A Unified Framework of Interaction-level Preference Ranking for Single-behavior and Multi-behavior Recommendation

Tung-Lin Wu[a], Wei Luok Ngu[b], Chih-Ming Chen[a], Chuan-Ju Wang[a], Ming-Feng Tsai[c,*]

[a]*Research Center for Information Technology Innovation, Academia Sinica, Taipei, 115201, Taiwan*
[b]*Data Science Degree Program, National Taiwan University, Taipei, 106319, Taiwan*
[c]*Department of Computer Science, National Chengchi University, Taipei, 116011, Taiwan*

Abstract

Learning representative user and item representations is the key to building effective recommender systems. However, mainstream studies emphasize node-level embeddings when constructing models, thereby overlooking the importance of interaction between nodes. Recent work has developed an interaction-level model, treating user-item interactions as optimization triplets rather than optimizing users and items individually. Nevertheless, the current interaction-level model has limited usage, as it was initially designed for single-behavior recommendation only. To address this limitation, we propose unified interaction-level preference ranking (UnifiedIPR), a unified interaction-level framework for multi-behavior recommendation, expanding upon previous work on interaction-level recommendation. Specifically, UnifiedIPR incorporates multi-behavioral information into the modeling process by learning user-specific and item-specific behavior embeddings for each type of behavior. The proposed method not only models multi-behavioral information in a more fine-grained way but also enables recommendations

---

*Corresponding Author

Email addresses: `tunglin.tlwu@gmail.com` (Tung-Lin Wu), `weiluokbear@gmail.com` (Wei Luok Ngu), `changecandy@gmail.com` (Chih-Ming Chen), `cjwang@citi.sinica.edu.tw` (Chuan-Ju Wang), `mftsai@nccu.edu.tw` (Ming-Feng Tsai)

for different behaviors by leveraging the designated behavior embeddings for users and items. Note that single-behavior recommendation is a particular case of multi-behavior recommendation; by setting the number of behaviors to one, UnifiedIPR also effectively provides recommendations for single-behavior tasks. Comprehensive experiments on ten public benchmark datasets demonstrate the effectiveness and efficiency of UnifiedIPR for both single-behavior and multi-behavior recommendation.

Keywords: recommender system, multi-behavior, single-behavior, preference ranking, representation learning

---

1. Introduction

With the rapid expansion of internet services, recommender systems have become a convenient tool for reducing user information overload and improving the user experience. Among various recommendation tasks, a popular approach to delivering practical recommendations is to learn the vector space of user and item embeddings. Such a node-level method brings item nodes closer to the user node in the vector space if the user prefers those items strongly. Research focusing on modeling recommendation tasks at the node level has thrived in the past decade. The success of these studies in academia and industry has prompted researchers to apply machine learning algorithms to various kinds of recommendation tasks. One notable research branch is multi-behavior recommendation, which leverages multiple behaviors in e-commerce platforms, such as click, add to cart, and purchase, to provide more accurate recommendations. These studies go beyond using a single behavior; they thoroughly explore information from multiple behaviors to learn the node-level vector space.

However, instead of characterizing the preferences of users and items with a node-level model, the interaction-level model [1] captures the similarity between user-item interactions, providing a more natural and intuitive way to distill useful information from data. Inspired by this idea, we propose unified interaction-level preference ranking (UnifiedIPR), a lightweight, interaction-level embedding learning framework for various recommendation tasks. In contrast to mainstream studies that treat nodes as basic training units, UnifiedIPR expands previous work [1], treating interactions as the primary training unit. This approach models the similarity between user-item interactions more naturally under a pairwise ranking framework. Furthermore, the most recent interaction-level method has limited usage because it utilizes only single-behavior data. To alleviate this issue, we base UnifiedIPR on multi-behavior recommendation, considering recommendation with only one behavior as a particular case. The resultant framework is more flexible and capable of handling both recommendation tasks. Specifically, if two users interact with the same (or a different) item under the same behavior, these interactions are clustered in the embedding space. Moreover, in the proposed framework, an interaction embedding (describing the relation between a user and an item under a specific behavior) comprises the associated user and item embeddings and the behavior embeddings corresponding to the user and the item. In this way, we exploit user and item preferences in a more fine-grained manner. Additionally, we present a simple yet effective approach to incorporating global behavior information to address the sparsity of less-frequent behaviors. Unlike many end-to-end recommendation models, UnifiedIPR generates a set of user and item embeddings and user and item behavior embeddings for recommendation. In practice, many calculations are completed offline or approximated by nearest-neighbor search. Notably, UnifiedIPR is a variation of Bayesian personalized ranking [2]. Thus our method boasts more efficient training than state-of-the-art deep learning-based methods. To summarize, the main contributions of this work are as follows:

- We propose unified interaction-level preference ranking (UnifiedIPR), which learns fine-grained behavior embeddings for single-behavior and multi-behavior recommendation.
- We conduct extensive experiments on ten real-world datasets: six for single-behavior recommendation and four for multi-behavior recommendation. The experimental results show that the proposed method is superior to other methods in predicting various recommendation tasks using a single unified model.
- We present an effective, efficient implementation with faster training time than recent state-of-the-art methods.[1]

[1]The source code will be available online at a GitHub repository upon publication.

2. Related Work

Much work has been done on recommendation systems, from conventional single-behavior recommendation to recent multi-behavior recommendation. Brief discussions follow.

Single-behavior recommendation: Single-behavior recommendation is the most common and extensively researched type of recommender system. Specifically, most single-behavior recommendation scenarios consist of a sparse matrix, as a user interacts with only a few items. A popular solution is collaborative filtering [3, 4, 5, 2, 6], which calculates the similarity of the given users or items and then filters out items unlikely to be favored by the user based on the similarity function. Recently, in lieue of collaborative filtering methods, many researchers represent users and items as vectors [7, 8, 9, 10, 11], using their embeddings to provide more accurate recommendations. For example, PinSage [7] combines random walks and a graph convolution network to generate item embeddings, whereas LightGCN [8] simplifies the GCN design, resulting in a more concise recommendation model. Some studies in turn use side information such as user reviews to address sparsity [12, 13, 14] and enhance recommendation performance. However, these are limited to leveraging a single type of behavioral data. If multiple behaviors exist in the dataset, they leave other informative behavioral data unexplored or unutilized.

Multi-behavior recommendation: Researchers have begun to better leverage multi-behavior data by considering different types of behavioral data when training models. Even so, most pioneering works treat a single behavior as the target behavior and optimize the model based on that behavior only [15, 16, 17, 18, 19]. For example, MBGCN [17] applies graph neural networks to learn user and item representations by optimizing the target behavior; upon recommendation, items are ranked for each user based on their similarity scores calculated with their representations against the user representation. More recently, some studies have applied multi-task learning to learn user and item embeddings as well as behavior embeddings for multi-behavior recommendation [20, 21, 22], where these different embeddings are jointly embedded into the modeling process, after which the user, item, and behavior embeddings are aggregated for prediction. Such designs enable recommendations with a unified model for different user behaviors. For example, GHCF [21] leverages graph neural networks to aggregate information concerning users, items, and behaviors to achieve state-of-the-art performance.

3. Problem Formulation

In real-world scenarios, users of online information systems interact with items in many ways. Take IMDb for an example: a user can interact with a movie by rating it. This situation, namely the single-behavior task, contains only one type of behavior: rating. Another example is the social website Reddit: users interact with a post by clicking, pushing, sharing, replying, and creating. This situation, namely the multi-behavior task, contains multiple types of behavior: click, push, share, reply, and create. In this study, we design a unified, interaction-level embedding learning framework to exploit better user and item relations for single-behavior and multi-behavior recommendation.

