Hybrid Interest Modeling for Long-tailed Users
Lifang Deng, Jin Niu, Angulia Yang, Qidi Xu, Xiang Fu, Jiandong Zhang, Anxiang Zeng
Alibaba Group, Beijing, China.

ABSTRACT
User behavior modeling is a key technique for recommender systems. However, most methods focus on head users with large-scale interactions and hence suffer from data sparsity issues. Several solutions integrate side information such as demographic features and product reviews, another is to transfer knowledge from other rich data sources. We argue that current methods are limited by the strict privacy policy and have low scalability in real-world applications and few works consider the behavioral characteristics behind long-tailed users. In this work, we propose the Hybrid Interest Modeling (HIM) network to hybrid both personalized interest and semi-personalized interest in learning long-tailed users' preferences in the recommendation. To achieve this, we first design the User Behavior Pyramid (UBP) module to capture the fine-grained personalized interest of high confidence from sparse even noisy positive feedbacks. Moreover, the individual interaction is too sparse and not enough for modeling user interest adequately, we design the User Behavior Clustering (UBC) module to learn latent user interest groups with self-supervised learning mechanism novelty, which capture coarse-grained semi-personalized interest from group-item interaction data. Extensive experiments on both public and industrial datasets verify the superiority of HIM compared with the state-of-the-art baselines.

KEYWORDS
Recommender Systems; Long-tailed User Modeling

1 INTRODUCTION
Benefit from the widespread usage of the Internet and mobile technologies, personalized recommender systems live at the heart of the industry, which aim to provide customized items for each user. User behavior modeling is a key technique for recommender systems in which historical interactions play a crucial role [27, 34, 37].

Most methods design complex model to capture user’s latent interest, a wealth of information rest in their abundant user interactions. User-item rich interactions prop up the accurate recommendation and good recommendation greatly decreases user churn rate. Such a running mode creates a positive feedback loop. However, most users do not express their interests explicitly (e.g., click, buy), the number of user’s interactions inherently follows a long-tailed distribution in real-world recommendation applications. Head users take their advantages in data richness to be well-served, while tail users are just the opposite. Such long-tailed users are silent majority in recommender systems, especially for those rising companies. Taking two recommender systems in the homepages of Taobao and Lazada for example (illustrated in Figure 1). Lazada is a rising e-commerce company in Southeast Asia. The user scale and interactions numbers are much smaller than Taobao. The visualized data difference is presented in Figure 2 (for privacy concern, we only visualize tendency and eliminate details), the average length of user behavior sequences in Taobao has a large increase than Lazada as time going. Long-tailed users dominate such rising e-commerce companies but are ill-served by the current user behavior modeling methods.

However, implementing high-performance recommendation for long-tailed users will also be full of challenges. Several solutions integrate side information such as demographic features and product reviews or to transfer knowledge from other rich data sources. We argue that current methods may have low scalability in real-world applications and raise privacy concerns. Another promising way is group recommendation. such methods capture coarse-grained semi-personalized interest from group-item interaction where user preferences on an unobserved item can be related to the group they belong to [3, 24]. But these methods either do not make use of a single user’s individual interactions to capture fine-grained personalized interest or need extra group information for initialization, making these methods hard to be deployed online.

In this paper, we propose a Hybrid Interest Modeling (HIM) network to improve the recommendation performance of long-tailed users in an end-to-end manner without the requirement of auxiliary data sources. To achieve this, we first take full consideration of behavior patterns behind long-tailed users. These users are a massive quick-changing group with few interactions and large intervals. We divide the long time interval interactions into multiple sessions and capture the users’ personalized interest and semi-personalized interest within each period. We first design the User Behavior Pyramid module (UBP) to capture fine-grained personalized interest. In this module, we consider user may click an item by mistake with no actual interest. This can bring a worse experience for long-tailed users than for head users due to fewer interactions. To conquer this, UBP takes both the positive(e.g., click) and negative(e.g., not click) interactions as inputs and models the differences between them to selectively extract intrinsic interest of high confidence from sparse even noisy positive interactions. The key idea is that interactions with higher confidence contribute more to the accurate recommendation. Moreover, individual interaction is too sparse and not enough for modeling user interest adequately. We further design the

Figure 1: Homepage recommendation in Taobao and Lazada.
User Behavior Clustering (UBC) module to learn latent user interest groups with self-supervised learning mechanism novelly in an end to end manner, which captures coarse-grained semi-personalized interest from group-item interaction data. HIM models user-item interactions as well as group-item interactions. The two learning processes reinforce each other and boost recommendation performance for long-tailed users.

