score really benefits the sequential recommendation.

5.5.1 Comparison with Baselines. To demonstrate that our proposed approach achieves better performance than the state-of-the-art methods, we compare the proposed approach with the baseline on the sequential recommendation task. The performances of different approaches in terms of AUC and F_1 are shown in **Table 3** and **Table 4** respectively. As we can see, DIN and PNN are models without the sequential mechanism, while DIN achieves better performances than PNN on most datasets. PNN considers the automative interaction between items and treats all items as equal. However, excepting for considering the interactions between items, DIN also captures the user's interest to obtain the attention score of different items. This demonstrates that using user intention to distinguish the attractiveness levels of different items is very helpful to sequential recommendation. NARM and DIEN capture the sequential dependencies through RNN, and thus perform better than PNN and DIN. Moreover, CaRNN and SLi-Rec significantly outperform NARM and DIEN, which demonstrates that the environment contextual information is beneficial to recommendation. However, the overly strong assumption of order employed in RNN limits the application of RNN in sequences with a flexible order. BST uses self-attention mechanism to capture the flexible dependencies, which obtains good performance on electrics dataset and Movies dataset, but obtains bad performance on Clothes dataset. From **Table 3** and **Table 4**, we can observe that IMfOU outperforms state-of-the-art methods in terms of two metrics. In a departure from previous studies, our proposed approach combine a sequential model (ODPL) and the nonsequential model (UPML) to capture the sequential and flexible dependencies. In addition, our proposed approach not only considers the sequential contextual information between items, but also considers the period contextual information existing in user-item interaction.

5.5.2 Comparison with Model Variants. We also compare our proposed approach with the variant IMfOU models, which aims to prove that the global and local item embedding (GLIE) and latent user intention modeling components are necessary and effective for sequential recommendation. **Table 5** and **Table 6** show the performance of the model variants in terms of AUC and F_1 respectively. From **Table 5** and **Table 6**, we can observe that the IMfOU is superior to the IMfOU without GLIE, and IMfOU achieves better performance than GLIE+OPDL. GLIE+OPDL model outperforms the GLIE+BiLSTM model, which means that the OPDL model we proposed is better than the BiLSTM model. The PcGRU is really beneficial to modeling user intention. IMfOU achieves significantly better performance than GLIE+OPDL, which demonstrates that the user purchase motivation captured by UPML has an influence on learning user purchase intention. GLIE+UPML achieves better performance than the GLIE+OPDL model for the Electronics and Movies dataset; however, the performance of GLIE+UPML on the Clothes dataset is lower than that of GLIE+OPDL. Through comparison with these variants, we can conclude that GLIE and latent user intention modeling we proposed are really beneficial to sequential recommendation.

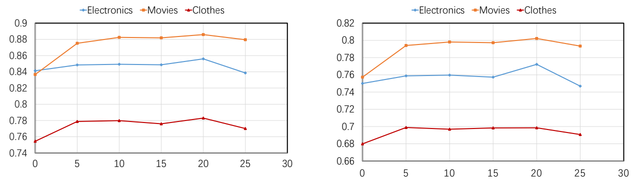
Table 5: The results (AUC) of the model variants

Model	Electronics	Movies	Clothes
IMfOU without GLIE	0.8503	0.8785	0.7700
GLIE+BiLSTM	0.8333	0.8300	0.7399
GLIE+OPDL	0.8413	0.8367	0.7544
GLIE+UPML	0.8496	0.8706	0.7251
IMfOU	**0.8560**	**0.8859**	**0.7830**

Table 6: The results (F_1) of the model variants

Model	Electronics	Movies	Clothes
IMfOU without GLIE	0.7597	0.7928	0.6882
GLIE+BiLSTM	0.7477	0.7430	0.6700
GLIE+OPDL	0.7500	0.7573	0.6799
GLIE+UPML	0.7611	0.7839	0.6611
IMfOU	**0.7639**	**0.8022**	**0.6985**

5.5.3 Comparison with Different Numbers of Recent Items. To obtain the best parameter of the number of items that UPML should process, we conduct experiments with different numbers of last