Definition 1 (User-item Interaction Graph). *Let U, I, and R denote the set of users, items, and relations, respectively. A user-item interaction graph is an indirect bipartite graph defined as $G(\mathcal{V}, \mathcal{E}, \psi)$, where $\psi(\cdot)$ is an edge-type mapping function $\psi : \mathcal{E} \rightarrow R$, $\mathcal{V}$ and $\mathcal{E}$ denote the sets of all nodes (i.e., $\mathcal{V} = U \cup I$) and all edges in the graph, respectively, and $(u, i) \in \mathcal{E}$ denotes an edge between a user $u \in U$ and an item $i \in I$.*

Given the user-item interaction graph $G(\mathcal{V}, \mathcal{E}, \psi)$ defined in Definition 1, our goal is to learn an embedding matrix $\Theta \in \mathbb{R}^{(n+m)\times k\times d}$, where $n = |U|$, $m = |I|$, $k = |R|$, and d denotes the embedding size. For each user u (or each item i), the model generates its personalized user embedding denoted as θ_u (or item embedding denoted as θ_i, respectively). Additionally, associated with each user (or item) is a relation embedding for each relation $r_i \in R$, denoted as $\theta_u^{r_i}$ (or $\theta_i^{r_i}$, respectively). Specifically, the relation can be either single-behavior (e.g., ratings) or multi-behavior (e.g., view, add to cart, or purchase). In the single-behavior task, $k = |R| = 1$, whereas in the multi-behavior task, $k = |R| > 1$. It is expected that with our unified learning framework, the learned embedding matrix Θ correctly encodes user-item interactions for recommendations. Furthermore, the proposed model enables us to recommend w.r.t. single or multiple behaviors by leveraging the designated behavior embeddings for users and items.

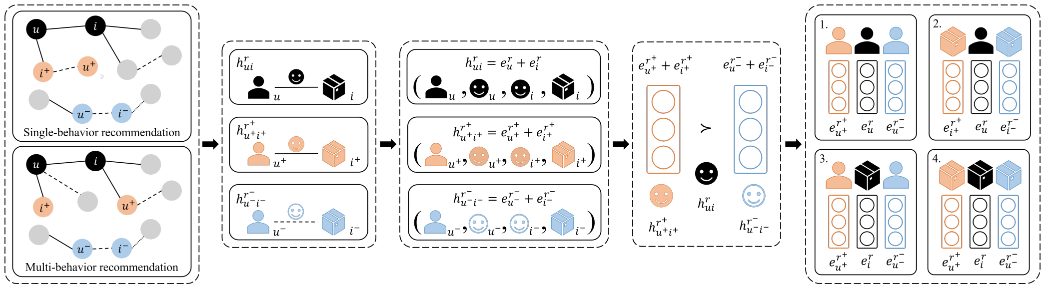


Figure 1: Overview of proposed UnifiedIPR framework

4. Methodology

To better exploit user and item relations for different recommendation tasks, we propose unified interaction-level preference ranking (UnifiedIPR), a unified embedding learning framework for single-behavior and multi-behavior recommendation (see Figure 1 for an overview of the framework). In this section, we first detail embedding learning for the proposed UnifiedIPR in Sections 4.1 and 4.2, after which we present a strategy to sample interaction triplets for optimization in Section 4.3. Then, we summarize the method with the procedure shown in Algorithm 1. Finally, we detail the scoring functions used for recommending items in Section 4.6.

4.1. Interaction-level Preference Ranking

The proposed UnifiedIPR models user and item interactions in a unified framework, generating a universal embedding matrix Θ for recommendation. We consider this a universal framework as the proposed model enables us to make the recommendation w.r.t. different recommendation tasks via the learned universal embedding matrix Θ. In the mainstream literature on recommender systems, node embeddings are used to capture relations between users and items via matrix factorization and derivative techniques (e.g., [23, 8, 24, 2, 25, 26]). Such methods, however, typically focus on modeling user and item nodes instead of the relations between and within various user behaviors; thus, they do not adequately leverage interactions of single-behavior and multi-behavior data.

We address this problem with UnifiedIPR, a pairwise interaction-level ranking algorithm for modeling the preferences of users and items. Inspired by [1], which changes the main idea of ranking-based recommendation algorithms from node-level [2] to interaction-level modeling and clusters similar user-item interactions in a self-supervised manner, we extend their work and further construct behavior embeddings for all users and items with different behaviors to facilitate the various recommendations.

Let $\mathcal{H}$ be the set containing all user-item interactions from $G(\mathcal{V}, \mathcal{E}, \psi)$, where each element $h^r_{ui} \in \mathcal{H}$ denotes a user-item interaction in which user u interacts with item i with relation r. Given an interaction h^r_{ui}, we define a basic training unit (also known as an interaction-level triplet) of the proposed UnifiedIPR as

$$(h^r_{ui}, h^{r^+}_{u^+i^+}, h^{r^-}_{u^-i^-}), \tag{1}$$

along with the relation $h^{r^+}_{u^+i^+} \succ_{h^r_{ui}} h^{r^-}_{u^-i^-}$ (see the second panel from the left in Figure 1). This relation denotes that the *positive interaction* $h^{r^+}_{u^+i^+}$ is "more alike" to h^r_{ui} than the *negative interaction* $h^{r^-}_{u^-i^-}$. Note that in this framework, $h^{r^+}_{u^+i^+}$ and $h^{r^-}_{u^-i^-}$ are built artificially regarding h^r_{ui} and do not necessarily appear in the graph $G(\mathcal{V}, \mathcal{E}, \psi)$; thus, they can be freely defined to correspond to different application scenarios. For general single-behavior and multi-behavior recommendation, we describe in Section 4.3 the proposed strategy to sample such triplets for optimization.

Given the triplet definition in Eq. (1), we then construct training data $D_{\mathcal{H}} : \mathcal{H} \times \mathcal{H}^+_{h^r_{ui}} \times \mathcal{H}^-_{h^r_{ui}}$ as

$$\begin{aligned} D_{\mathcal{H}} := \Big\{ \Big(h^r_{ui}, h^{r^+}_{u^+i^+}, h^{r^-}_{u^-i^-} \Big) \Big| \, & h^r_{ui} \in \mathcal{H} \wedge \\ & h^{r^+}_{u^+i^+} \in \mathcal{H}^+_{h^r_{ui}} \wedge h^{r^-}_{u^-i^-} \in \mathcal{H}^-_{h^r_{ui}} \Big\}, \end{aligned} \tag{2}$$

where $\mathcal{H}^+_{h^r_{ui}}$ ($\mathcal{H}^-_{h^r_{ui}}$) denotes the set of positive interactions (that of negative interactions, respectively) w.r.t. h^r_{ui}.

4.2. Embedding Matrix Learning

The pioneering work on interaction-level preference ranking [2] focuses on single-behavior recommendation, utilizing user and item embeddings to model different user-item ratings. Nevertheless, the learning algorithm they propose is difficult to generalize for both single-behavior and multi-behavior recommendation. To this end, in addition to the original user and item embedding (i.e., θ_u and θ_i), we further incorporate behavior embeddings for users and items to model different types of user-item interactions, resulting in $(\theta_u, \theta^r_u, \theta_i, \theta^r_i)$ for each user-item interaction. Thus (θ_u, θ^r_u) can be treated as a user u with his/her behavioral preference, whereas (θ_i, θ^r_i) denotes an item i with its behavior characteristics. For example, daily necessities are more frequently purchased than luxury goods; in our design, this is modeled properly via behavior embeddings for items.

Next, given an interaction h^r_{ui} (or a pseudo interaction $h^{r^+}_{u^+i^+}$ or $h^{r^-}_{u^-i^-}$), we define its embedding as $\mathbf{h}^r_{ui} := f(\theta_u, \theta^r_u, \theta_i, \theta^r_i)$, where $f(\cdot)$ is an arbitrary function to combine the user, item, and behavior embeddings. In this paper, as in [8], we adopt the addition operator as our aggregator, resulting in

$$\mathbf{h}^r_{ui} = \theta_u + \theta^r_u + \theta_i + \theta^r_i. \tag{3}$$

Note that single-behavior recommendation can be considered a special case of multi-behavior recommendation, wherein there is no need to learn θ^r_u and θ^r_i to differentiate between different behaviors. Therefore, we set θ^r_u and θ^r_i to $\overrightarrow{0}$ in this case, resulting in the simplification of $\mathbf{h}^r_{ui} = \theta_u + \theta_i$.

