arXiv:2511.08378v4 [cs.IR] 17 Jun 2026

Bid Farewell to Seesaw: Towards Accurate Long-tail Session-based Recommendation via Dual Constraints of Hybrid Intents

Xiao Wang, Ke Qin, Dongyang Zhang, Xiurui Xie, Shuang Liang*

University of Electronic Science and Technology of China
wangxiao16@std.uestc.edu.cn, qinke@uestc.edu.cn, dyzhang@uestc.edu.cn, xiexiurui@uestc.edu.cn, shuangliang@uestc.edu.cn

Abstract

Session-based recommendation (SBR) aims to predict anonymous users' next interaction based on their interaction sessions. In the practical recommendation scenario, low-exposure items constitute the majority of interactions, creating a long-tail distribution that severely compromises recommendation diversity. Existing approaches attempt to address this issue by promoting tail items but incur accuracy degradation, exhibiting a "see-saw" effect between long-tail and accuracy performance. We attribute such conflict to session-irrelevant noise within the tail items, which existing long-tail approaches fail to identify and constrain effectively. To resolve this fundamental conflict, we propose **HID** (**H**ybrid **I**ntent-based **D**ual Constraint Framework), a plug-and-play framework that transforms the conventional "see-saw" into "win-win" through introducing the hybrid intent-based dual constraints for both long-tail and accuracy. Two key innovations are incorporated in this framework: (i) *Hybrid Intent Learning*, where we reformulate the intent extraction strategies by employing attribute-aware spectral clustering to reconstruct the item-to-intent mapping. Furthermore, discrimination of session-irrelevant noise is achieved through the assignment of the target and noise intents to each session. (ii) *Intent Constraint Loss*, which incorporates two novel constraint paradigms regarding the *diversity* and *accuracy* to regulate the representation learning process of both items and sessions. These two objectives are unified into a single training loss through rigorous theoretical derivation. Extensive experiments across multiple SBR models and datasets demonstrate that HID can enhance both long-tail performance and recommendation accuracy, establishing new state-of-the-art performance in long-tail recommender systems. The source code is available at "https://github.com/jarviswww/Code4HID".

Introduction

Session-based recommendation (SBR) addresses information overload by predicting the next item from short-term interactions, particularly in privacy-sensitive scenarios lacking long-term user profiles (Li et al. 2025; Latifi, Mauro, and Jannach 2021). While deep learning methods in SBR (e.g., deep sequential models (Hidasi et al. 2016; Li et al. 2017; Liu et al. 2018; Yuan et al. 2021; Hou et al. 2022) and deep graphic models (Wu et al. 2019; Qiu et al. 2019; Wang et al. 2020; Xia et al. 2021b,a; Pan et al. 2020)) can effectively model item correlations, their model-centric focus overlooks inherent data biases. A key challenge is the long-tail distribution in recommendation data (Sundaresan 2011; Yang et al. 2023; Liu and Zheng 2020a), where a small number of high-exposure items (i.e., head items) dominate the model's attention, while a significantly larger number of low-exposure items (i.e., tail items) are often disregarded. This unfair phenomenon leads to the overlooking of potentially essential but low-exposure tail items, limiting the diversity of recommendations (Turgut et al. 2023; Yin et al. 2024; Lee, Kim, and Shin 2024). Besides, the long-tail distribution causes the model to be more inclined to recommend head items, resulting in a vicious cycle.

Previous advancements in long-tail SBR focus on devel-

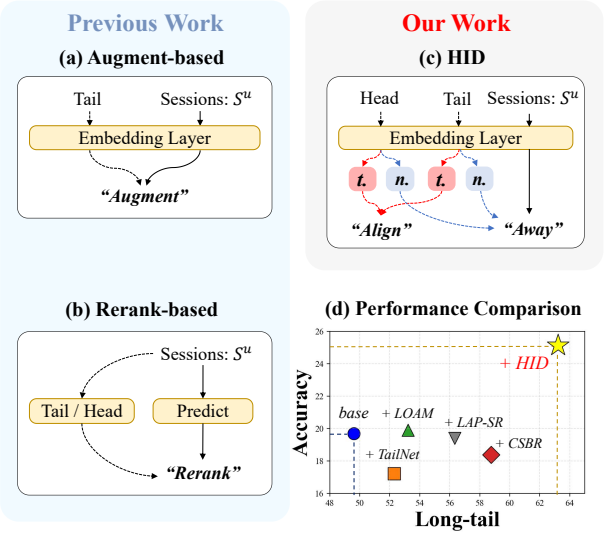


Figure 1: Comparison between the our proposed HID and previous work. (a) illustrates the design of HID, where ***t.*** and ***n.*** denotes target and noise items for session S^u, respectively; (c) and (d) demonstrate the frameworks of previous long-tail approaches. (b) evaluates the accuracy (i.e., HR@20) and long-tail performance (i.e., tCov@20) of the base SBR model GRU4Rec (Hidasi et al. 2016) and GRU4Rec + long-tail approaches on Tmall dataset.

*Corresponding author.

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

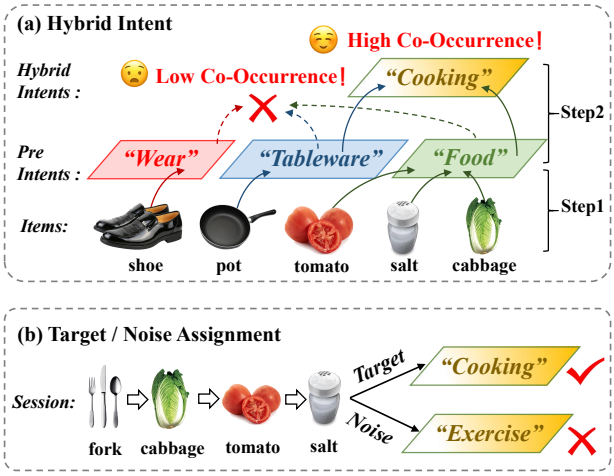


Figure 2: The demonstration of: (a) Hybrid Intent: Step 1 groups items by shared attributes (e.g., food) as the *preliminary intents*; Step 2 combines attributes with high co-occurrence (e.g., food + pot) to form the *hybrid intents* (e.g., cooking). (b) Intent Assignment: Assigns target (relevant) and noise (irrelevant) hybrid intents to anonymous sessions.

oping plugins that seamlessly integrate with existing SBR models, emphasizing the significance of tail items to mitigate the long-tail issue (Liu and Zheng 2020a; Chen et al. 2023; Yang et al. 2023; Peng and Zhou 2024). Broadly, long-tail SBR approaches fall into two categories: (i) Augment-based approaches, which employ augmentation strategies to refine the tail item embeddings or session embeddings (Yang et al. 2023; Kim et al. 2023; Huang et al. 2024; Liu et al. 2024), and (ii) Rerank-based approaches, which predict head/tail item distributions based on interaction sessions and directly modify the final ranking results (Liu and Zheng 2020a; Chen et al. 2023; Peng and Zhou 2024). Both approaches consistently emphasize the significance of tail items. The brief demonstrations of their frameworks are given in (a) and (b) of Figure 1. Despite their success, two critical limitations remain unresolved: (i) their undifferentiated emphasis on tail items introduces session-irrelevant noise (e.g., "clothing" for a session consists of books.), as ***not all tail items align with session-specific user requirements***, resulting in the degradation of recommendation accuracy, and (ii) they lack explicit supervisory signals for long-tail objectives, thus relying on indirect optimization via cross-entropy loss. Crucially, such augmentation and reranking strategies often conflict with the cross-entropy optimization objective due to the inclusion of potential session-irrelevant items, resulting in a "see-saw" effect (Wang et al. 2021; Wei et al. 2024). To address these flaws, our work revolves around two key innovations: (i) the effective discrimination of noise, restricting the consideration of long-tail issues to session-relevant tail items, and (ii) the introduction of explicit long-tail supervisory signals to concurrently improve the long-tail and accuracy performance.

For the noise discrimination, given that interaction sessions are driven by user intent (Li et al. 2023; Wang et al. 2024), we employ intent modeling to capture the overarching preference of the anonymous user. Previous work primarily derives user intents from restricted sequential segments (e.g., sliding windows) or semantically clustered items within individual sessions (Wang et al. 2019; Zhang et al. 2023; Choi et al. 2024a; Wang et al. 2024), but suffer from unreliable intent extraction due to noise interference and neglect cross-session intent consistency (Choi et al. 2024b; Wang et al. 2024). Therefore, we propose the *hybrid intent*, which captures the user preference through attribute consistency (e.g., commodity categories, music genres) and item co-occurrence patterns, as shown in (a) of Figure 2. Following this, we assign target and noise intents to each session to enable the discrimination of session-irrelevant items, as shown in (b) of Figure 2.