(a) In terms of AUC (b) In terms of F_1

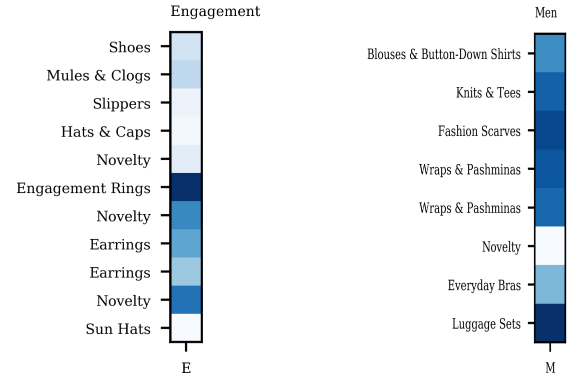
Figure 5: The Results of Comparison with Different number of Recent Items

items. The performance (AUC and F_1) of those experiments are shown in **Figure 5(a)**, and **Figure 5(b)** respectively. From these figures, we can observe that the results are a rise followed by a fall, and that we can obtain the best performance when the number of last items is 20. The results of Movies datasets and Electronics dataset is significantly greater than Clothes datasets, which may have been influenced by the length of sequences and the number of data points in the different datasets. In addition, when the number of last items is 5, the performance on the Clothes and Movies datasets is significantly greater than the model without UPML. This demonstrates that unordered purchase motivation has an important influence on the Clothes and the Movies datasets, but only a slight influence on the Electronics dataset.

5.5.4 Intention Score Visualization in UMPL. To prove that the intention score based on purchase cycle in UPML really is able to distinguish items with different importance for user intention modeling, we visualized the intention score of two users' historic sequence from the Clothes dataset. The result of these experiments is shown in **Figure 6**. For example, the categories of the first sequence of items are Shoes, Mules&Clogs, Slippers, Hats&Caps, Novelty, Earrings, Earrings, and Sun Hats. The target item category is Engagement. We can observe that user has more intention for Engagement Rings and has little intention on other things when a user is looking to purchase items under the Engagement category. The categories of the second sequence of items are Button-Down Shirts, Knits, Fashion Scarves, Wraps, Novelty, Everyday Bras, Luggage Sets, and the target item category is Men. In the real world, Men may have more purchase intention for business trips like Luggage Sets, and men have a very small probability of buying bras. Via the results of these two experiments, we can conclude that the purchase motivation we proposed is really beneficial to user intention modeling.

6 CONCLUSION

In this paper, we introduce a novel intention modeling from ordered and unordered facets (IMfOU) for user latent dynamic intention modeling. To understand the user intention more accurately, moreover, we propose global and local item embedding (GLIE) to capture the global sequential contextual information and highlight the important features that users have intention. Then, we design the ordered preference drift learning (OPDL) and the unordered purchase motivation learning (UPML) for modeling user latent intention, which obtains the dynamic preference and current purchase motivation respectively. By modeling users' dynamic preference and current motivation from ordered and unordered facets respectively, we can understand user intention more accurately. Finally, evaluation on three real-world datasets demonstrate that our proposed approach outperforms the state-of-the-art approaches and achieves good performance for sequential recommendation.



(a) Sequence 1 (b) Sequence 2

Figure 6: Intention score visualization in UMPL

ACKNOWLEDGMENTS

This work is supported by the National Key Research and Development Program of China(No. 2018YFB1003903), National Natural Science Foundation of China (No. 61502033,61472034, 61772071, 61272361 and 61672098) and the Fundamental Research Funds for the Central Universities.

REFERENCES

[1] Qiwei Chen, Huan Zhao, Wei Li, Pipei Huang, and Wenwu Ou. 2019. Behavior Sequence Transformer for E-commerce Recommendation in Alibaba. *arXiv preprint arXiv:1905.06874* (2019).

[2] Chen Cheng, Haiqin Yang, Michael R. Lyu, and Irwin King. 2013. Where You Like to Go Next: Successive Point-of-interest Recommendation. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence (IJCAI '13)*. AAAI Press, 2605–2611. http://dl.acm.org/citation.cfm?id=2540128.2540504

[3] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*. ACM, 7–10.

[4] Tim Donkers, Benedikt Loepp, and Jürgen Ziegler. 2017. Sequential user-based recurrent neural network recommendations. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*. ACM, 152–160.

[5] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 855–864.

[6] Ruining He and Julian McAuley. 2016. Fusing similarity models with markov chains for sparse sequential recommendation. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*. IEEE, 191–200.

[7] Balázs Hidasi and Alexandros Karatzoglou. 2018. Recurrent neural networks with top-k gains for session-based recommendations. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. ACM, 843–852.

[8] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939* (2015).

[9] Binbin Hu, Chuan Shi, Wayne Xin Zhao, and Philip S Yu. 2018. Leveraging meta-path based context for top-n recommendation with a neural co-attention model.