To sum up, HIM contributes to the following aspects:

1. We find that existing recommendation models ignore the behavior characteristics of long-tailed users. We explore long-tailed users’ behavior patterns and reorganize interactions in an effective way.

2. We propose HIM to improve user behavior modeling for long-tailed users and two specially designed modules UBP and UBC can effectively reduce data sparsity and boost recommendation performance.

3. We validate on both public and industrial datasets and verify the superiority of HIM compared with the state-of-the-art baselines. It is notable that HIM has been deployed in Lazada online recommender systems and has obtained more than 7% item page view (IPV) improvements across multiple Southeast Asian countries.

To address this issue, existing methods generally integrate auxiliary information of different modalities such as user profiles, products review, title, or other item side information to boost recommendation performance with various deep model architectures [1, 11, 13, 17, 18, 36]. However, the accessibility of additional auxiliary information limits their scalability when they are deployed in different recommendation scenarios and raise privacy concerns.

2.2 Group Recommendation

Group recommendation received a lot of attention in recent years and has been widely applied in various domains. Most previous group recommendation methods rely on pre-defined users’ group information such as social relation [3, 19, 29, 35] to give recommendation to a group of people. AGGREG [3] exploits attention mechanisms with group embeddings. DBRec [24] uses a dual-bridge framework that initializes with pre-trained embeddings jointly integrate collaborative filtering, latent group discovery and hierarchical modeling tasks into a unified network. PreHash [32] defines anchor users and groups users into each bucket by a hashing network. However, these methods need to be well initialized, the group information or the anchor user is explicitly required, hence those methods cannot be scaled to the scenarios where the initialization information is not explicitly available.

3 HYBRID INTEREST MODELING (HIM)

Based on the sparse interactions dilemma for long-tailed users and deficiency of previous methods, we proposed Hybrid Interest Modeling (HIM) network. The model structure is shown in Figure 3. In the following subsection, we will illustrate each module of HIM in detail.

3.1 Reorganization Layer

In the e-commerce recommender system, users can click items, add items to cart, or buy items. We treat these behaviors actively performed by users as positive feedback. If users only browse the recommended item with no further action, we mark it as negative feedback which is usually ignored in previous user behavior modeling. Here we argue that negative feedback can potentially indicate items they are not interested in and can be used as supplementary information to alleviate the sparsity problem of positive feedback. Thus we collect both positive feedback and negative feedback sequences as user behavior input to enrich user interest learning.

As for interactions reorganization, we observe that interactions of long-tailed users have larger time interval and informative temporal dependency is rare, the interactions often shows session-based traits. Thus following the idea of session-based recommendation [16], we propose to divide the entire long behavior sequence into $T$ sessions. The specific length of the session $T$ is a hyperparameter that we can adjust to get the optimal performance by user behavior distribution.

Then, in each session, we collect one user’s interactions and re-rank them by interaction frequency instead of temporal order. We argue that more frequent interactions have greater confidence in indicating user authentic interest. For example, if a user clicks a mobile phone 8 times, the user may have a greater preference for
Auto-encoder to learn latent user interest groups by constructing a self-supervised mechanism based on the UBP personalized

The above-mentioned Reorganization Layer is the basis for mod-

the mobile phone than other items clicked once. Both the positive and negative feedback are reranked in frequently descending order. Besides, HIM adopts target item components and other related features as input as well, which is a commonly used pre-processing operation. We omit the details here.