For the long-tail supervisory signals, since the long-tail issue stems from the disparity in embedding distributions between head and tail items, resulting in discrepancies in their similarity to user embeddings (Yin et al. 2012; Gupta et al. 2019), we propose explicit constrains on these similarities during the training process to provide direct supervised signal. Specifically, to address the distribution inconsistency between head and tail items, we align their similarity scores to each session through a novel constraint objective. This constraint is termed the *Constraint for Long-tail*, which operates exclusively on session-relevant items (i.e., items belong to the target intents). Furthermore, to ensure the recommendation accuracy, we introduce an additional *Constraint for Accuracy* that explicitly enlarges the similarity discrepancy between sessions and session-irrelevant items (i.e., items belong to noise intents) during the training process. The mutual independence of target and noise intents ensures that the two constraints are not conflicting. The brief framework of constraints is given in Figure 1 (c).

Incorporating the above innovations, we name this novel approach as the **H**ybrid **I**ntent-based **D**ual Constraint Framework (**HID**). This model-agnostic and plug-and-play framework can be easily integrated into existing SBR models. Specifically, HID consists of the *hybird intent learning* module and the *intent constraint loss* (ICLoss). The *hybird intent learning* module first aggregates items that share the same attribute to form preliminary intent units. Subsequently, based on the attribute co-occurrence relations from all interaction sessions, a *preliminary intent graph* is constructed whose nodes are the preliminary intents and edge weights represent their co-occurrence frequency. After that, we employ spectral clustering, grouping the preliminary intents into hybrid intents. Furthermore, we derive the theoretical formulations of the *Constraint for Long-tail* and the *Constraint for Accuracy*, and combine them to acquire the *intent constraint loss*, which aligns embeddings of head and tail items within the target intent while repelling noise intents from the current session in the feature space. As shown in (d) of Figure 1, HID achieves significant improvements in both accuracy and diversity over previous long-tail competitors, due to its session-irrelevant noise discrimination capability and dual constraints of long-tail and accuracy.

To sum up, we conclude the main contributions of this work as follows:

- We propose a novel framework named HID aimed at achieving accurate long-tail SBR. Its brevity ensures easy reproduction and integration with existing SBR models.
- We innovatively propose a novel concept of the hybrid intent, which advances session-based recommendation by jointly modeling attribute-level correlations and attribute co-occurrence patterns to redefine the item-intent mapping.
- We explicitly model the learning objective of accurate long-tail SBR through two novel constraint paradigms for both the long-tail and accuracy, and integrate them into a unified, theoretically-grounded intent constraint loss that optimizes both objectives.
- Extensive experiments conducted on various SBR models and long-tail competitors demonstrate the effectiveness of HID in addressing the long-tail issue and improving recommendation accuracy.

Related Works

Augment-based Approaches. This technical route primarily focuses on enhancing the embeddings of tail items or emphsizing the significance of tail items when generating the session embeddings. LOAM (Yang et al. 2023) enhances tail items and sessions through the Niche-Walk Augmentation and Tail Session Mixup. GALORE (Luo et al. 2023) introduces a graph augmentation approach to enhance the edge of tail items in the interaction graph. GUME (Lin et al. 2024) employs the graphs and user modalities enhancement. MelT (Kim et al. 2023) employs mutual enhancement of tail users and items, which jointly mitigates the long-tail issue. Additionally, some approaches have explored the useage of large language models (LLMs). LLM-ESR (Liu et al. 2024) utilizes the semantic embeddings derived from LLMs to enhance the tail items.

Rerank-based Approaches. This technical route aims to infer the distribution of tail and head items from interaction sessions, thereby enabling direct adjustment of recommendation results. TailNet (Liu and Zheng 2020a) introduces a preference mechanism to predict the adjustment index of head and tail items. CSBR (Chen et al. 2023) proposes two additional training objectives: distribution prediction and distribution alignment to calibrate the recommendation results. LAP-SR (Peng and Zhou 2024) adjusts the weight scores of recommended items based on the long-tail items and the intra-session similarity.

Although the above methods have made contributions to addressing the long-tail problem, they all neglect the consideration of noise in tail items and lack explicit modeling of the long-tail objective.

Preliminaries

Problem Definition

Let $V = \{v_1, v_2, \ldots, v_m\}$ represent the set of all unique items, where m is their total counts. An anonymous session is represented as $S^u = \{v_1^u, v_2^u, \ldots, v_l^u\}$, where u is the session ID, l is the length of the interaction session, and $v_t^u \in V$ (0 ¡ t ¡ l) is the item ID which is interacted at timestep t. *In this paper, all symbols in bold represent the vector embeddings.* For example, in $S^u = \{\mathbf{v}_1^u, \mathbf{v}_2^u, \ldots, \mathbf{v}_l^u\}$, $\mathbf{v}_t^u \in \mathbb{R}^d$ represents the vector embedding of item v_t^u. Given a session S^u, the task in session-based recommendation is to predict the next-interacted item v_{l+1}^u (i.e., the ground truth item). According to the Pareto principle (Box and Meyer 1986), the top 20% of items with the highest frequency of occurrence are considered to be head items, while the remainings are tail items.

Session-based Recommendation Models

Session-based recommendation (SBR) models follow a two-stage paradigm: a SBR encoder to transform the inputs into session embeddings, and a prediction layer to generate the recommendations. The basic structure of the SBR model is demonstrated in the blue components of Figure 3.

Given a session $S^u = \{\mathbf{v}_1^u, \mathbf{v}_2^u, \ldots, \mathbf{v}_l^u\}$, whose vector embeddings are initialized using the Gaussian distribution, SBR models typically propagate it into a SBR encoder, which is denoted as $F(x)$ in Figure 3, to generate the session embedding: $\mathbf{S}^u = F(S^u)$, where $S^u \in \mathbb{R}^{l \times d}$, $\mathbf{S}^u \in \mathbb{R}^d$.

After acquiring session embedding $\mathbf{S}^u$, SBR models multiply it with the candidate item embeddings and apply a softmax to calculate the probabilities of each item being the next-interacted one: $y_i' = \text{softmax}(\mathbf{S}^{u^T}\mathbf{v}_i)$, where $\mathbf{v}_i$ is the embedding of item $v_i \in V$. Then, the next-item prediction task is adopted as the learning objective, where the cross-entropy loss is usually leveraged as the objective function: $\mathcal{L}_p = -\sum_{i=1}^{m} y_i \log(y_i')$. where y_i is the one-hot encoding vector of the ground truth.

Proposed Method

Hybrid Intent Learning

Existing intent mining approaches exhibit two weaknesses: (i) only temporal relations among items are considered, which is not always reliable due to the interaction noise, and (ii) only a single session is considered, neglecting that items from different sessions can reflect the same intent. Therefore, we propose attribute-aware spectral clustering, giving the brief demonstration in right part of Figure 3.

Note that the whole process of acquiring hybrid intents can be pre-computed and stored locally. Therefore, during training or serving, only providing the item for retrieval enables the acquisition of hybrid intents.

Preliminary Intent. Since items sharing the same attribute can typically reflect similar user preferences (e.g., electronic products or books), we consider the item attribute as the preliminary intent unit. Given item attribute set $C' = \{c_1', c_2', ..., c_k'\}$, where c_i' (1¡i¡k) is the i-th attributes that represents a specific preliminary intent, and k is their total counts. For each c_i', we denote it as a set of items $c_i' = \{v_{c_i,1}, v_{c_i,2}, ..., v_{c_i,|c_i'|}\}$.