In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 1531–1540.

[10] Jing Li, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tao Lian, and Jun Ma. 2017. Neural attentive session-based recommendation. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM, 1419–1428.

[11] Qiang Liu, Shu Wu, Diyi Wang, Zhaokang Li, and Liang Wang. 2016. Context-aware sequential recommendation. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*. IEEE, 1053–1058.

[12] Chen Ma, Peng Kang, and Xue Liu. 2019. Hierarchical Gating Networks for Sequential Recommendation. *arXiv preprint arXiv:1906.09217* (2019).

[13] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 701–710.

[14] Yanru Qu, Han Cai, Kan Ren, Weinan Zhang, Yong Yu, Ying Wen, and Jun Wang. 2016. Product-based neural networks for user response prediction. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*. IEEE, 1149–1154.

[15] Massimo Quadrana, Alexandros Karatzoglou, Balázs Hidasi, and Paolo Cremonesi. 2017. Personalizing session-based recommendations with hierarchical recurrent neural networks. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*. ACM, 130–137.

[16] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized markov chains for next-basket recommendation. In *Proceedings of the 19th international conference on World wide web*. ACM, 811–820.

[17] Pengfei Wang, Jiafeng Guo, Yanyan Lan, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2015. Learning hierarchical representation model for nextbasket recommendation. In *Proceedings of the 38th International ACM SIGIR conference on Research and Development in Information Retrieval*. ACM, 403–412.

[18] Shoujin Wang, Liang Hu, Yan Wang, Longbing Cao, Quan Z. Sheng, and Mehmet Orgun. 2019. Sequential Recommender Systems: Challenges, Progress and Prospects. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. International Joint Conferences on Artificial Intelligence Organization, 6332–6338. https://doi.org/10.24963/ijcai.2019/883

[19] Xiang Wang, Dingxian Wang, Canran Xu, Xiangnan He, Yixin Cao, and Tat-Seng Chua. 2019. Explainable reasoning over knowledge graphs for recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 5329–5336.

[20] Bo Wu, Wen-Huang Cheng, Yongdong Zhang, Qiushi Huang, Jintao Li, and Tao Mei. 2017. Sequential prediction of social media popularity with deep temporal context networks. *arXiv preprint arXiv:1712.04443* (2017).

[21] Chao-Yuan Wu, Amr Ahmed, Alex Beutel, Alexander J Smola, and How Jing. 2017. Recurrent recommender networks. In *Proceedings of the tenth ACM international conference on web search and data mining*. ACM, 495–503.

[22] Qitian Wu, Hengrui Zhang, Xiaofeng Gao, Peng He, Paul Weng, Han Gao, and Guihai Chen. 2019. Dual Graph Attention Networks for Deep Latent Representation of Multifaceted Social Effects in Recommender Systems. In *The World Wide Web Conference*. ACM, 2091–2102.

[23] Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. 2019. Session-based recommendation with graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 346–353.

[24] Feng Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. A dynamic recurrent model for next basket recommendation. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. ACM, 729–732.

[25] Zeping Yu, Jianxun Lian, Ahmad Mahmoody, Gongshen Liu, and Xing Xie. 2019. Adaptive User Modeling with Long and Short-Term Preferences for Personalized Recommendation. In *International Joint Conferences on Artificial Intelligence (IJCAI)*. AAAI Press.

[26] Fajie Yuan, Alexandros Karatzoglou, Ioannis Arapakis, Joemon M Jose, and Xiangnan He. 2019. A Simple Convolutional Generative Network for Next Item Recommendation. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. ACM, 582–590.

[27] Guorui Zhou, Na Mou, Ying Fan, Qi Pi, Weijie Bian, Chang Zhou, Xiaoqiang Zhu, and Kun Gai. 2019. Deep interest evolution network for click-through rate prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 5941–5948.

[28] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 1059–1068.

[29] Xiangmin Zhou, Lei Chen, Yanchun Zhang, Dong Qin, Longbing Cao, Guangyan Huang, and Chen Wang. 2017. Enhancing online video recommendation using social user interactions. *The VLDB Journal—The International Journal on Very Large Data Bases* 26, 5 (2017), 637–656.

[30] Yu Zhu, Hao Li, Yikang Liao, Beidou Wang, Ziyu Guan, Haifeng Liu, and Deng Cai. 2017. What to Do Next: Modeling User Behaviors by Time-LSTM.. In *IJCAI*. 3602–3608.