With $D_{\mathcal{H}}$, our objective is to find an embedding matrix Θ that maximizes the likelihood function from observed user-item interactions:

$$\mathcal{O}_{\text{UnifiedIPR}} = \prod_{t \in D_{\mathcal{H}}} p\left(h^{r^-}_{u^-i^-} \prec_{h^r_{ui}} h^{r^+}_{u^+i^+} \middle| \Theta\right), \tag{4}$$

where $t = (h^r_{ui}, h^{r^+}_{u^+i^+}, h^{r^-}_{u^-i^-})$. Furthermore, with the definition of the interaction embeddings in Eq. (3), the individual probability that an interaction $h^{r^+}_{u^+i^+}$ is more similar to h^r_{ui} than $h^{r^-}_{u^-i^-}$ is defined as

$$p\left(h^{r^-}_{u^-i^-} \prec_{h^r_{ui}} h^{r^+}_{u^+i^+} \middle| \Theta\right) = \sigma\left(\left\langle \mathbf{h}^r_{ui}, \mathbf{h}^{r^+}_{u^+i^+} - \mathbf{h}^{r^-}_{u^-i^-} \right\rangle\right), \tag{5}$$

where $\sigma(\cdot)$ denotes the sigmoid function and $\langle \cdot, \cdot \rangle$ denotes the dot product between two vectors.

With Eqs. (3)–(5), we formulate the maximum posterior estimator to derive the optimization criterion for the proposed UnifiedIPR as

$$\begin{aligned} \text{UnifiedIPR-OPT} &:= \ln p(\Theta | \prec_{h^r_{ui}}) \propto \ln p\left(\prec_{h^r_{ui}} |\Theta\right) p\left(\Theta\right) \\ &= \ln \prod_{t \in D_{\mathcal{H}}} p\left(h^{r^-}_{u^-i^-} \prec_{h^r_{ui}} h^{r^+}_{u^+i^+}\right) p(\Theta) \\ &= \sum_{t \in D_{\mathcal{H}}} \ln \sigma(\langle \mathbf{h}^r_{ui}, \mathbf{h}^{r^+}_{u^+i^+} - \mathbf{h}^{r^-}_{u^-i^-} \rangle) - \lambda_\Theta ||\Theta||^2, \end{aligned} \tag{6}$$

where λ_Θ is a model-specific regularization parameter.

To explore the advantages of such behavioral-based interaction triplets, we further decompose the interaction embedding $\mathbf{h}_{ui}^{r}$ into two components $\mathbf{e}_{u}^{r}$ and $\mathbf{e}_{i}^{r}$ for analysis, where $\mathbf{e}_{u}^{r} := \theta_u + \theta_u^r$ and $\mathbf{e}_{i}^{r} := \theta_i + \theta_i^r$ (see Eq. (3)). Then, the likelihood in Eq. (5) can be rewritten as

$$\begin{aligned}
&\sigma\left(\left\langle \mathbf{e}_u^r + \mathbf{e}_i^r, \left(\mathbf{e}_{u^+}^{r^+} + \mathbf{e}_{i^+}^{r^+}\right) - \left(\mathbf{e}_{u^-}^{r^-} + \mathbf{e}_{i^-}^{r^-}\right) \right\rangle\right) \\
&\quad = \sigma\left(\left\langle \mathbf{e}_u^r, \left(\mathbf{e}_{u^+}^{r^+} - \mathbf{e}_{u^-}^{r^-}\right) + \left(\mathbf{e}_{i^+}^{r^+} - \mathbf{e}_{i^-}^{r^-}\right) \right\rangle\right. \\
&\quad\quad \left. + \left\langle \mathbf{e}_i^r, \left(\mathbf{e}_{u^+}^{r^+} - \mathbf{e}_{u^-}^{r^-}\right) + \left(\mathbf{e}_{i^+}^{r^+} - \mathbf{e}_{i^-}^{r^-}\right) \right\rangle\right).
\end{aligned} \tag{7}$$

The above likelihood in Eq. (7) can be decomposed into four components:

1. $\left\langle \mathbf{e}_u^r, \mathbf{e}_{u^+}^{r^+} - \mathbf{e}_{u^-}^{r^-} \right\rangle$: Models user interaction similarity to user u with relation r regarding users u^+ with relation r^+ and u^- with relation r^-.
2. $\left\langle \mathbf{e}_u^r, \mathbf{e}_{i^+}^{r^+} - \mathbf{e}_{i^-}^{r^-} \right\rangle$: Models item interaction preference ranking between item i^+ with relation r^+ and i^- with relation r^- for user u with relation r.
3. $\left\langle \mathbf{e}_i^r, \mathbf{e}_{u^+}^{r^+} - \mathbf{e}_{u^-}^{r^-} \right\rangle$: Models user interaction preference ranking between user u^+ with relation r^+ and u^- with relation r^- for item i with relation r.
4. $\left\langle \mathbf{e}_i^r, \mathbf{e}_{i^+}^{r^+} - \mathbf{e}_{i^-}^{r^-} \right\rangle$: Models item interaction similarity to item i with relation r regarding items i^+ with relation r^+ and i^- with relation r^-.

The above components correspond to 1 to 4 in the rightmost panel of Figure 1. Moreover, for 1 (or 4), the model tends to cluster users (items) that involve similar interactions with items (users) in the embedding space; as for 2 and 3, the model tends to cluster users and items that involve similar interactions with each other in the embedding space. Such a design not only enables fine-grained modeling for different types of user-item interactions but also naturally yields a powerful representation matrix Θ that is suitable for various behavioral recommendation tasks.

4.3. Sampling Strategy

Recall that the positive interaction $h_{u^+i^+}^{r^+}$ and the negative equivalent $h_{u^-i^-}^{r^-}$ in a triplet given h_{ui}^{r} defined in Eq. (1) can be freely defined to correspond to different application scenarios. We list two strategies for two different cases: single-behavior recommendation and multi-behavior recommendation. The details are shown below.

For single-behavior recommendation, we follow previous work [1] in sampling such triplets to construct the training data $D_{\mathcal{H}}$ in Eq. (2). Specifically, for an interaction h^r_{ui} in $G(\mathcal{V}, \mathcal{E}, \psi)$, this is the set of positive interactions:

$$\mathcal{H}^+_{h^r_{ui}} := \left\{ h^{r^+}_{u^+i^+} \,\middle|\, (u^+, i^+), (u, i^+) \in \mathcal{E} \wedge \psi(u, i^+) = r^+ = r \right\}. \tag{8}$$

For general multi-behavior recommendation, we propose a strategy to sample such triplets. For an interaction h^r_{ui} in $G(\mathcal{V}, \mathcal{E}, \psi)$, this is the set of pseudo positive interactions:

$$\begin{aligned}\mathcal{H}^+_{h^r_{ui}} := \Big\{ h^{r^+}_{u^+i^+} \,\Big|\, & (u, i^+), (u^+, i) \in \mathcal{E} \wedge \\ & r^+ = r \wedge r^+ \in \psi(u, i^+) \wedge r^+ \in \psi(u^+, i) \Big\}. \end{aligned} \tag{9}$$

We illustrate this sampling strategy with Figure 1 (see the leftmost panel of the figure). Given an interaction between user u and item i with a specific relation r (the solid line between two black nodes), i.e., h^r_{ui}, we sample a positive interaction $h^{r^+}_{u^+i^+}$ constructed by a sampled positive item i^+ and a sampled positive user u^+ with relation r (both nodes are orange). Note that the positive item is a neighbor of u, but the positive user is a neighbor of i (i^+) for single-behavior (multi-behavior) recommendation. In this sampling strategy, the positive interaction for multi-behavior recommendation is a "pseudo" relation, as the sampled (u^+) and (i^+) may lack any interaction. For the negative interaction $h^{r^-}_{u^-i^-} \in \mathcal{H}^-_{h^r_{ui}}$, for simplicity, we randomly select an interaction from all interactions in $G(\mathcal{V}, \mathcal{E}, \psi)$ (i.e., $\mathcal{H}$) to construct $\mathcal{H}^-_{h^r_{ui}}$. Since single-behavior recommendation has only one behavior, different relations r imply different behavior magnitudes (e.g. rating 5 or rating 3), whereas different relations r of multi-behavior recommendation imply different behaviors (e.g. purchase or add to cart). We leave more complicated settings for future work.