Preliminary Intent Graph. To explore the attributes relations within all sessions, we first replace the item IDs within each session with their corresponding attribute IDs. After that, we iterate over all attributes within each session and count the 1-hop neighbors of each attribute, along with the

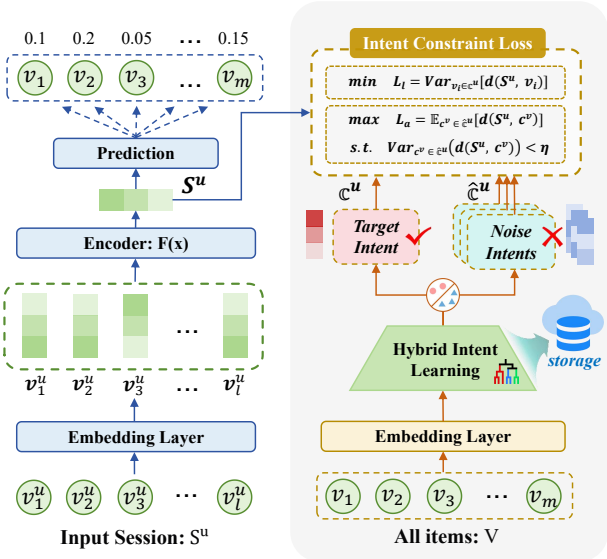


Figure 3: The overall architecture of SBR *model* (left) + HID (right). The Hybrid Intent Learning module first assigns items to k preliminary intents, and then further divides them into n hybrid intents C based on the topological relationships in the preliminary intent graph. After refining the hybrid intents, the intent constraint loss is introduced to regulate the learning process of session embedding $\mathbf{S}^u$.

frequency of their occurrences to form the preliminary intent graph. This intent graph is denoted as $\mathcal{G} = (\mathcal{P}, \mathcal{E}, \mathcal{W})$, where $\mathcal{P}$ is the set of attribute IDs, $\mathcal{E} = \{(c'_i, c'_j) \mid c'_i \in C', c'_j \in \mathcal{N}_{c'_i}\}$ is the edge between attribute c'_i and c'_j, where $\mathcal{N}_{c'_i}$ is the neighbor set of attribute c'_i, and $\mathcal{W}$ is the set of weights, where $w_{ij} \in \mathcal{W}$ of the edge (c'_i, c'_j) is the co-occurrence frequency of attribute c'_i and c'_j.

Hybrid Intent. After acquiring the preliminary intent graph $\mathcal{G}$, with the aim of mining the global co-occurance patterns of attributes, the spectral clustering is employed to learn the topological relations among attributes. Given the graph $\mathcal{G} = (\mathcal{P}, \mathcal{E}, \mathcal{W})$, we first calculate its Laplace matrix:

$$L = I - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}, \tag{1}$$

where $D_{ii} = \sum_j w_{ij}$. Then, we compute the eigenvalues and eigenvectors of the normalized Laplacian matrix L. Let $\lambda_1 \leq \lambda_2 \leq ... \leq \lambda_q$ be the smallest q eigenvalues and their corresponding eigenvectors $\hat{\lambda}_1, \hat{\lambda}_2, ..., \hat{\lambda}_q$ form the eigenvector matrix. Each row of eigenvector matrix represents the embedding of a node in the reduced q-dimensional space.

After that, we apply the k-means algorithm on the rows of the eigenvector matrix. The i-th row of the eigenvector matrix corresponds to the i-th attribute of C', which also corresponds to a node in the preliminary intent graph. Therefore, the attributes are reclassified into n clusters. Since the attributes reprsent the preliminary intents, we combine attributes belonging to the same cluster to form the hybrid intent. The hybrid intent set is defined as $C = \{c_1, c_2, ..., c_n\}$. The embedding of the hybrid intent is derived from the item embeddings associated with the attributes it contains. To reduce time complexity, we concatenate the items within attributes and then apply average pooling to obtain the hybrid intent embedding, which can be formulated as follows:

$$\mathbf{c}_i = \frac{1}{|c_i|} \sum_{v_j \in c_i} \mathbf{v}_j. \tag{2}$$

where $c_i = \{v_{c_i,1}, v_{c_i,2}, ..., v_{c_i,|c_i|}\}$, $v_{c_i,1}$ to $v_{c_i,|c_i|}$ are the items from the attributes that form the hybrid intents c_i, and $1 < i < n$.

Target and Noise Intents. After acquiring the set of hybrid intents, for each batch of sessions $\mathcal{B} = \{S^1, S^2, ..., S^b\}$, we define the *target intent* and *noise intents* for session $S^u = \{v^u_1, v^u_2, ..., v^u_l\}$ where $1 < u < b$ as follows:

Definition 1 (Target Intent). *For session S^u, the hybrid intents that contain its next-item v^u_{l+1} are considered as its target intent set $\mathcal{C}^u$.*

$$\mathcal{C}^u = \{c_i \mid v^u_{l+1} \in c_i, c_i \in C\} \tag{3}$$

Definition 2 (Noise Intent). *For session S^u, given the mini-batch $\mathcal{B}$, target intents of other sessions $S^v \in \mathcal{B} \setminus S^u$ that are not within $\mathcal{C}^u$ are considered as its noise intent set $\hat{\mathcal{C}}^u$.*

$$\hat{\mathcal{C}}^u = \{c_i \mid v^v_{l+1} \in c_i, S^v \in \mathcal{B} \setminus S^u, c_i \in C \setminus \mathcal{C}^u\} \tag{4}$$

Both the target and noise intents are only leveraged in the intent constraint loss as supervisory signals during the training stage, so there is no risk of data leakage.

Dual Constraints for Long-tail and Accuracy

Following the extraction of hybrid intent embeddings $\mathbf{C} = \{\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_n\}$, the subsequent objective involves imposing constraints on the learning process of session embeddings. Given the session embedding $\mathbf{S}^u$ learned by traditional SBR models such as STAMP (Liu et al. 2018) or SRGNN (Wu et al. 2019), our next aim is to construct supervisory signals regarding the long-tail performance and recommendation accuracy. The demonstration of these supervisory signals are given in Figure 3.

The sequential nonlinear transformations in $F(x)$ introduces potential misalignment between the scale of hybrid intent embeddings and derived session embeddings. To mitigate this discrepancy and enforce commensurable embedding spaces, we employ L_2-norm to project both embedding sets onto a unit hypersphere, thereby establishing a unified metric space for subsequent operations:

$$\mathbf{c}_i = \frac{\mathbf{c}_i}{||\mathbf{c}_i||_2}, \ \mathbf{S}^u = \frac{\mathbf{S}^u}{||\mathbf{S}^u||_2}. \tag{5}$$

Subsequently, we delineate the implementation details of the *Constraint for Long-tail* and the *Constraint for Accuracy*, introducing their formulations and roles in the optimization framework.

Constraint for Long-tail The long-tail problem emerges due to the pronounced disparity in session-item similarity between tail and head items, as documented in previous research (Yin et al. 2012; Liu and Zheng 2020b). Based on this

observation, we propose a novel constraint: minimizing the variance of similarity scores between sessions and items belonging to the target intent. This constraint can reduce the divergence in similarity distributions between session-to-head and session-to-tail, thereby promoting more balanced recommendation performance. Formally, the constraint is defined as follows:

Definition 3 (Constraint for Long-tail). *Given the session embedding $\boldsymbol{S}^u$, the variance of its Euclidean distances to the embeddings of all items belonging to its target intent should be minimized, which can be formulated as:*

$$\min\ \mathcal{L}_l = \text{Var}_{v_i \in \mathcal{C}^u}\left[d(\mathbf{S}^u, \mathbf{v}_i)\right]. \tag{6}$$

where Var is the variance calculation, and $d(x, y)$ measures the Euclidean distance between x and y. The time complexity of the above operation is $O(Nd)$, where N is the number of items belonging to the target intent $\mathcal{C}^u$, and d is the embedding dimension. Since HID is a model-agnostic plugin, the complexity is a key concern. Therefore, we further propose an approximate formulation of Equation (6) with lower complexity. As shown in the following theorem:

Theorem 1 (Optimizing Equivalence). *The Equation (6) with time complexity of $O(Nd)$ can be approximated to an equation with time complexity of $O(d)$ during the optimization process:*

$$\min\ \mathcal{L}_l = \text{Var}_{v_i \in \mathcal{C}^u}[d(\mathbf{S}^u, \mathbf{v}_i)] \ \sim\ \min\ d(\mathbf{S}^u, \mathbf{c}^u). \tag{7}$$

where $\mathbf{c}^u$ is the embedding of the target intent. The detailed proof of Theorem 1 is provided in **Appendix A**. Given the Theorem 1, the optimization process that maximizes the similarity between the session embedding $\mathbf{S}^u$ and the target intent embedding $\mathbf{c}^u$ is mathematically equivalent to solving Equation (6). This concise constraint provides an efficient mechanism for enhancing tail item coverage within the target intent space while excluding noise intents.