3.2 User Behavior Pyramid (UBP)

The above-mentioned Reorganization Layer is the basis for modeling long-tailed users’ behavior characteristics. Based on the reorganization module, the User Behavior Pyramid (UBP) is built to further capture users’ interest within a session by modeling the correlation hidden between positive and negative feedback.

Within each session, \( E^i = \{ e^i_1, e^i_2, ..., e^i_n \} \in \mathbb{R}^{n \times d} \) represents the sequence embedding in which the user has interacted with in the \( i \)-th session. We select the top \( n \) most frequent interacted items as input. The value of \( n \) depends on different data distributions. \( d \) is embedding dimension. \( F^i = \{ f^i_1, f^i_2, ..., f^i_n \} \in \mathbb{R}^n \) corresponds to frequency information of \( E^i \). Here we enhance the confidence of interactions by multiplying each behavior sequence embedding by the corresponding frequency value:

\[
\hat{e}^i_j = f^i_j \star e^i_j, \quad j \in [1, n], i \in [1, T]
\]

where \( \star \) is scalar-vector product. Meanwhile, inspired by local activation unit [38] which adaptively calculate the activation stage of each behavior embedding, we develop a distance-based attention mechanism to model the relevance of each positive feedback towards compact negative feedback. Here, we first take a frequency weighted sum pooling on negative feedback to generate a fixed-length compact negative feedback representation \( \hat{e}^i_{neg} \) to reduce the noises within single negative feedback, since these negative feedback may be caused by various reasons besides dislike [15].

\[
\begin{align*}
\hat{e}^i_{pos} &= f^i_{pos} \star e^i_{pos}, \quad j \in [1, n], i \in [1, T], \\
\hat{e}^i_{neg} &= \text{pooling} \left( \hat{e}^i_{neg} \star e^i_{neg} \right), \\
\hat{d}^i_j &= \text{sim} \left( \hat{e}^i_{pos}, \hat{e}^i_{neg} \right)
\end{align*}
\]

\( \text{sim} \left( \hat{e}^i_{pos}, \hat{e}^i_{neg} \right) \) is the euclidean distance between the \( j \)-th positive feedback and compact negative feedback in the \( i \)-th session.
Different from the inner-product based attention mechanism, we manage to assign more weights to that positive feedback which is less similar to negative feedback. Then we apply the softmax function to get a normalized attentive weight \( \alpha_j^i \) and multiply it with the frequency weighted positive embedding:

\[
\alpha_j^i = \frac{\exp \left( a_j^i \right)}{\sum_{k=1}^{n} \exp \left( a_k^i \right)}
\]

\[
e_{j_pos}^i = \alpha_j^i \cdot e_{j_pos}^i
\]

(3)

To get an enhanced representation, we utilize GRU [8] to obtain the sequence embeddings. We feed positive feedback to the GRU unit in a descending order of frequency instead of temporal order, the gating mechanism within GRU captures the relationship between the high-frequency feedback and the low-frequency feedback in frequency order. Besides, compared with other recurrent models like RNN or LSTM, GRU is computationally more efficient. Next, we flatten each hidden state of positive feedback to get a fixed-size positive embedding:

\[
p_{pos}^i = \text{flatten} \left( \text{GRU} \left( e_{j_pos}^i \mid j = 1, 2, \ldots, n \right) \right)
\]

(4)

For each session, we follow the same scheme and compute concurrently. Then we split into two computing branches. On one branch, \( p^i_{pos} \) is fed into the self-attention unit [31] to get aggregated personalized representation \( p^i_z \) across all sessions, \( p^i_z \) is computed as the weighted sum of linearly transformed input elements, weight coefficient \( \alpha_j^i \) is computed using a softmax function:

\[
p^i_z = \sum_{t=1}^{T} \alpha_j^i (W^i t_{pos}^i)
\]

\[
\alpha_j^i = \frac{\exp (w_j^i)}{\sum_{j=1}^{T} \exp (w_j^i)}
\]

where \( w_j^i \) is computed using a compatibility function that compares two input elements \( p^i_{pos} \) and \( p^i_z \) correspondingly. On the other branch, we feed the two UBP embedding vectors \( p^i_{ac} \) and \( p^i_{ac} \) with a progressive relationship into the UBC module to help to construct latent user groups in each session.