4.4. Optimization

With the training data $D_{\mathcal{H}}$ in Eq. (2) and the objective function in Eq. (6), we optimize the embedding matrix as

$$\Theta \leftarrow \Theta + \alpha \left(\frac{\partial \text{PMi-OPT}}{\partial \Theta} \right), \tag{10}$$

where α is the learning rate. Specifically, for each given interaction $h^r_{ui} \in \mathcal{H}$, we randomly sample a positive interaction $h^{r^+}_{u^+i^+} \in \mathcal{H}^+_{h^r_{ui}}$ defined in Eq. (8)

or (9) and a negative interaction $h^{r^-}_{u^-i^-} \in \mathcal{H}$. The resulting interaction-level triplet $(h^r_{ui}, h^{r^+}_{u^+i^+}, h^{r^-}_{u^-i^-}) \in D_{\mathcal{H}}$ is adopted to update the model parameter matrix Θ with this gradient:

$$\begin{aligned}\frac{\partial \text{PMi-OPT}}{\partial \Theta} &= \frac{\partial}{\partial \Theta} \ln \sigma(\hat{x}) - \lambda_\Theta \frac{\partial}{\partial \Theta} ||\Theta||^2 \\ &\propto \frac{e^{-\hat{x}}}{1+e^{-\hat{x}}} \frac{\partial}{\partial \Theta} \hat{x} - \lambda_\Theta \Theta, \end{aligned} \tag{11}$$

where $\hat{x} := \langle \mathbf{e}^r_u + \mathbf{e}^r_i, (\mathbf{e}^{r^+}_{u^+} + \mathbf{e}^{r^+}_{i^+}) - (\mathbf{e}^{r^-}_{u^-} + \mathbf{e}^{r^-}_{i^-}) \rangle$.

Additionally, we follow [26, 23, 27] by utilizing asynchronous stochastic gradient descent (ASGD) [28] to efficiently update parameter matrix Θ in a parallel manner. Algorithm 1 details the complete model training procedure.

Algorithm 1 Training with proposed UnifiedIPR framework

Inputs: $G(\mathcal{V}, \mathcal{E}, \psi)$, N iterations
Output: Θ

1: Randomly initialize Θ
2: $\mathcal{H} \leftarrow$ all user-item interactions from $G(\mathcal{V}, \mathcal{E}, \psi)$
3: **for** $e = 1$ to N **do**
4: Draw user-item interaction h^r_{ui} from $\mathcal{H}$
5: Construct positive interaction set $\mathcal{H}^+_{h^r_{ui}}$
6: Construct negative interaction set $\mathcal{H}^-_{h^r_{ui}}$
7: Draw positive interaction $h^{r^+}_{u^+i^+} \in \mathcal{H}^+_{h^r_{ui}}$
8: Draw negative interaction $h^{r^-}_{u^-i^-} \in \mathcal{H}^-_{h^r_{ui}}$
9: Update Θ with Eqs. (10)–(11)
10: **end for**

4.5. Global Behavior Embedding for Multi-behavior Recommendation

To fully leverage multi-behavior information and account for the sparsity of less frequent behaviors, we additionally incorporate global behavior information in the multi-behavior user-item interaction graph $G(\mathcal{V}, \mathcal{E}, \psi)$. That is, if there exist ℓ different types of behavior in a multi-behavior recommendation dataset, we additionally create a pseudo behavior, namely the global behavior r_g, resulting in $|R| = \ell + 1 = k$ types of behaviors in the graph. Specifically, for any edge (u, i) between a user u and an item i in $\mathcal{E}$, $r_g \in \psi((u, i))$.

Table 1: Single-behavior dataset statistics

Dataset	Users	Items	Interactions	Sparsity
MovieLens-1M	6,040	3,706	1,000,188	95.5316%
AMZ I-S	11,039	5,328	123,312	99.7903%
AMZ PS	236,966	42,529	3,357,320	99.9667%
AMZ CP-A	157,186	48,178	1,805,498	99.9762%
AMZ VG	55,219	17,404	796,122	99.9172%
AMZ DM	16,563	11,777	271,648	99.8607%

Table 2: Multi-behavior dataset statistics

Dataset	Users	Items	Views	Carts	Purchases
Beibei	21,716	7,977	2,412,586	642,622	304,576
Taobao	48,749	39,493	1,548,126	193,747	259,747
Ecommerce	55,608	48,547	1,945,122	1,860,450	905,847
Rees46	20,399	31,972	112,652	48,313	47,368

In other words, there exists a global relation between user u and item i if u has interacted with i with any of the relations $r \in \{r|r \in R \wedge r \neq r_g\}$. The experimental results in Section 6 show the effectiveness of such a design.

4.6. Scoring Function

Algorithm 1 yields the embedding matrix $\Theta \in \mathbb{R}^{(n+m)\times k\times d}$. The learned Θ enables us to make single-behavior or multi-behavior recommendations. Specifically, for single-behavior recommendation at the inference stage, for each user u, we calculate the scores considering all items as $\hat{y}^r_{ui} = \theta_u \cdot \theta_i$. Likewise, for any target behavior $r \in \{r|r \in R \wedge r \neq r_g\}$ in multi-behavior recommendation, the scoring function is $\hat{y}^r_{ui} = \left(\theta_u + \theta^r_u \,\big\|\, \theta_u + \theta^{r_g}_u\right) \cdot \left(\theta_i + \theta^r_i \,\big\|\, \theta_i + \theta^{r_g}_i\right)$, where $a\big\|b$ denotes the concatenation of vectors a and b, and $r_g \in R$ is the global behavior (see Section 4.5). The items to be recommended are then obtained by ranking the items based on the score for each user.

5. Experiments

Here, we describe experiments conducted on several public datasets to demonstrate the effectiveness of the proposed UnifiedIPR framework. The experiments include two tasks: single-behavior recommendation and multi-behavior recommendation.

5.1. Datasets and Preprocessing

5.1.1. Single-behavior recommendation

We conducted experiments on six real-world datasets to evaluate the single-behavior recommendation algorithm of UnifiedIPR, including MovieLens-1M and five Amazon datasets [29]: Amazon Industrial and Scientific (AMZ I-C), Amazon Pet Supplies (AMZ PS), Amazon Cell Phones and Accessories (AMZ CP-A), Amazon Video Games (AMZ VG), and Amazon Digital Music (AMZ DM). We randomly split 80% of the interactions for training. The remaining interactions are the test data for evaluation. The statistics of all datasets are summarized in Table 1.

5.1.2. Multi-behavior recommendation

We conducted experiments on four public recommendation datasets to evaluate the UnifiedIPR multi-behavior recommendation algorithm. All datasets contain three common types of e-commerce behaviors, as summarized in Table 2. For Beibei, Taobao, and E-commerce, we followed the settings in [21], which filters out users and items with fewer than five purchase interactions. For the smaller Rees46 dataset, we filtered out users with fewer than two purchase interactions. For each dataset, we took each user's last purchase record as the test data for the purchase recommendation evaluation. We also took each user's last cart and view records as the test data for the cart and view behavior evaluations, respectively, for both Rees46 and Ecommerce. As Beibei and Taobao did not include timestamps for each interaction, we could not split the dataset for other behaviors such as view and cart; therefore, for these, we evaluated the recommendation performance only for the purchase behavior, which is available in the original datasets.