Constraint for Accuracy To further mitigate session-irrelevant recommendations, it is crucial to proactively limit the presence of noise items in the recommendations. Therefore, we propose minimizing the mean of similarity scores between sessions and noise intents. Besides, to prevent extreme cases, the variance of similarity scores should also be constrained. By regulating both the mean and variance, we ensure that the noise intent distribution remains distant from the specific sessions. This constraint can be formulated as:

Definition 4 (Constraint for Accuracy). *Given the session representation $\boldsymbol{S}^u$, the mean and vairance of its Euclidean distances to the representations of noise intents within the same batch should be maximized and restricted, respectively, which can be formulated as:*

$$\begin{aligned} &\max\ \mathcal{L}_a = \mathbb{E}_{c^v \in \hat{\mathcal{C}}^u} d(\mathbf{S}^u, \mathbf{c}^v) \propto \sum_{c^v \in \hat{\mathcal{C}}^u} d(\mathbf{S}^u, \mathbf{c}^v), \\ &\text{s.t. } \text{Var}_{c^v \in \hat{\mathcal{C}}^u}\left(d(\mathbf{S}^u, \mathbf{c}^v)\right) < \eta, \end{aligned} \tag{8}$$

where η is the threshold of variance.

Intent Constraint Loss Optimizing these two constraints independently presents certain challenges. Therefore, we combine the objectives of Equation (7) and Equation (8), unifying them into a single loss function:

$$\begin{aligned} &\min\ \mathcal{L}_c = \sum_{S^u \in \mathcal{B}} \log \frac{\exp(d(\mathbf{S}^u, \mathbf{c}^u))}{\exp(d(\mathbf{S}^u, \mathbf{c}^u)) + \sum\limits_{c^v \in \hat{\mathcal{C}}^u} \exp(d(\mathbf{S}^u, \mathbf{c}^v))}, \\ &\text{s.t. } \text{Var}_{c^v \in \hat{\mathcal{C}}^u}(d(\mathbf{S}^u, \mathbf{c}^v)) < \eta, \end{aligned} \tag{9}$$

where exp(x) is leveraged to amplify the difference between the target and noise intents, $\exp(d(\mathbf{S}^u, \mathbf{c}^u))$ is incorporated into the denominator to stabilize the loss range and mitigate the impact of the number of negative samples on the loss scale. To further minimize the effect of the noise intents, we give another theorem:

Theorem 2 (Triplet Loss Approximation). *The optimization of the objective function in Equation (9) is approximately proportional to optimize a (N-1)-triplet loss with a fixed margin of 2*:

$$\mathcal{L}_c \propto \sum_{S^u \in \mathcal{B}} \sum_{c^v \in \hat{\mathcal{C}}^u} \left(\|\mathbf{S}^u - \mathbf{c}^u\|^2 - \|\mathbf{S}^u - \mathbf{c}^v\|^2 + \mathbf{2}\right). \tag{10}$$

The proof of Theorem 2 is given in **Appendix B**. The constant term '2' is the fixed margin that decides the distinction of $d(\mathbf{S}^u, \mathbf{c}^u)$ and $d(\mathbf{S}^u, \mathbf{c}^v)$. However, this fixed margin is inadequate for distinguishing the target and noise intents, especially in scenarios with high variability in intent distributions or in the presence of ambiguous intents. Therefore, we introduce a flexible coefficient to replace the original constant, enabling flexible margin adjustment based on the recommendation scenario:

$$\min\ \mathcal{L}_c = \tag{11}$$

$$\sum_{S^u \in \mathcal{B}} \log \frac{\exp(d(\mathbf{S}^u, \mathbf{c}^u)/\sigma)}{\exp(d(\mathbf{S}^u, \mathbf{c}^u)/\sigma) + \sum\limits_{c^v \in \hat{\mathcal{C}}^u} \exp(d(\mathbf{S}^u, \mathbf{c}^v)/\sigma)}, \tag{12}$$

$$\text{s.t. } \text{Var}_{c^v \in \hat{\mathcal{C}}^u}(d(\mathbf{S}^u, \mathbf{c}^v)) < \eta, \tag{13}$$

where σ is the flexible coefficient. To directly apply the gradient descent for updates and avoid the complexity of constraint optimization, we reformulate the hard variance constraint $\text{Var}_{c^v \in \hat{\mathcal{C}}^u}(d(\mathbf{S}^u, \mathbf{c}^v)) < \eta$ as a penalty term p^u:

$$p^u = \max(0, \operatorname*{Var}_{c^v \in \hat{\mathcal{C}}^u}(d(\mathbf{S}^u, \mathbf{c}^v)) - \eta). \tag{14}$$

In addition, previous research has found that cosine similarity can achieve better alignment and uniformity of embeddings (Wang and Isola 2020). Therefore, we adopt cosine similarity instead of Euclidean distance. The final training objective of the intent constraint loss (ICLoss) is formulated as:

$$\min\ \mathcal{L}_c = -\sum_{S^u \in \mathcal{B}} \log \frac{\mathbf{X}}{(1 + \lambda p^u)(\mathbf{X} + \mathbf{Y})}, \tag{15}$$

where λ is the hyper-parameter that controls penalty, $\mathbf{X}$ is $\exp(\cos(\mathbf{S}^u, \mathbf{c}^u)/\sigma)$, $\mathbf{Y}$ is $\sum\limits_{c^v \in \hat{\mathcal{C}}^u} \exp(\cos(\mathbf{S}^u, \mathbf{c}^v)/\sigma)$, and p^u is rescaled within (0,1). The equivalence of Euclidean distance and cosine similarity is ensured by the L_2 normalization of Equation (5).