### 3.3 User Behavior Clustering (UBC)

User Behavior Clustering (UBC) is another module we designed to further release sparsity, where the individual preference for an unseen item can be referred from the users within the same latent group who have interacted with the item. Since UBC can models the interest of a group of people, UBC can be seen as a coarse-grained semi-personalized interest modeling module.

Auto-encoder (AE) is one special category of Deep learning methods that compress the data into a dense code and then map the code into the reconstruction of the original input. The appeal of AE lies in the fact that they can learn representations in a fully unsupervised way. However, empirical experience tells that learning group embeddings solely from un-preprocessed input features by AE may not promise robust performance [24]. We deal with this dilemma very tactfully by the two progressive representations learned from UBP as input and constructing a heuristic self-supervised AE to enhance the clustering.

Refer to Figure 3(b), we learn user group embeddings \( G^u \in \mathbb{R}^{k \times d} \) within each session, group number \( k \) is a rather smaller number compared to the number of users, \( d \) denotes the dimension of group embedding. Recall the intermediate representation \( p_{ac}^u \) and self attention unit output \( p_{ac}^u \) in UBP, two representations share same feature dimension, meanwhile \( p_{ac}^u \) is a higher order representation compared to \( p_{ac}^u \), thus certain progressive relationship hold.

In UBC, we initialize AE with random weights, then make it learn effective group embeddings from scratch. In the intermediate layer, we set the representation dimension to be \( k \), which is exactly equal to the number of user groups, each value of the vector denotes the probability of user belonging to each group, the learned reconstruction representation \( p_{ac}^u \) after AE compute as follows:

\[
\beta^i = \text{softmax} \left( W_c p_{ac}^i + b_c \right)
\]

\[
\mu^i = \sum_{k=1}^{k} \beta^i_k g^k
\]

\[
p_{ac}^i = \sigma \left( W_z^i \mu^i + b_r^i \right)
\]

(6)

where \( \sigma \) is the sigmoid activation function, \( W_{ac}^i \in \mathbb{R}^{k \times d} \), \( W_z^i \in \mathbb{R}^{d \times d} \), \( b_c \in \mathbb{R}^{k} \), \( b_r \in \mathbb{R}^{d} \), \( \beta^i_k \) is the \( s \)-th dimension of vector \( \beta^i \), \( g^k \) is the \( s \)-th row of group embedding matrix \( G^u \). Reconstructed user representation \( p_{ac}^i \) will be used for the follow-mentioned reconstruction loss in the UBC. The learned semi-personalized representation \( c^i_z \) as follows:

\[
c^i_z = g^j \]

\[
j = \arg \max \left( \beta^i_k \right), s \in \{ 1, k \}
\]

(7)

where \( j \) is the label of user group that user has the maximum activation, \( g^j \) is the corresponding group embedding in \( G^u \).

The intuition behind self-supervision is that users with similar personal preferences are more likely to belong to the same group. So in the learning process, the learned reconstruction representation \( p_{ac}^i \) from \( p_{ac}^i \) after group embeddings projection should close to the high-level personalized representation \( p_{ac}^i \) to keep the consistency of individual interest learning and group interest learning.

In loss construction stage, we deploy the contrastive max-margin objective function that is commonly used in previous work [20, 33] to minimize the distance between \( p_{ac}^i \) and \( p_{ac}^i \):

\[
L_g = \sum_{i=1}^{T} \sum_{j=1}^{P} \max \left( 0, 1 - p_{ac}^i p_{ac}^j + p_{ac}^i p_{ac}^j \right)
\]

(8)

Where \( L_g \) is defined as hinge loss that maximize the cosine similarity between \( p_{ac}^i \) and \( p_{ac}^i \) and simultaneously minimize that between \( p_{ac}^i \) and negative samples, \( p_{ac}^j \) represent negative embeddings where we randomly select \( p \) users as negative users.