5.2. Baselines

To demonstrate the effectiveness of the UnifiedIPR framework, we compared it with several baseline methods. These baselines can be categorized into two groups: 1) single-behavior models that utilize only single-behavior data, and 2) multi-behavior models that consider all types of behavioral data in the training process.

5.2.1. Single-behavior models

- **MF** [30]: a widely used pairwise learning framework that considers node-level triplets for model training via a square loss objective function.

- **MFBPR** [2]: an MF method with the Bayesian personalized ranking (BPR) loss function for optimization.
- **NCF** [24]: a general version of MF that replaces the inner product with a neural architecture on the latent features of users and items.
- **NGCF** [25]: a recommendation framework that exploits the user-item graph structure by injecting the collaborative signal into the embedding process.
- **LightGCN** [8]: a simplified architecture of the graph neural network (GNN) from NGCF that usually achieves state-of-the-art performance for single-behavior recommendation.
- **CDAE** [31]: a deep learning-based model that formulates top-N recommendations with denoising auto-encoder frameworks.
- **DeepICF** [32]: item-based collaborative filtering that models higher-order relationships among items using nonlinear neural networks.

5.2.2. Multi-behavior models

- **MC-BPR** [18]: a multi-behavior recommendation algorithm that assumes an importance order between different behaviors and extends BPR [2] by building sampling pairs with a type of positive behavior and a type of weaker behavior.
- **NMTR** [33]: a neural model that involves joint optimization based on the multi-task learning framework, where the optimization on each behavior is treated as a task.
- **MBGCN** [17]: a graph convolutional network (GCN) that learns the strengths of different behaviors by the user-item propagation layer and the item-item propagation layer.
- **EHCF** [20]: a non-sampling transfer learning solution model good for modeling both single- and multi-behavior data.
- **GHCF** [21]: a GCN-based model that jointly embeds user, item, and behavior representations for multi-behavior modeling, which also utilizes non-sampling optimization as in [20] to improve performance.

For simplicity, we conducted experiments only on two single-behavior models (MFBPR and LightGCN) for the multi-behavior task. This approach neglects other behavioral data during the training process. We re-trained single-behavior models for each type of behavior and evaluated the performance based on the corresponding model. For multi-behavior models, note the following: 1) as MC-BPR models only the importance order between different behaviors and does not provide behavior-dependent recommendation, we use the same recommended list for different behavior evaluation; 2) as MBGCN utilizes the target behavior to optimize the loss function, we had to re-train the model for different behaviors by changing the target behavior for different behavior evaluation; 3) as NMTR, EHCF, and GHCF predict for each behavior via their multi-task learning frameworks, there was no need to re-train the model; thus evaluation for different behaviors leveraged the learned behavior embeddings.

5.3. Experimental Settings

5.3.1. Single-behavior recommendation

We followed the settings in [1] to evaluate the performance of the ranking list. The dimension d of the embedding vectors of all the baselines and the proposed model was set to 100. We set the L_2 regularization coefficient λ and learning rate α to 0.001 and 0.025, respectively, and used a grid search over different settings, selecting the hyperparameters that yielded the best performance. To evaluate the performance of the ranking list, we adopted two common metrics for top-N recommendation: recall (Recall@N) and normalized discount cumulative gain (NDCG@N) with $N = 1, 3, 10$ in our experiments. We used a smaller N to test the first-glance recommendation performance. Larger values of N are for "continuous scrolling" recommendations.

5.3.2. Multi-behavior recommendation

For multi-behavior recommendation, we set dimension d of the embedding vectors of all the baselines and the proposed model to 128. Other hyperparameters were the same as the settings in the single-behavior recommendation. We initialized the hyperparameters for the baselines and used a grid search over different settings per the corresponding papers, selecting the hyperparameters that yielded the best performance. For each method, the final reported results were calculated by averaging the results over five repetitions. Following [2] and [34], we treated all items that the user had not interacted with as negative for each user in the test set under the target behavior. Then, we used each method to generate a ranking list for each user with the user's preference scores over all the items, except for the positive ones in the training set of the target behavior. We adopted Recall@N and NDCG@N and set $N = 10, 50, 100$.

5.4. Experimental Results

In this section, we compare the proposed UnifiedIPR framework with several baselines for single-behavior and multi-behavior recommendation, as shown in Tables 3–6. The best results are in boldface; the best-performing method among all the baselines is indicated by "†"; "Improv. (%)" indicates the percentage improvement of the proposed model w.r.t. the best-performing baselines. Below we separately discuss the results for the three prediction tasks.

Table 3: Overall single-behavior recommendation performance

	MovieLens-1M						AMZ I-S					
	Recall			NDCG			Recall			NDCG		
	@1	@3	@10	@1	@3	@10	@1	@3	@10	@1	@3	@10
MF	0.0131	0.0353	0.1036	0.3035	0.2869	0.2717	0.0476	0.0476	0.1104	0.0776	0.0721	0.0861
MFBPR	0.0203	0.0515	0.1280	0.4260	0.3917	0.3453	†0.0501	†**0.1147**	0.1147	†0.0816	0.0752	0.0903
CDAE	0.0215	0.0571	0.1492	0.4296	0.4046	0.3652	0.0413	0.0701	0.1166	0.0674	0.0677	0.0855
LightGCN	0.0216	0.0547	0.1326	0.4548	0.4152	0.3552	0.0494	0.0837	†**0.1371**	0.0803	†0.0803	†**0.1006**
DeepICF	†0.0256	†0.0635	†0.1520	†0.4960	†0.4492	†**0.3903**	0.0366	0.0598	0.0958	0.0593	0.0584	0.0721
UnifiedIPR	**0.0262**	**0.0650**	**0.1550**	**0.5003**	**0.4531**	0.3896	**0.0536**	0.0814	0.1255	**0.0874**	**0.0812**	0.0981
Improv. (%)	2.34%	2.36%	1.97%	0.87%	0.87%	-0.18%	6.99%	-29.03%	-8.46%	7.11%	1.12%	-2.49%

	AMZ PS						AMZ VG					
	Recall			NDCG			Recall			NDCG		
	@1	@3	@10	@1	@3	@10	@1	@3	@10	@1	@3	@10
MF	0.0093	0.0202	0.0449	0.0162	0.0188	0.0285	0.0150	0.0348	0.0799	0.0287	0.0333	0.0507
MFBPR	†**0.0462**	†0.0634	0.0862	†0.0808	†0.0676	†0.0754	†0.0204	0.0426	0.0897	0.0386	0.0416	0.0595
CDAE	0.0095	0.0188	0.0438	0.0171	0.0182	0.0279	0.0151	0.0372	0.0864	†0.0393	0.0348	0.0538
LightGCN	0.0274	0.0484	†0.0891	0.0496	0.0483	0.0639	0.0196	†0.0449	†0.1051	0.0392	†0.0437	†0.0664
DeepICF	0.0083	0.0146	0.0325	0.0152	0.0147	0.0217	0.0081	0.0205	0.0534	0.0165	0.0199	0.0320
UnifiedIPR	0.0452	**0.0692**	**0.1047**	**0.0808**	**0.0721**	**0.0847**	**0.0272**	**0.0569**	**0.1151**	**0.0519**	**0.0559**	**0.0777**
Improv. (%)	-2.16%	9.15%	17.51%	00.00%	6.66%	12.33%	33.33%	26.73%	9.51%	32.06%	27.92%	17.02%

	AMZ DM						AMZ CP-A					
	Recall			NDCG			Recall			NDCG		
	@1	@3	@10	@1	@3	@10	@1	@3	@10	@1	@3	@10
MF	0.0204	0.0416	0.0768	0.0396	0.0419	0.0546	0.0095	0.0207	0.0438	0.0158	0.0186	0.0276
MFBPR	†0.0277	0.0495	0.0869	†0.0507	0.0506	0.0639	†0.0244	0.0360	0.0546	†0.0398	†0.0362	0.0431
CDAE	0.0186	0.0377	0.0783	0.0340	0.0372	0.0526	0.0072	0.0169	0.0384	0.0125	0.0150	0.0234
LightGCN	0.0253	†0.0513	†0.0991	0.0491	†0.0513	†0.0688	0.0189	†0.0382	†0.0766	0.0323	0.0354	†0.0503
DeepICF	0.0188	0.0379	0.0719	0.0349	0.0375	0.0500	OOM	OOM	OOM	OOM	OOM	OOM
UnifiedIPR	**0.0332**	**0.0568**	**0.1001**	**0.0623**	**0.0599**	**0.0748**	**0.0285**	**0.0462**	**0.0787**	**0.0474**	**0.0454**	**0.0578**
Improv. (%)	19.86%	10.72%	1.01%	22.88%	16.76%	8.72%	16.80%	20.94%	2.74%	19.10%	25.41%	14.91%

5.4.1. Single-behavior recommendation

Table 3 shows the results of UnifiedIPR compared to several single-behavior baselines. The findings are listed below and are consistent with previous research [1].