Datasets		Tmall						Diginetica						Retailrocket					
Metrics		Accuracy		Long-tail				Accuracy		Long-tail				Accuracy		Long-tail			
SBR Models	Methods	HR	MRR	tHR	tMRR	tCov	Tail	HR	MRR	tHR	tMRR	tCov	Tail	HR	MRR	tHR	tMRR	tCov	Tail
STAMP (Sequential)	*base*	26.10	14.67	25.98	14.61	69.46	77.77	50.15	17.24	47.81	16.56	90.71	68.70	50.54	26.34	49.66	26.40	53.70	68.68
	+ *TailNet*	20.61	9.91	20.77	10.01	71.33	78.01	45.39	14.79	42.68	14.89	91.23	68.21	47.00	24.37	46.21	24.21	51.56	63.76
	+ *CSBR*	25.43	14.20	25.46	14.28	69.15	77.58	49.86	17.28	47.80	16.48	91.61	68.66	49.82	25.96	49.51	25.93	54.51	70.65
	+ *LOAM*	24.31	13.80	24.37	13.74	71.68	77.23	46.19	15.28	43.39	14.50	89.96	**70.26**	50.27	26.13	49.51	26.27	55.67	71.79
	+ *LAP-SR*	25.21	14.13	25.24	14.20	72.11	77.61	49.87	17.16	47.69	16.37	91.32	68.55	49.59	25.89	48.78	25.93	55.32	71.41
	+ HID	**28.26**	**15.84**	**28.35**	**15.93**	**73.65**	**78.19**	**50.39**	**17.58**	**48.09**	**17.28**	**93.05**	69.24	**52.38**	**27.99**	**52.09**	**28.34**	**56.02**	**72.59**
	p-value (<)	0.001	0.001	0.001	0.001	0.001	0.05	0.05	0.05	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001	0.001
GRU4Rec (Sequential)	*base*	19.69	9.58	19.53	9.57	49.60	71.80	50.23	16.96	47.49	15.79	84.97	65.14	45.01	24.33	44.12	23.78	69.98	73.29
	+ *TailNet*	17.21	8.25	17.09	8.18	52.31	73.42	46.51	15.30	45.36	14.29	87.91	67.58	43.09	22.98	42.28	22.53	70.62	73.66
	+ *CSBR*	19.90	10.11	20.00	10.27	53.22	78.52	49.92	16.68	47.01	15.43	88.74	68.24	43.39	23.17	43.41	22.71	70.99	74.21
	+ *LOAM*	18.40	9.14	18.65	9.31	58.76	**79.57**	47.53	15.65	45.79	14.84	**91.65**	**71.49**	45.32	24.21	44.37	23.89	72.29	75.19
	+ *LAP-SR*	19.41	9.33	19.37	9.29	56.32	76.12	49.91	16.50	46.98	15.31	90.21	68.03	44.59	24.02	43.67	23.61	71.45	74.03
	+ HID	**25.13**	**13.95**	**25.21**	**13.98**	**63.21**	77.92	**52.23**	**17.79**	**50.92**	**16.83**	90.73	68.92	**48.89**	**26.43**	**47.91**	**26.19**	**73.21**	**75.89**
	p-value (<)	0.001	0.001	0.001	0.001	0.001	-	0.001	0.001	0.001	0.001	-	-	0.001	0.001	0.001	0.001	0.001	0.005
SRGNN (Graphic)	*base*	27.45	14.27	27.12	14.32	53.60	77.65	51.47	17.95	49.04	17.01	94.16	68.85	50.55	26.88	49.16	26.24	53.96	69.94
	+ *TailNet*	25.79	13.05	25.81	13.39	64.01	76.33	49.86	17.42	48.21	16.98	88.97	65.77	47.87	25.14	47.11	24.78	54.13	71.47
	+ *CSBR*	26.98	13.83	26.89	13.90	53.00	77.61	51.22	17.89	49.16	17.02	93.89	68.94	49.93	26.55	48.76	25.79	55.32	71.65
	+ *LOAM*	26.33	13.52	26.56	13.75	**69.95**	77.23	49.27	17.19	48.03	16.70	95.92	**72.11**	50.29	26.81	49.02	26.20	**56.16**	**73.97**
	+ *LAP-SR*	26.76	13.95	26.89	14.05	61.35	75.38	51.04	17.84	48.86	16.92	95.22	71.94	50.32	26.37	48.76	26.02	54.99	71.59
	+ HID	**28.38**	**14.66**	**28.13**	**14.50**	66.40	**78.12**	**52.09**	**18.26**	**49.79**	**17.25**	**96.22**	70.05	**53.45**	**29.47**	**52.61**	**29.51**	55.75	73.54
	p-value (<)	0.001	0.001	0.001	0.05	-	0.01	0.001	0.001	0.001	0.05	0.005	-	0.001	0.001	0.001	0.001	-	-
GCEGNN (Graphic)	*base*	32.42	13.98	32.35	13.94	81.99	77.49	53.84	18.87	51.55	18.04	91.43	45.82	54.97	28.47	54.61	28.13	72.54	72.13
	+ *TailNet*	29.91	12.50	29.70	12.41	83.76	77.91	47.47	16.78	45.59	15.90	92.13	46.01	53.19	27.57	52.87	27.33	73.85	72.73
	+ *CSBR*	29.60	14.07	29.37	13.86	82.80	78.21	52.24	18.06	50.13	17.35	93.81	46.32	52.26	26.90	51.99	26.45	73.23	72.59
	+ *LOAM*	30.96	13.54	30.79	13.47	**84.97**	78.11	52.31	17.42	50.19	16.77	93.01	46.13	53.78	27.81	53.47	27.56	**75.11**	74.47
	+ *LAP-SR*	32.11	13.67	32.05	13.63	82.03	77.89	53.19	18.20	50.89	17.51	92.44	45.91	53.26	27.40	52.96	27.02	74.02	73.97
	+ HID	**33.53**	**14.43**	**33.31**	**14.37**	83.25	**78.70**	**54.22**	**19.18**	**51.83**	**18.37**	**94.21**	**46.67**	**55.37**	**28.82**	**54.99**	**28.59**	74.74	**74.89**
	p-value (<)	0.001	0.001	0.001	0.001	-	0.001	0.005	0.001	0.005	0.05	0.001	0.05	0.001	0.001	0.05	0.005	-	0.01

Table 1: The accuracy and long-tail performance (K=20) of SBR models with long-tail methods over three datasets. Bold labeled scores indicate the best results for each dataset under certain baseline and underlined scores represent second-best results. The p-value is calculated through two-sided t-test.

Datasets		Tmall						Diginetica						Retailrocket					
SBR Model	Comparisons	HR	MRR	tHR	tMRR	tCov	Tail	HR	MRR	tHR	tMRR	tCov	Tail	HR	MRR	tHR	tMRR	tCov	Tail
STAMP	HID	**28.26**	**15.84**	**28.35**	**15.93**	**73.65**	**78.19**	**50.39**	**17.38**	**48.09**	**17.28**	**93.05**	**69.24**	**52.38**	**27.99**	**52.09**	**28.34**	**56.02**	**72.59**
	HID w/o HI	27.43	15.24	27.56	15.40	69.29	77.98	50.17	17.24	47.96	17.19	91.96	68.83	51.75	27.37	51.51	27.82	55.31	71.80
	HID w/o FC	26.77	14.86	26.86	15.09	70.20	77.94	49.76	17.24	47.52	16.31	92.15	68.91	50.89	26.51	50.60	26.71	55.67	72.16
SRGNN	HID	**28.38**	**14.66**	**28.13**	**14.50**	**66.40**	**78.12**	**52.09**	**18.26**	**49.79**	**17.25**	**96.02**	**70.05**	**53.45**	**29.47**	**52.61**	**29.51**	**55.75**	**73.54**
	HID w/o HI	27.48	14.34	27.31	14.36	61.00	77.12	51.96	18.01	49.46	17.03	92.94	68.57	53.10	29.18	52.27	29.21	54.01	72.77
	HID w/o FC	27.36	14.33	27.23	14.27	62.92	77.49	51.16	17.40	48.90	16.41	93.56	69.10	52.80	28.79	51.98	28.83	55.11	73.03

Table 2: Ablation study on Tmall, Diginetica and Retailrocket.

Multi-task Learning. To incorporate HID into traditional SBR models, we introduce a multi-task learning loss to combine the learning of ICLoss with the typically used cross-entorpy loss. Specifically, a hyper-parameter ϵ is introduced to control the scale of ICLoss. The total loss can be expressed as: $\mathcal{L} = \mathcal{L}_p + \epsilon\mathcal{L}_c$. Besides, the time complexity analysis of HID is provided in **Appendix** C.

Experiments

Datasets. We evaluate our proposed HID with the three real-world datasets, namely $Tmall$[1], $RetailRocket$[2], $Diginetica$[3]. *Tmall* is from the IJCAI-15 competition and consists of shopping logs of unnamed users on the Tmall online shopping platform. *RetailRocket* RetailRocketis released by an e-commerce corporation for the Kaggle competition and contains users' browsing activity. *Diginetica* comes from CIKM Cup 2016.

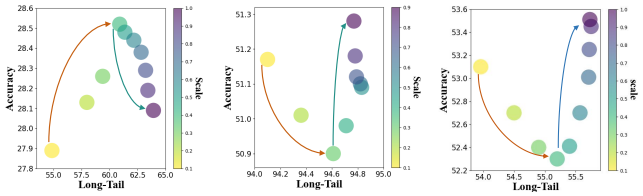
Base Models and Competitors. To demonstrate the effectiveness of our proposed HID, we select some well-known SBR models from both sequential approaches (GRU4Rec (Hidasi et al. 2016), STAMP (Liu et al. 2018)) and graphic approaches (SR-GNN (Wu et al. 2019), GCE-GNN (Wang et al. 2020)) as the base SBR models. Apart from the above base SBR models, we also introduce Tail-Net (Liu and Zheng 2020a), CSBR (Chen et al. 2023), LOAM (Yang et al. 2023), LAP-SR (Peng and Zhou 2024) as the plug-and-play long-tail competitors.

Metrics. To evaluate the recommendation accuracy and long-tail performance, we employ three widely used accuracy metrics, including the HR@K, and MRR@K. Following previous works on long-tail issue (Abdollahpouri, Burke, and Mobasher 2019; Liu and Zheng 2020a; Yang

[1]https://tianchi.aliyun.com/dataset/dataDetail?dataId=42

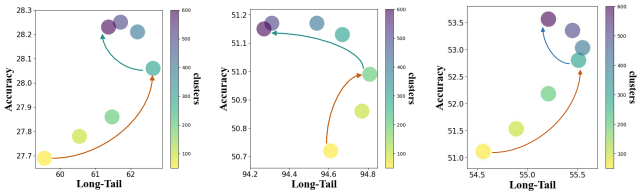
[2]https://www.kaggle.com/retailrocket/ecommerce-dataset

[3]https://competitions.codalab.org/competitions/11161



(a) Tmall (b) Diginetica (c) Retailrocket

Figure 4: The changes in accuracy (HR@20) and long-tail (tCov@20) metrics with the increase of scale ϵ. The model is SRGNN+HID.