### 3.4 Prediction Making Layers

Up to now, we obtain user behavior feature embeddings including individual-based \( p^i \) and group-based \( c^i_z \) which concatenate each
To model the mutual information between the target item and interactions even the user have few user-item interactions. embeddings are shared for learning group-item and user-item inter-

ized interest modeling and semi-personalized interest modeling.

be represented as:

\[ L = \text{including group loss} \]

\[ L = \exp(p_z^T W_p e_t) + \exp(c_z^T W_c e_t) \times p_z, \]

\[ e_p = \exp(p_z^T W_p e_t) \]

\[ e_c = \exp(c_z^T W_c e_t) \times \exp(p_z^T W_p e_t) \times e_c \]

where \( W_p \in \mathbb{R}^{d_p \times d_t}, W_c \in \mathbb{R}^{d_c \times d_t}, d_p \) is the dimension of \( p_z, d_c \) is the dimension of \( c_z, d_t \) is the dimension of \( e_t \). The attention-

weighted user behavior feature, target item component feature together with other context and cross features are concatenated together and fed into a multilayer perceptron (MLP) to get two probabilistic logits, which indicate the probability of this sample together and fed into a multilayer perceptron (MLP) to get two


together with other context and cross features are concatenated

Overall representation vector for the target item. Here, we apply the common-used dot-product attention mechanism:

\[ e_p = \frac{\exp(p_z^T W_p e_t)}{\exp(p_z^T W_p e_t) + \exp(c_z^T W_c e_t)} \times p_z, \]

\[ e_c = \frac{\exp(c_z^T W_c e_t)}{\exp(p_z^T W_p e_t) + \exp(c_z^T W_c e_t)} \times c_z \]

To the best of our knowledge, no public datasets with complete long-tailed user interactions have been released. Public dataset like Amazon usually lacks real negative feedback. So we also experiment on the industrial dataset, which is a sampling version of the whole data. The data is constructed by impression and click logs from Lazada online recommender system. Lazada as a growing company, set up an e-commerce business across multiple Southeast Asian countries. In experiment practice, train, validation, and test set are split along the time sequence, which is a traditional industrial setting. Table 1 lists the statistics.

**Experimental Setup** In HIM, we subdivide interactions into different sessions. Figure 4 shows the distribution of user interactions’ time interval in Public and Industrial datasets, respectively. We choose the time interval corresponding to approximately 10%, 30%, 50% distribution. For Musical Instruments, session list is \{14d, 6m, 12m, all\}, which contain users’ interactions during the last 14 days, interactions during the last 6 months to the last 14 days, interactions during the last 12 months to the last 6 months and earlier interactions, respectively. For Electronics dataset, session list is \{3m, 9m, 18m, all\}. For Industrial dataset, we select the user’s interactions in the last month, session list is \{3d, 7d, 14d, 30d\}. As previously stated, the user behavior pattern varies greatly. It’s inaccurate to report all users’ results uniformly when we highlight effectiveness on long-tailed users who have fewer interactions. Hence we divide users into three categories based on users’ interaction frequencies. For public datasets, tailed user means users’ behavior sequence length is shorter than 3; the behavior sequence of body user is between 3 and 5; the behavior sequence of head user is more than 5. For the industrial dataset, tailed user means users occurred in

\[ \sum_{(x,y) \in \mathcal{D}} [y \log \hat{y}_{uv} + (1-y) \log (1 - \hat{y}_{uv})] \]

\[ L_c = -\frac{1}{N} \sum_{(x,y) \in \mathcal{D}} \mathrm{log} \hat{y}_{uv} \]

\[ \mathcal{L} = \alpha \mathcal{L}_g + \mathcal{L}_c \]

| Dataset                  | Users | Items | Samples | tailed user ratio | body user ratio | head user ratio |
|--------------------------|-------|-------|---------|------------------|----------------|----------------|
| Musical Instruments      | 51,253| 14,194| 129,867 | 47%              | 34%            | 19%            |
| Electronics              | 622,308| 70,323| 1,589,018| 43%              | 35%            | 22%            |
| Industrial (sampling data)| 3 million | 4 million | 0.1 billion | 55%            | 19%            | 26%            |
7 and 15 days; the head user is more than 15 days. As for perfor-
Industrial datasets respectively. For a fair comparison, the LR model
Table 2 shows the results on Public and
these methods keep the same with the original work.