- The proposed UnifiedIPR algorithm yields the best results across most of the datasets. This proves the effectiveness of our framework.
- The improvement percentages of UnifiedIPR are better for positions k=1 compared to positions like k=10, which indicates that interaction-level modeling is remarkably good at first-glance recommendation.

5.4.2. Multi-behavior recommendation

We compared UnifiedIPR with several single-behavior and multi-behavior baselines for recommendation tasks w.r.t. different behaviors, including purchase, cart, and view prediction, as shown in Tables 4–6.

Table 4: Overall purchase recommendation performance

	Beibei						Taobao					
	Recall			NDCG			Recall			NDCG		
	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
MFBPR	0.0355	0.1276	0.2264	0.0182	0.0374	0.0533	0.0342	0.0664	0.0824	0.0204	0.0274	0.0300
LightGCN	0.0444	0.1339	0.1984	0.0212	0.0404	0.0501	0.0438	0.0819	0.1001	0.0258	0.0342	0.0371
MCBPR	0.0488	0.1969	0.3228	0.0226	0.0540	0.0743	0.0713	0.1190	0.1423	0.0383	0.0488	0.0526
NMTR	0.0414	0.2708	0.4534	0.0172	0.0651	0.0947	0.0803	0.1308	0.1666	0.0411	0.0523	0.0581
MBGCN	0.0582	0.3319	0.4823	0.0294	0.1506	0.171	0.1092	0.1854	0.2465	0.0553	0.0788	0.0802
EHCF	0.2424	0.4149	0.5009	0.1365	0.1748	0.1887	0.1175	0.2387	0.3108	0.0667	0.0931	0.1048
GHCF	†**0.2912**	†0.4595	†0.5395	†**0.1569**	†0.1947	†0.2077	†0.1359	†0.2833	†0.3676	†0.0768	†0.1090	†0.1226
UnifiedIPR	0.2387	**0.5596**	**0.6514**	0.1173	**0.1957**	**0.2102**	**0.2187**	**0.5270**	**0.6287**	**0.1087**	**0.1765**	**0.1932**
Improv. (%)	-21.99%	21.78%	20.74%	-33.76%	0.51%	1.20%	60.93%	86.02%	71.03%	41.53%	61.93%	57.59%

	Ecommerce						Rees46					
	Recall			NDCG			Recall			NDCG		
	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
MFBPR	0.0571	0.1269	0.1737	0.0320	0.0471	0.0547	0.0876	0.2201	0.2972	0.0473	0.0759	0.0883
LightGCN	0.0651	0.1853	0.2608	0.0332	0.0538	0.0660	0.1656	0.1864	0.3925	0.0914	0.1143	0.1371
MCBPR	0.0792	0.2070	0.2919	0.0407	0.0683	0.0820	0.1743	0.3087	0.3598	0.0837	0.1139	0.1222
NMTR	0.0691	0.2359	0.3543	0.0355	0.0630	0.0821	0.1724	0.3160	0.3759	0.0825	0.1141	0.1238
MBGCN	0.0657	0.1942	0.2790	0.0345	0.0591	0.0729	0.1783	0.3469	0.4199	0.0990	0.1365	0.1483
EHCF	0.1659	0.3881	0.5123	0.0907	0.1390	0.1592	0.3563	0.5624	0.6223	0.2161	0.2623	0.2720
GHCF	†0.2330	†0.4351	†0.5347	†0.1375	†0.1819	†0.1981	†0.3945	†0.5749	†0.6255	†0.2410	†0.2818	†0.2901
UnifiedIPR	**0.2621**	**0.5532**	**0.6861**	**0.1447**	**0.2086**	**0.2302**	**0.4118**	**0.6070**	**0.6620**	**0.2569**	**0.3010**	**0.3099**
Improv. (%)	12.49%	27.14%	28.31%	5.24%	14.68%	16.20%	4.39%	5.58%	5.84%	6.60%	6.81%	6.83%

Purchase recommendation

This is the typical recommendation task evaluated in most studies on multi-behavior recommender systems. Table 4 tabulates the results on the four datasets; below are the findings.

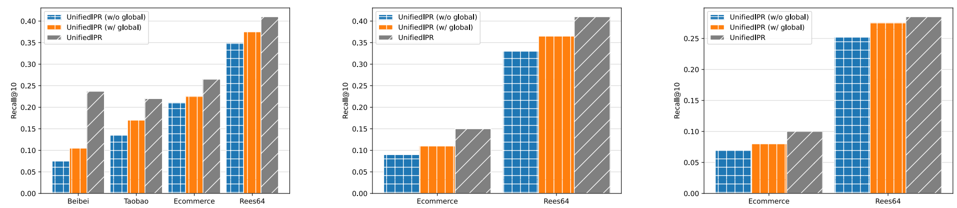
- Our algorithm consistently outperforms all baselines over the four datasets except for Recall@10 and NDCG@10 on the Beibei dataset, which justifies the effectiveness of our model.
- All multi-behavior models outperform the two single-behavior models, which attests the effectiveness of leveraging multiple types of behavioral data. This is consistent with previous findings [21, 20].
- GHCF is the strongest baseline of the compared methods. Nevertheless, the proposed multi-behavior algorithm still yields significant improvements on all four datasets (e.g., ranging from 5.58% to 86.02% improvement in terms of Recall@50).

Table 5: Overall cart recommendation performance

	Ecommerce						Rees46					
	Recall			NDCG			Recall			NDCG		
	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
MFBPR	0.0285	0.0697	0.1004	0.0153	0.0242	0.0291	0.1058	0.2587	0.3291	0.0577	0.0913	0.1027
LightGCN	0.0606	0.1245	0.2261	0.0324	0.0485	0.0644	0.2027	0.3453	0.4167	0.1226	0.1540	0.1656
MCBPR	0.0644	0.1702	0.2457	0.0407	0.0683	0.0820	0.1431	0.2631	0.3148	0.0685	0.0951	0.1035
NMTR	0.0744	0.1498	0.1951	0.0393	0.0558	0.0632	0.1718	0.3143	0.3736	0.0823	0.1137	0.1233
MBGCN	0.0708	0.1369	0.1894	0.0355	0.0521	0.0620	0.2257	0.4001	0.4846	0.1363	0.2029	0.2253
EHCF	0.0247	0.0743	0.1126	0.0121	0.0227	0.0289	0.3217	0.5339	0.5945	0.1896	0.2370	0.2469
GHCF	†0.0889	†0.2129	†0.2938	†0.0479	†0.0747	†0.0878	†0.3616	†0.5459	†0.5977	†0.2092	†0.2510	†0.2594
UnifiedIPR	**0.1494**	**0.3592**	**0.4780**	**0.0803**	**0.1259**	**0.1451**	**0.4089**	**0.6004**	**0.6565**	**0.2516**	**0.2948**	**0.3039**
Improv. (%)	68.05%	68.72%	62.70%	67.64%	68.54%	65.26%	13.08%	9.98%	9.84%	20.27%	17.45%	17.15%