(a) Tmall (b) Diginetica (c) Retailrocket

Figure 5: The changes in accuracy (HR@20) and long-tail (tCov@20) metrics with the increase of clusters n. The model is SRGNN+HID.

et al. 2023), we introduce some well-known long-tail metrics, including tHR@K, tMRR@K, tCov@K, and Tail@K.

More details on the preprocessing process, baselines, metrics and implementation details are given in **Appendix** D.

Ablation Study

To investigate our proposed method, we construct two variants of our proposed method which are the HID w/o HI (i.e., Hybrid Intent) where the hybrid intent is substituted with the commonly used intent definition based on the last few items (3 in this experiment, and average pooling is adopted to aggregate them) of each session (Zhang et al. 2023), and the HID w/o FC (i.e., Flexible Coefficient) where the flexible coefficient σ is dropped. Experiments are demonstrated in Table 2. Overall, both HID w/o HI and HID w/o FC exhibit performance degradation compared to HID across the two SBR models and datasets. Removing HI impacts diversity more, while removing FC affects accuracy more, consistent with our prior analysis. Furthermore, the hybrid intents have greater impact on Tmall than on Diginetica/Retailrocket, as its longer sessions exhibit more frequent intent shifts, making target intent modeling crucial.

Overall Performance

Refer to results in Table 1, we draw following conclusions:

For Previous Work. The results indicate that almost all existing long-tail approaches improve long-tail performance with the *sacrifice of accuracy* compared with base SBR models. This trade-off arises from their neglect of the substantial amount of session-irrelevant items, which introduces noise into the recommendations when prioritizing tail items.

For Our Proposed HID. Compared with previous appraoches, SBR models with HID demonstrate improvements in both accuracy and long-tail performance. This improvement arises from two aspects: (i) The representative hybrid intent endows HID with the capability to perceive users' high-level intents, providing a solid foundation for the effectiveness of the overall framework; (ii) The intent constraint loss effectively emphasizes tail items within the target intent while driving session representations away from noise distributions, thus achieving accurate long-tail SBR.

Hyperparameter Exploration

Balance between ICLoss and CE Loss. We systematically study the balance between cross-entropy loss and ICLoss by tuning the scaling parameter n from 0.1 to 0.9. As shown in Figure 4, the Tmall dataset demonstrates distinct behavior: both accuracy (HR@20) and long-tail performance (tCov@20) improve as clusters n increases from 0 to 0.4, beyond which accuracy declines while long-tail performance continues to improve, establishing n=0.4 as the optimal trade-off point. In contrast, Diginetica and Retailrocket exhibit different patterns - their long-tail performance initially improves then stabilizes with increasing n, while accuracy shows non-monotonic variations. Therefore, on SRGNN, for the Tmall dataset, increasing the weight of ICLoss in the range from 0.1 to 0.4 can further improve both accuracy and long-tail performance. For Diginetica and Retailrocket, the range is from 0.3 to 0.9 and 0.4 to 0.9.

Number of Hybrid Intents. In this section, we investigate the impact of cluster numbers (i.e., number of hybrid intents) n of spectral clustering on the recommendation accuracy and long-tail performance. As shown in Figure 5, on both Tmall and Diginetica, we observe the same trend that as the number of hybrid intents increasing, the accuracy increases initially and then stabilizes while the long-tail performance exhibits a peak-shaped pattern, reaching its maximum when the number of clusters is 4 for Tmall and Retailrocket, and 3 for Diginetica. This indicates that when the number of hybrid intents increases (i.e., each intent contains fewer items), more items are classified as noise intents. As a result, HID is able to exclude more noise items from the recommendation list. However, for the diversity metric, there is less items belonging to the target intent, which leads to fewer long-tail items being considered by HID, causing a decline in long-tail performance.

Replacing Attribute with Semantic Clusters

We conduct additional experiments to investigate the performance of HID without attributes in the **Appendix D**.

Conclusion

This paper addresses the challenge of balancing long-tail performance and recommendation accuracy in traditional SBR methods by proposing a Hybrid Intent-based Dual

Constraint Framework (HID), transforming the typical "see-saw" into the "win-win". We introduce two novel constraints targeting both long-tail performance and recommendation accuracy, enforced through a hybrid intent learning process that captures both the attributes of items and actions of anonymous users. Additionally, we propose the intent constraint loss (ICLoss), which guides session representation learning and integrates seamlessly with existing SBR models. Extensive experiments on multiple baselines and datasets validate effectiveness of HID, proving that it can improve both accuracy and long-tail performance for SBR.

References

Abdollahpouri, H.; Burke, R.; and Mobasher, B. 2019. Managing Popularity Bias in Recommender Systems with Personalized Re-Ranking. In *FLAIRS*, 413–418. Sarasota, Florida, USA: AAAI Press.

Box, G. E.; and Meyer, R. D. 1986. An analysis for unreplicated fractional factorials. *Technometrics*, 28(1): 11–18.

Chen, J.; Wu, W.; Shi, L.; Zheng, W.; and He, L. 2023. Long-tail session-based recommendation from calibration. *Appl. Intell.*, 53(4): 4685–4702.

Chen, Y.; Liu, Z.; Li, J.; McAuley, J. J.; and Xiong, C. 2022. Intent Contrastive Learning for Sequential Recommendation. In *TheWebConf*, 2172–2182. Lyon, France: ACM.

Choi, M.; Kim, H.; Cho, H.; and Lee, J. 2024a. Multi-intent-aware Session-based Recommendation. In *SIGIR*, 2532–2536. Washington DC, USA: ACM.

Choi, M.; Kim, H.; Cho, H.; and Lee, J. 2024b. Multi-intent-aware Session-based Recommendation. In *SIGIR*, 2532–2536. Washington DC, USA: ACM.

Gupta, P.; Garg, D.; Malhotra, P.; Vig, L.; and Shroff, G. M. 2019. NISER: normalized item and session representations with graph neural networks. *arXiv preprint arXiv:1909.04276*, 43: 128–134.

Hidasi, B.; Karatzoglou, A.; Baltrunas, L.; and Tikk, D. 2016. Session-based recommendations with recurrent neural networks. In *ICLR*. San Juan, Puerto Rico: OpenReview.net.

Hou, Y.; Hu, B.; Zhang, Z.; and Zhao, W. X. 2022. CORE: Simple and Effective Session-based Recommendation within Consistent Representation Space. In *SIGIR*, 1796–1801. Madrid, Spain: ACM.

Huang, Y.; Yang, Z.; Hu, W.; Xu, B.; and Zhang, Z. 2024. CSLP: Collaborative Solution to Long-Tail Problem and Popularity Bias in Sequential Recommendation. In *SMC*, 4404–4411. Kuching, Malaysia: IEEE.

Kim, K.; Hyun, D.; Yun, S.; and Park, C. 2023. MELT: Mutual Enhancement of Long-Tailed User and Item for Sequential Recommendation. In *SIGIR*, 68–77. Taipei, Taiwan: ACM.

Latifi, S.; Mauro, N.; and Jannach. 2021. Session-aware recommendation: A surprising quest for the state-of-the-art. *Information Sciences*, 573: 291–315.

Lee, G.; Kim, K.; and Shin, K. 2024. Post-Training Embedding Enhancement for Long-Tail Recommendation. In *CIKM*, 3857–3861. Boise, ID, USA: ACM.

Li, H.; Wang, X.; Zhang, Z.; Ma, J.; Cui, P.; and Zhu, W. 2023. Intention-aware Sequential Recommendation with Structured Intent Transition. In *ICDE*, 3759–3760. Anaheim, CA, USA: IEEE.

Li, J.; Ren, P.; Chen, Z.; Ren, Z.; Lian, T.; and Ma, J. 2017. Neural Attentive Session-based Recommendation. In *CIKM*, 1419–1428. Singapore: ACM.

Li, Z.; Yang, C.; Chen, Y.; Wang, X.; Chen, H.; Xu, G.; Yao, L.; and Sheng, M. 2025. Graph and Sequential Neural Networks in Session-based Recommendation: A Survey. *ACM Comput. Surv.*, 57(2): 40:1–40:37.

Lin, G.; Meng, Z.; Wang, D.; Long, Q.; Zhou, Y.; and Xiao, M. 2024. GUME: Graphs and User Modalities Enhancement for Long-Tail Multimodal Recommendation. In *=CIKM=*, 1400–1409. Boise, ID, USA: ACM.