4.2 Compared Methods
In the experiment, we conduct ablation study to
further discuss the contribution of each module in HIM. Since there is no authentic negative feedback in the Public datasets, the UBP
module learned in the public datasets is a variant version that ex-
cludes the modeling of negative feedback, which refers to positive
feedback sequence and
includes the modeling of negative feedback, which is the pro-
long-tailed users’ behavior characteristics better, capture
user interest with limited behaviors more accurately.

Ablation Study In this section, we conduct ablation study to
further discuss the contribution of each module in HIM. Since there is no authentic negative feedback in the Public datasets, the UBP
module learned in the public datasets is a variant version that ex-
cludes the modeling of negative feedback, which refers to positive
feedback sequence and
includes the modeling of negative feedback, which is the pro-
long-tailed users’ behavior characteristics better, capture
user interest with limited behaviors more accurately.

Ablation Study In this section, we conduct ablation study to
further discuss the contribution of each module in HIM. Since there is no authentic negative feedback in the Public datasets, the UBP
module learned in the public datasets is a variant version that ex-
cludes the modeling of negative feedback, which refers to positive
feedback sequence and
includes the modeling of negative feedback, which is the pro-
long-tailed users’ behavior characteristics better, capture
user interest with limited behaviors more accurately.

Ablation Study In this section, we conduct ablation study to
further discuss the contribution of each module in HIM. Since there is no authentic negative feedback in the Public datasets, the UBP
module learned in the public datasets is a variant version that ex-
cludes the modeling of negative feedback, which refers to positive
feedback sequence and
includes the modeling of negative feedback, which is the pro-
long-tailed users’ behavior characteristics better, capture
user interest with limited behaviors more accurately.

Ablation Study In this section, we conduct ablation study to
further discuss the contribution of each module in HIM. Since there is no authentic negative feedback in the Public datasets, the UBP
module learned in the public datasets is a variant version that ex-
cludes the modeling of negative feedback, which refers to positive
feedback sequence and
includes the modeling of negative feedback, which is the pro-
long-tailed users’ behavior characteristics better, capture
user interest with limited behaviors more accurately.

Ablation Study In this section, we conduct ablation study to
further discuss the contribution of each module in HIM. Since there is no authentic negative feedback in the Public datasets, the UBP
module learned in the public datasets is a variant version that ex-
cludes the modeling of negative feedback, which refers to positive
feedback sequence and
includes the modeling of negative feedback, which is the pro-
long-tailed users’ behavior characteristics better, capture
user interest with limited behaviors more accurately.
Table 2: Results of Different Model(AUC)

| Datasets   | Model   | all   | tailed user | body user | head user |
|------------|---------|-------|-------------|-----------|-----------|
| Musical    | Popularity | 0.798 | 0.796       | 0.800     | 0.807     |
|            | LR      | 0.813 | 0.810       | 0.814     | 0.823     |
|            | BaseModel | 0.817 | 0.809       | 0.820     | 0.835     |
|            | GRU4Rec | 0.817 | 0.812       | 0.818     | 0.830     |
|            | DIEN    | 0.820 | 0.813       | 0.826     | 0.836     |
|            | DBRec   | 0.827 | 0.823       | 0.825     | 0.840     |
|            | HIM     | 0.831 | 0.825       | 0.833     | 0.848     |
| Electronics| Popularity | 0.811 | 0.811       | 0.812     | 0.813     |
|            | LR      | 0.867 | 0.866       | 0.869     | 0.869     |
|            | BaseModel | 0.874 | 0.870       | 0.878     | 0.888     |
|            | GRU4Rec | 0.876 | 0.873       | 0.879     | 0.886     |
|            | DIEN    | 0.878 | 0.875       | 0.882     | 0.890     |
|            | DBRec   | 0.874 | 0.871       | 0.877     | 0.884     |
|            | HIM     | 0.883 | 0.880       | 0.887     | 0.896     |
| Industrial | Popularity | 0.555 | 0.566       | 0.556     | 0.549     |
|            | LR      | 0.555 | 0.553       | 0.555     | 0.557     |
|            | BaseModel | 0.573 | 0.557       | 0.576     | 0.580     |
|            | GRU4Rec | 0.576 | 0.556       | 0.579     | 0.583     |
|            | DIEN    | 0.579 | 0.550       | 0.585     | 0.585     |
|            | DBRec   | 0.574 | 0.560       | 0.574     | 0.578     |
|            | HIM     | 0.607 | 0.610       | 0.612     | 0.608     |