Table 6: Overall view recommendation performance

	Ecommerce						Rees46					
	Recall			NDCG			Recall			NDCG		
	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
MFBPR	0.0257	0.0630	0.0890	0.0152	0.0232	0.0274	0.1964	0.3775	0.4647	0.1094	0.1491	0.1632
LightGCN	0.0560	0.1432	0.2025	0.0230	0.0487	0.0583	0.2294	0.3845	0.4510	0.1365	0.1708	0.1816
MCBPR	0.0433	0.1141	0.1649	0.0225	0.0378	0.0460	0.1449	0.2707	0.3234	0.0849	0.1125	0.1211
NMTR	0.0368	0.1161	0.1881	0.0189	0.0357	0.0473	0.1625	0.3007	0.3575	0.0779	0.1085	0.1176
MBGCN	0.0425	0.1298	0.1930	0.0296	0.0438	0.0531	0.1960	0.3953	†0.4804	0.1035	0.1472	0.1610
EHCF	0.0314	0.0975	0.1482	0.0155	0.0296	0.0378	†0.2137	†0.4101	0.4773	0.1132	†0.1567	†0.1676
GHCF	†0.0766	†0.1796	†0.2465	†0.0411	†0.0634	†0.0742	0.2120	0.3939	0.4553	†0.1137	0.1543	0.1643
UnifiedIPR	**0.1015**	**0.2481**	**0.3337**	**0.0547**	**0.0866**	**0.1005**	**0.2863**	**0.4722**	**0.5366**	**0.1806**	**0.2220**	**0.2325**
Improv. (%)	32.51%	38.14%	35.38%	33.09%	36.59%	35.44%	33.97%	12.14%	11.70%	58.31%	41.67%	38.72%



(a) Purchase (b) Cart (c) View

Figure 2: Ablation studies on global behavior embeddings

Cart and view recommendation

To verify the effectiveness of our model for recommendation regarding different behaviors, we further evaluated cart and view recommendations, tasks that are often overlooked and thus not evaluated in most literature. We report the results conducted on the Ecommerce and Rees46 datasets, as shown in Tables 5 and 6, from which we itemize the following findings.

- The proposed multi-behavior algorithm significantly outperforms all baselines over the two datasets in terms of all metrics, demonstrating its ability to predict not only the "target behavior" but also other behaviors compared to state-of-the-art methods.
- GHCF remains the strongest baseline, indicating the superiority of this state-of-the-art GNN-based model for multi-behavior recommendation.

Overall, the multi-behavior algorithm of UnifiedIPR shows the effectiveness of leveraging interaction as training units and incorporating fine-grained behavior embeddings of users and items to learn a unified embedding matrix for multi-behavior recommendation. These results demonstrate notable performance improvements over all three prediction tasks compared to state-of-the-art multi-behavior recommendation approaches.

6. Discussion

In this section, we study the effectiveness of the global behavior embeddings, as well as the sensitivity of hyperparameters λ_Θ and α. We discuss only the impact of multi-behavior recommendation, as single-behavior recommendation has already been studied in the literature [1].

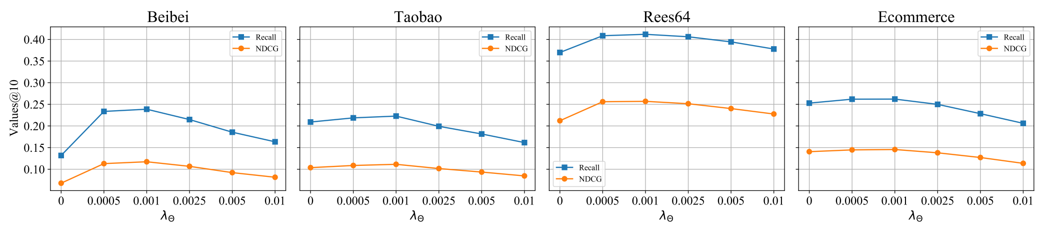


Figure 3: Sensitivity analysis of λ_Θ

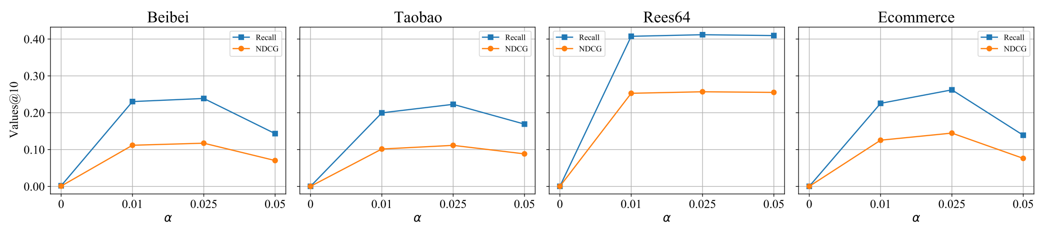


Figure 4: Sensitivity analysis of α

6.1. Ablation Studies on Global Behavior Embeddings

To understand the impact of our mechanism for global behavior embeddings, we additionally considered two variants of the proposed model: UnifiedIPR (w/o global) and UnifiedIPR (w/ global), which disables and enables the global behavior r_g in $G(\mathcal{V}, \mathcal{E}, \psi)$, respectively (see Section 4.5). Moreover, both variants apply $\hat{y}^r_{ui} = (\theta_u + \theta^r_u) \cdot (\theta_i + \theta^r_i)$ as the scoring function to estimate the likelihood that user u interacts with item i under behavior r. (Note that the above score function is different from that used in the original UnifiedIPR in Section 4.6, which additionally concatenates the global behavior embeddings when calculating the scores.)

As shown in Figure 2, adding global behavioral information indeed yields better performance for all types of recommendation tasks (see the bars representing the results of UnifiedIPR (w/ global) and UnifiedIPR). Moreover, UnifiedIPR is shown to consistently outperform UnifiedIPR (w/ global), which demonstrates that including global behavior embeddings in the scoring function further benefits performance and thus yields superior results.

6.2. Sensitivity Analysis on Hyperparameters

Due to space limitations, we only report the hyperparameter analysis on the *purchase* prediction results in Figures 3 and 4; predictions on other behaviors exhibit similar phenomena. For the L_2 regularization parameter λ_Θ, all datasets perform relatively poorly when $\lambda_\Theta = 0$ and show the best performance when $\lambda_\Theta = 0.001$. For the learning rate parameter α, $\alpha = 0.025$ yields the best results for all datasets.

6.3. Computational Efficiency Comparison

We used E-commerce, the largest dataset, to compare the computational efficiency among different models. Figure 5 plots the training time of the proposed model and the seven compared methods.[2] As shown in the figure, the proposed UnifiedIPR requires approximately 600 seconds to complete the training process, which is much faster than all models except BPR. Note also that deep learning-based models NMTR and EHCF require more training time to achieve satisfactory performance than GNN-based models MBGCN and GHCF. Moreover, while BPR, MCBPR, and UnifiedIPR use only CPU computations, other models use GPU computations. Such results demonstrate the lightweight nature and computational efficiency of the proposed embedding learning framework, which is thus more practical than other advanced methods.

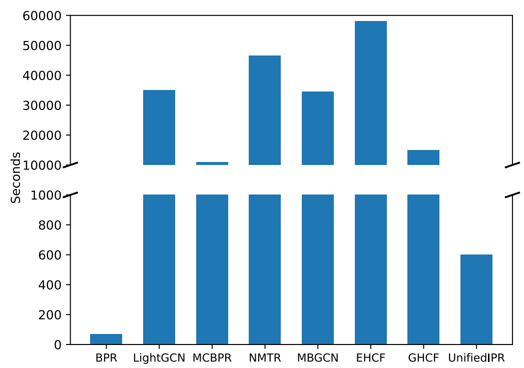


Figure 5: Training time

7. Conclusion

We propose unified interaction-level preference ranking (UnifiedIPR), a unified interaction-based pairwise ranking embedding framework for single-behavior and multi-behavior embedding learning. UnifiedIPR samples and constructs interaction triples, leveraging a pairwise ranking algorithm to capture user and item preferences under each behavior based on interaction similarity. Its effectiveness and efficiency are demonstrated through extensive experiments and analysis. Notably, this framework is highly flexible and can be easily adapted to other recommendation tasks with minimal modification. We plan to investigate the performance of more recommendation tasks, such as incorporating metadata from users and items.