Liu, Q.; Wu, X.; Wang, Y.; Zhang, Z.; Tian, F.; Zheng, Y.; and Zhao, X. 2024. LLM-ESR: Large Language Models Enhancement for Long-tailed Sequential Recommendation. In *NeurIPS 2024*. Vancouver, BC, Canada.

Liu, Q.; Zeng, Y.; Mokhosi, R.; and Zhang, H. 2018. STAMP: short-term attention/memory priority model for session-based recommendation. In *SIGKDD*, 1831–1839. London,United Kingdom: ACM.

Liu, S.; and Zheng, Y. 2020a. Long-tail Session-based Recommendation. In *RecSys*, 509–514. Virtual Event, Brazil: ACM.

Liu, S.; and Zheng, Y. 2020b. Long-tail Session-based Recommendation. In *RecSys*, 509–514. Brazil: ACM.

Luo, S.; Ma, C.; Xiao, Y.; and Song, L. 2023. Improving Long-Tail Item Recommendation with Graph Augmentation. In *CIKM*, 1707–1716. Birmingham, United Kingdom: ACM.

Pan, Z.; Cai, F.; Chen, W.; Chen, H.; and de Rijke, M. 2020. Star Graph Neural Networks for Session-based Recommendation. In *CIKM*, 1195–1204. Virtual Event, Ireland: ACM.

Peng, D.; and Zhou, Y. 2024. A long-tail alleviation post-processing framework based on personalized diversity of session recommendation. *Expert Syst. Appl.*, 249: 123769.

Qiu, R.; Li, J.; Huang, Z.; and Yin, H. 2019. Rethinking the Item Order in Session-based Recommendation with Graph Neural Networks. In *CIKM*, 579–588. Beijing, China: ACM.

Sundaresan, N. 2011. Recommender systems at the long tail. In *RecSys*, 1–6. Chicago, IL, USA: ACM.

Turgut, H.; Yetki, T. D.; Bali, Ö.; and Yücel, T. A. 2023. Prod2Vec-Var: A Session Based Recommendation System with Enhanced Diversity. In *CIKM*, 5253–5254. Birmingham, United Kingdom: ACM.

Wang, S.; Hu, L.; Wang, Y.; Sheng, Q. Z.; Orgun, M. A.; and Cao, L. 2019. Modeling Multi-Purpose Sessions for Next-Item Recommendations via Mixture-Channel Purpose Routing Networks. In *IJCAI*, 3771–3777. Macao, China: ijcai.org.

Wang, T.; and Isola, P. 2020. Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere. In *ICML*, volume 119, 9929–9939. Virtual Event: PMLR.

Wang, W.; Feng, F.; He, X.; Nie, L.; and Chua, T. 2021. Denoising Implicit Feedback for Recommendation. In *WSDM*, 373–381. Virtual Event, Israel: ACM.

Wang, X.; Dai, T.; Liu, Q.; and Liang, S. 2024. Spatial-Temporal Perceiving: Deciphering User Hierarchical Intent in Session-Based Recommendation. In *IJCAI*, 2415–2423. Jeju, South Korea: ijcai.org.

Wang, Z.; Wei, W.; Cong, G.; Li, X.; Mao, X.; and Qiu, M. 2020. Global Context Enhanced Graph Neural Networks for Session-based Recommendation. In *SIGIR*, 169–178. Virtual Event, China: ACM.

Wei, W.; Ren, X.; Tang, J.; Wang, Q.; Su, L.; Cheng, S.; Wang, J.; Yin, D.; and Huang, C. 2024. LLMRec: Large Language Models with Graph Augmentation for Recommendation. In *WSDM*, 806–815. Merida, Mexico: ACM.

Wu, S.; Tang, Y.; Zhu, Y.; Wang, L.; Xie, X.; and Tan, T. 2019. Session-Based Recommendation with Graph Neural Networks. In *AAAI*, 346–353. Honolulu, Hawaii, USA: AAAI Press.

Xia, X.; Yin, H.; Yu, J.; Shao, Y.; and Cui, L. 2021a. Self-Supervised Graph Co-Training for Session-based Recommendation. In *CIKM*, 2180–2190. Queensland, Australia: ACM.

Xia, X.; Yin, H.; Yu, J.; Wang, Q.; Cui, L.; and Zhang, X. 2021b. Self-Supervised Hypergraph Convolutional Networks for Session-based Recommendation. In *AAAI*, 4503–4511. Virtual Event: AAAI Press.

Yang, H.; Choi, Y.; Kim, G.; and Lee, J. 2023. LOAM: Improving Long-tail Session-based Recommendation via Niche Walk Augmentation and Tail Session Mixup. In *SIGIR*, 527–536. Taipei,Taiwan: ACM.

Yin, H.; Cui, B.; Li, J.; Yao, J.; and Chen, C. 2012. Challenging the Long Tail Recommendation. *Proc. VLDB Endow.*, 5(9): 896–907.

Yin, Q.; Fang, H.; Sun, Z.; and Ong, Y. 2024. Understanding Diversity in Session-based Recommendation. *ACM Trans. Inf. Syst.*, 42(1): 24:1–24:34.

Yuan, J.; Song, Z.; Sun, M.; Wang, X.; and Zhao, W. X. 2021. Dual Sparse Attention Network For Session-based Recommendation. In *AAAI*, 4635–4643. Virtual Event: AAAI Press.

Zhang, P.; Guo, J.; Li, C.; Xie, Y.; Kim, J.; Zhang, Y.; Xie, X.; Wang, H.; and Kim, S. 2023. Efficiently Leveraging Multi-level User Intent for Session-based Recommendation via Atten-Mixer Network. In *WSDM*, 168–176. Singapore: ACM.

A Proof of Theorem 1 (Optimizing Equivalence)

Theorem 1 (Optimizing Equivalence). *The Equation (6) (in the main paper file) with time complexity of $O(N \times d)$ can be approximated to an equation with time complexity of $O(d)$:*

$$\min\ \mathcal{L}_l = \mathrm{Var}_{v_i \in \mathcal{C}^u}[d(\mathbf{S}^u, \mathbf{v}_i)] \ \sim\ \min\ d(\mathbf{S}^u, \mathbf{c}^u). \quad (16)$$

Proof. The Theorem 1 can be proved as follows:

$$\begin{aligned}\mathcal{L}_l &= \mathrm{Var}_{\mathbf{v}_i \in \mathcal{C}^u}[d(\mathbf{S}^u, \mathbf{v}_i)] \\ &= \sum_{\mathbf{v}_i \in \mathcal{C}^u} \frac{\|\mathbf{S}^u - \mathbf{v}_i\|^2}{|\mathcal{C}^u|} - \left(\sum_{\mathbf{v}_i \in \mathcal{C}^u} \frac{\|\mathbf{S}^u - \mathbf{v}_i\|}{|\mathcal{C}^u|} \right)^2 .\end{aligned} \quad (17)$$

Then, we calculate the gradient of the $\mathcal{L}_l$:

$$\begin{aligned}\nabla \mathcal{L}_l = \ & 2(\mathbf{S}^u - \sum_{\mathbf{v}_i \in \mathcal{C}^u} \frac{\mathbf{v}_i}{|\mathcal{C}^u|}) - 2 \left(\sum_{\mathbf{v}_i \in \mathcal{C}^u} \frac{\|\mathbf{S}^u - \mathbf{v}_i\|}{|\mathcal{C}^u|} \right) \times \\ & \left(\sum_{\mathbf{v}_i \in \mathcal{C}^u} \frac{\mathbf{S}^u - \mathbf{v}_i}{|\mathcal{C}^u| \cdot \|\mathbf{S}^u - \mathbf{v}_i\|} \right).\end{aligned} \quad (18)$$

After setting it to zero, we obtain the expression for $\mathbf{S}^u$:

$$\begin{aligned}\mathbf{S}^u = & \sum_{\mathbf{v}_i \in \mathcal{C}^u} \frac{\mathbf{v}_i}{|\mathcal{C}^u|} + \left(\sum_{\mathbf{v}_i \in \mathcal{C}^u} \frac{\|\mathbf{S}^u - \mathbf{v}_i\|}{|\mathcal{C}^u|} \right) \times \\ & \left(\sum_{\mathbf{v}_i \in \mathcal{C}^u} \frac{\mathbf{S}^u - \mathbf{v}_i}{|\mathcal{C}^u| \cdot \|\mathbf{S}^u - \mathbf{v}_i\|} \right).\end{aligned} \quad (19)$$

Considering that the second term $\sum_{\mathbf{v}_i \in \mathcal{C}^u} \frac{\mathbf{S}^u - \mathbf{v}_i}{|\mathcal{C}^u| \cdot \|\mathbf{S}^u - \mathbf{v}_i\|} \sim 0$ when the sum of the unit vectors pointing from $\mathbf{S}^u$ to each point $\mathbf{v}_i$ is about 0, which can be satisfied when $\mathbf{v}_i$ exhibit an approximately symmetric distribution around $\mathbf{S}^u$, indicating that $\mathbf{S}^u$ is the centorid of all $\mathbf{v}_i$. Observing the Equation (18), when the second term is approximated to 0, S^u will be approximately equal to the first term $\sum_{\mathbf{v}_i \in \mathcal{C}^u} \frac{\mathbf{v}_i}{|\mathcal{C}^u|}$, which is exactly the centorid of all $\mathbf{v}_i$.