Table 3: Results on Different Components of HIM

| Datasets | Components   | all   | tailed user | body user | head user |
|----------|--------------|-------|-------------|-----------|-----------|
| Musical  | BaseModel    | 0.817 | 0.809       | 0.820     | 0.835     |
|          | + UBP        | 0.826 | 0.820       | 0.827     | 0.843     |
|          | HIM (+ UBP&UBC) | 0.831 | 0.825       | 0.833     | 0.848     |
| Electronics | BaseModel   | 0.874 | 0.870       | 0.878     | 0.888     |
|          | + UBP        | 0.879 | 0.876       | 0.882     | 0.891     |
|          | HIM (+ UBP&UBC) | 0.883 | 0.880       | 0.887     | 0.896     |
| Industrial| BaseModel   | 0.573 | 0.557       | 0.576     | 0.580     |
|          | + UBP        | 0.604 | 0.606       | 0.609     | 0.605     |
|          | HIM (+UBP&UBC) | 0.607 | 0.610       | 0.612     | 0.608     |

and Industrial datasets, indicating that semi-personalized recommendation is a good supplement for personalized recommendation, especially for those long-tailed users.

**Effect of Confidence Modeling in UBP** We argue that there exists a random click in positive feedback. Thus, we design the euclidean distance-based attention to model the similarity between negative aggregated embedding and each positive item embedding. Refer to Figure 5(a), we visualize the distance for users of different sparsity. We have the following observations: First, the distance for tailed users is relatively small, the smaller euclidean distance reflects the higher uncertainty of positive behaviors. Second, for head users, distance varies largely among different sessions. This indicates that users’ behavior shifts across time, the session can help understand users’ preferences instead of modeling the whole behavior sequence directly.

**Effect of UBC** When we hybrid the embedding from UBP and UBC, we apply the target item embedding to automatically learn their weights. To study the effect of this attention and have a better understanding of UBC, we analyze the learned weight and drill down to the different types of users. As shown in Figure 5(b), UBP’s weight is consistently higher (higher than 0.7) than UBC’s weight (around 0.26), indicating that personalized embedding

Figure 5: Performance under different interaction sparsity: (a) shows the euclidean distance between positive and negative feedback, the smaller euclidean distance reflects the higher uncertainty of positive behaviors. (b) shows the weights allocation between UBP and UBC, users with higher sparsity have a higher UBC weight.
We have deployed the proposed HIM in Lazada online recommender system. This indicates that such semi-personalized group embedding is promising when the interaction is extremely sparse.

**Effect of hyperparameters** In this section, we further discuss the performance of HIM under different pre-specified user group number $k$, and the loss weight $\alpha$ for hybrid modeling in a limited set. Empirically, group number is varied amongst $[5, 10, 15, 20, 25, 30]$, and the $\alpha [0.0001, 0.001, 0.001, 0.1, 1.1, 10]$. As shown in Figure 6, we present the performance across the datasets. Figure 6(a) and Figure 6(b) shows the performance has a rapid degradation with group number $k = 30$ and loss weight $\alpha = 1$ on Amazon Musical instruments dataset. Overall, the performance is relatively stable under a wide-range choice and achieves the best performance when clustering into $5$ groups. In public datasets, the best performance achieves with $\alpha = 0.0001$ while for Industrial dataset is $0.1$. Our analysis result is proved to be related with the size of the dataset.