[2] Values reported in the figure vary when different implementations are applied.

References

[1] S.-Y. Liu, H. H. Chen, C.-M. Chen, M.-F. Tsai, C.-J. Wang, IPR: Interaction-level Preference Ranking for Explicit Feedback, in: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2022, pp. 1912–1916.

[2] S. Rendle, C. Freudenthaler, Z. Gantner, L. Schmidt-Thieme, BPR: Bayesian Personalized Ranking from Implicit Feedback, arXiv preprint arXiv:1205.2618 (2012).

[3] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, Item-based Collaborative Filtering Recommendation Algorithms, in: Proceedings of the 10th International Conference on World Wide Web, 2001, pp. 285–295.

[4] Y. Hu, Y. Koren, C. Volinsky, Collaborative Filtering for Implicit Feedback Datasets, in: Proceedings of the 8th IEEE International Conference on Data Mining, 2008, pp. 263–272.

[5] S. Kabbur, X. Ning, G. Karypis, FISM: Factored Item Similarity Models for Top-n Recommender Systems, in: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2013, pp. 659–667.

[6] S. Rendle, Factorization Machines, in: Proceedings of the 10th IEEE International Conference on Data Mining, 2010, pp. 995–1000.

[7] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, J. Leskovec, Graph Convolutional Neural Networks for Web-scale Recommender Systems, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2018, pp. 974–983.

[8] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, M. Wang, LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2020, pp. 639–648.

[9] X. Wang, H. Jin, A. Zhang, X. He, T. Xu, T.-S. Chua, Disentangled Graph Collaborative Filtering, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2020, pp. 1001–1010.

[10] T. Huang, Y. Dong, M. Ding, Z. Yang, W. Feng, X. Wang, J. Tang, MixGCF: An Improved Training Method for Graph Neural Network-based Recommender Systems, in: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2021, pp. 665–674.

[11] F. Liu, Z. Cheng, L. Zhu, Z. Gao, L. Nie, Interest-aware Message-passing GCN for Recommendation, in: Proceedings of the 30th International Conference on World Wide Web, 2021, pp. 1296–1305.

[12] T. Badriyah, E. T. Wijayanto, I. Syarif, P. Kristalina, A Hybrid Recommendation System for E-commerce Based on Product Description and User Profile, in: Proceedings of the 7th International Conference on Innovative Computing Technology, 2017, pp. 95–100.

[13] Y. Deldjoo, M. Elahi, P. Cremonesi, F. Garzotto, P. Piazzolla, M. Quadrana, Content-based Video Recommendation System Based on Stylistic Visual Features, Journal on Data Semantics 5 (2016) 99–113.

[14] X. Jin, Y. Zhou, B. Mobasher, A Maximum Entropy Web Recommendation System: Combining Collaborative and Content Features, in: Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2005, pp. 612–617.

[15] L. Xia, C. Huang, Y. Xu, P. Dai, B. Zhang, L. Bo, Multiplex Behavioral Relation Learning for Recommendation via Memory Augmented Transformer Network, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2020, pp. 2397–2406.

[16] L. Xia, C. Huang, Y. Xu, P. Dai, X. Zhang, H. Yang, J. Pei, L. Bo, Knowledge-enhanced Hierarchical Graph Transformer Network for Multi-behavior Recommendation, in: Proceedings of the 35th AAAI Conference on Artificial Intelligence, 2021, pp. 4486–4493.

[17] B. Jin, C. Gao, X. He, D. Jin, Y. Li, Multi-behavior Recommendation with Graph Convolutional Networks, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2020, pp. 659–668.

[18] B. Loni, R. Pagano, M. Larson, A. Hanjalic, Bayesian Personalized Ranking with Multi-channel User Feedback, in: Proceedings of the 10th ACM Conference on Recommender Systems, 2016, pp. 361–364.

[19] H. Qiu, Y. Liu, G. Guo, Z. Sun, J. Zhang, H. T. Nguyen, BPRH: Bayesian Personalized Ranking for Heterogeneous Implicit Feedback, in: Information Sciences, Vol. 453, 2018, pp. 80–98.

[20] C. Chen, M. Zhang, Y. Zhang, W. Ma, Y. Liu, S. Ma, Efficient Heterogeneous Collaborative Filtering without Negative Sampling for Recommendation, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 19–26.

[21] C. Chen, W. Ma, M. Zhang, Z. Wang, X. He, C. Wang, Y. Liu, S. Ma, Graph Heterogeneous Multi-relational Recommendation, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 3958–3966.

[22] X. Zhou, D. Liu, J. Lian, X. Xie, Collaborative Metric Learning with Memory Network for Multi-relational Recommender Systems, in: Proceedings of the 28th International Joint Conference on Artificial Intelligence, 2019, pp. 4454–4460.

[23] C.-M. Chen, C.-J. Wang, M.-F. Tsai, Y.-H. Yang, Collaborative Similarity Embedding for Recommender Systems, in: Proceedings of the 28th International Conference on World Wide Web, 2019, pp. 2637–2643.

[24] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, T.-S. Chua, Neural Collaborative Filtering, in: Proceedings of the 26th International Conference on World Wide Web, 2017, pp. 173–182.

[25] X. Wang, X. He, M. Wang, F. Feng, T.-S. Chua, Neural Graph Collaborative Filtering, in: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019, pp. 165–174.

[26] J.-H. Yang, C.-M. Chen, C.-J. Wang, M.-F. Tsai, HOP-rec: High-order Proximity for Implicit Recommendation, in: Proceedings of the 12th ACM Conference on Recommender Systems, 2018, pp. 140–144.

[27] Y.-N. Chuang, C.-M. Chen, C.-J. Wang, M.-F. Tsai, Y. Fang, E.-P. Lim, TPR: Text-aware Preference Ranking for Recommender Systems, in: Proceedings of the 29th ACM International Conference on Information and Knowledge Management, 2020, pp. 215–224.

[28] B. Recht, C. Re, S. Wright, F. Niu, Hogwild!: A Lock-free Approach to Parallelizing Stochastic Gradient Descent, in: Proceedings of the 24th International Conference on Neural Information Processing Systems, 2011, pp. 693–701.

[29] J. Ni, J. Li, J. McAuley, Justifying Recommendations Using Distantly-labeled Reviews and Fine-grained Aspects, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, 2019, pp. 188–197.

[30] Y. Koren, R. Bell, C. Volinsky, Matrix Factorization Techniques for Recommender Systems, Computer 42 (8) (2009) 30–37.

[31] Y. Wu, C. DuBois, A. X. Zheng, M. Ester, Collaborative Denoising Auto-encoders for Top-n Recommender Systems, in: Proceedings of the 9th ACM International Conference on Web Search and Data Mining, 2016, pp. 153–162.

[32] F. Xue, X. He, X. Wang, J. Xu, K. Liu, R. Hong, Deep Item-based Collaborative Filtering for Top-n Recommendation, ACM Transactions on Information Systems 37 (3) (2019) 1–25.

[33] C. Gao, X. He, D. Gan, X. Chen, F. Feng, Y. Li, T.-S. Chua, D. Jin, Neural Multi-task Recommendation from Multi-behavior Data, in: Proceedings of the 35th IEEE International Conference on Data Engineering, 2019, pp. 1554–1557.

[34] W. Krichene, S. Rendle, On Sampled Metrics for Item Recommendation, in: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2020, pp. 1748–1757.