### 4.4 Online A/B testing

We have deployed the proposed HIM in Lazada online recommender scenario across different Southeast Asian countries, including Indonesia (ID), Malaysia (MY), Vietnam (VN), Thailand (TH). A standard A/B test is conducted online, we perform the online experiments for one month, and the average item page view (IPV) Gain of different user groups are reported. As shown in Table 4, all users’ IPV are improved. A higher IPV indicates that users are more willing to browse and click items on our platform. Especially for the tail users, the improvement is large, because HIM can well learn long-tailed users’ preference, which leads to more positive feedback.

| Country | all IPV Gain | tail user | body user | head user |
|---------|--------------|-----------|-----------|-----------|
| ID      | +7.2%        | +6.7%     | +7.5%     | +7.2%     |
| MY      | +8.6%        | +11.1%    | +9.6%     | +8.1%     |
| TH      | +7.5%        | +7.3%     | +8.1%     | +7.5%     |
| VN      | +10.5%       | +12.5%    | +12.7%    | +9.8%     |

In short, HIM is a successful practice for our online campaign and can be an instructive recommendation solution for other similar newborn e-commerce business.

**REFERENCES**

[1] Trapit Bansal, David Belanger, and Andrew McCallum. 2016. Ask the gru: Multitask learning for deep text recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 107–114.

[2] Alex Beutel, Ed H Chi, Zhiyuan Cheng, Hubert Pham, and John Anderson. 2017. Beyond globally optimal: Focused learning for improved recommendations. In Proceedings of the 26th International Conference on World Wide Web. 203–212.

[3] Da Cao, Xiangnan He, Lianhai Xiao, Yahui An, Chao Yang, and Richang Hong. 2018. Attentive group recommendation. In The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. ACM, 645–654.

[4] Jia Chen, Jiaxuan Mao, YiQun Liu, Xin Zhang, and Shaoping Ma. 2020. A Contest-Aware Click Model for Web Search. In WSDM ’20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3–7, 2020, James Caverlee, Xia (Ben) Hu, Mounia Lalmas, and Wei Wang (Eds.). ACM, 88–96.

[5] Jingyuan Chen, Hanwang Zhang, Xiangnan He, Lijiang Nie, Wei Liu, and Tat-Seng Chua. 2017. A entive collaborative filtering: Multimedia recommendation with item-and component-level attention. In Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval.

[6] Tianq Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 785–794.

[7] 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.

[8] Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555 (2014).

[9] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In Proceedings of the 10th ACM conference on recommender systems. ACM, 191–198.

[10] Paolo Cremonesi, Yehuda Koren, and Roberto Turrin. 2010. Performance of recommender algorithms on top-n recommendation tasks. In Proceedings of the fourth ACM conference on Recommender systems. ACM, 39–46.

[11] Wenjing Fu, Zhaohui Peng, Senzhang Wang, Yang Xu, and Jin Li. 2019. Deeply Fusing Reviews and Contents for Cold Start Users in Cross-Domain Recommendation Systems. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 94–101.

[12] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: A Factorization-Machine based Neural Network for CTR Prediction. In IJCAI ijcai.org, 1725–1731.

[13] Xiangnan He, Lin Liao, Hanwang Zhang, Lijiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In Proceedings of the 26th international conference on world wide web. International World Wide Web Conferences Steering Committee, 173–182.

[14] Xinzan He, Junfeng Pan, Ou Jin, Tianxing Hu, Bo Liu, Tao Xu, Yaxin Shi, Antoine Atallah, Ralf Herbrich, Stuart Bowers, et al. 2014. Practical lessons from predicting clicks on ads at facebook. In Proceedings of the Eighth International Workshop on Data Mining for Online Advertising. 1–9.