In that case, $\mathbf{S}^u = \sum_{\mathbf{v}_i \in \mathcal{C}^u} \frac{\mathbf{v}_i}{|\mathcal{C}^u|}$ is an approximate solution of optimizing Equation (16). Referring to Equation (2), the target intent embedding is aquired through the average pooling: $\mathbf{c}^u = \sum_{v_i \in \mathcal{C}^u} \frac{\mathbf{v}_i}{|\mathcal{C}^u|}$. Therefore, minimizing $d(\mathbf{S}^u, \mathbf{c}^u)$ can be approximately equal to minimizing Equation (6).

B Proof of Theorem 2 (Triplet Loss Approximation)

Theorem 2 (Triplet Loss Approximation). *The optimization of the objective function in Equation (9) is approximately proportional to optimize a (N-1)-triplet loss with a fixed margin of 2*:

$$\mathcal{L}_c \propto \sum_{S^u \in \mathcal{B}} \sum_{c^v \in \hat{\mathcal{C}}^u} \left(\|\mathbf{S}^u - \mathbf{c}^u\|^2 - \|\mathbf{S}^u - \mathbf{c}^v\|^2 + \mathbf{2} \right), \quad (20)$$

Proof. Since the $\mathbf{S}^u$ and $\mathbf{c}^u$ have been L_2 normalized before sending to ICLoss, minimizing $d(\mathbf{S}^u, \mathbf{c}^u)$ is equivalent to maximizing $\mathbf{S}^u \cdot \mathbf{c}^u$. Then, the Theorem 2 can be proved

as follows:

$$\begin{aligned}\mathcal{L}_c &= -\sum_{S^u\in\mathcal{B}} \log \frac{\exp(\mathbf{S}^u\cdot\mathbf{c}^u)}{\exp(\mathbf{S}^u\cdot\mathbf{c}^u)+\sum_{c^v\in\hat{\mathcal{C}}^u}\exp(\mathbf{S}^u\cdot\mathbf{c}^v)},\\ &= \sum_{S^u\in\mathcal{B}} \log[1+\sum_{c^v\in\hat{\mathcal{C}}^u}\exp(\mathbf{S}^u\cdot\mathbf{c}^v-\mathbf{S}^u\cdot\mathbf{c}^u)],\\ &\simeq \sum_{S^u\in\mathcal{B}}\sum_{c^v\in\hat{\mathcal{C}}^u}\exp(\mathbf{S}^u\cdot\mathbf{c}^v-\mathbf{S}^u\cdot\mathbf{c}^u),\\ &\simeq \sum_{S^u\in\mathcal{B}}\sum_{c^v\in\hat{\mathcal{C}}^u}(\mathbf{S}^u\cdot\mathbf{c}^v-\mathbf{S}^u\cdot\mathbf{c}^u+1),\\ &\propto \sum_{S^u\in\mathcal{B}}\sum_{c^v\in\hat{\mathcal{C}}^u}(||\mathbf{S}^u-\mathbf{c}^u||^2-||\mathbf{S}^u-\mathbf{c}^v||^2+2).\end{aligned}$$

C Time Complexity Analysis

The main components of HID are the hybrid intent learning module and the intent constraint loss. In the hybrid intent learning module, since the item attributes and the connections between attributes across all sessions can be pre-obtained from the dataset, we construct the preliminary intent graph for each dataset and store the results of spectral clustering in advance. In that case, the complexity of this module arises solely from the average pooling used to obtain the hybrid intent representation, which is $O(md)$. For the intent constraint loss, for each batch, the complexity of the *Constraint for long-tail* is $O(Bd)$ where B is the batch size, the complexity of the *Constraint for Accuracy* is $O(BKd)$ where K is the average number of noise intents for sessions. Therefore, the complexity of the intent constraint loss for each batch is $O(B(d+Kd)) = O(BKd)$ since K is typically larger than d.

D Experimental Details

D.1 Preprocess of the Datasets.

Following (Xia et al. 2021a), we conduct preprocessing steps over each dataset. Specifically, sessions with a length of 1 and items that appeared fewer than 5 times are excluded. Similar to (Wang et al. 2020), we set the sessions of last week (i.e., latest data) as the test data, and the remaining historical data for training. Additionally, we use a session splitting preprocess method to augment session $S = \{s_1, s_2, ..., s_n\}$ in these datasets, and generate multiple sessions with corresponding labels $([s_1, s_2]; s_3), ([s_1, s_2, s_3]; s_4), ..., ([s_1, s_2, ..., s_{n-1}]; s_n)$. The statistics of the datasets are presented in Table 1.

D.2 Computation Resources

The experiments are run on Linux with Intel(R) Xeon(R) Gold 6342 CPU with max CPU speed of 2.80GHz. We implement all the algorithms in this paper using PyTorch. All algorithms are run with a single Nvidia GeForce RTX 3090 GPU.

D.3 SBR Models and Comparison Approaches.

From the perspective of data modeling, we select some well-known SBR models from both sequentail and graphic approaches as the base SBR models:

Dataset	Tmall	RetailRocket	Diginetica
training sessions	351,268	433,643	719,470
test sessions	25,898	15,132	60,858
# of items	40,728	36,968	43,097
average lengths	6.69	5.43	5.12

Table 3: Statistics of datasets used in experiments.

- **STAMP** (Liu et al. 2018) explores the capability of attention layers on session-based recommendation instead of RNNs. It optimizes the attention mechanism of previous work by emphasizing the user's short-term memory.
- **GRU4Rec** (Hidasi et al. 2016) is An RNN based deep learning model for session based recommendation, which utilizes session-parallel mini-batch training process and also employs ranking-based loss functions during the training.
- **SR-GNN** (Wu et al. 2019) employs GNNs to learn item embeddings and fuse the item-level information to get the session representation by leveraging the soft-attention mechanism.
- **GCE-GNN** (Wang et al. 2020) constructs two types of graphs to capture the global and local information from input sessions and combine them to enhance the feature presentations of items.

Note that these are not comparison targets for HID, but rather base models that integrate with HID. Therefore, we have selected the representative high-citated SBR models to demonstrate the generalizability of HID. For comparisons, we select some long-tail approaches which are also plugins:

- **TailNet** (Liu and Zheng 2020a) is the first classical work in SBR to consider recommendation diversity through the preference mechanism to adjust the importance of tail and head items.
- **CSBR** (Chen et al. 2023) addresses the long-tail issue of recommendations with two additional training objectives including the distribution prediction and distribution alignment.
- **LOAM** (Yang et al. 2023) address the long-tail issue of recommendation results through the niche-walk augmentation and the tail session mixup.
- **LAP-SR** (Peng and Zhou 2024) is a post-processing approach that aims to alleviate the long-tail impact in session-based recommender systems by using personalized diversity.

Since HID focuses on addressing the long-tail issue, *existing intent-based SBR models (Chen et al. 2022; Choi et al. 2024b; Zhang et al. 2023; Wang et al. 2024) which only concentrate on the accuracy of recommendations does not serve as the competitors*. Besides, Since MELT (Kim et al. 2023) , GUME (Lin et al. 2024), GALORE (Luo et al. 2023), and LLM-ESR (Liu et al. 2024) utilize collaborative signals from users, which are not available in session-based recommendation due to the anonymity, we have not included them in the competitors either.

Table 4: The accuracy and long-tail comparison of HID and HID (w/o attr.) which replace attributes of items by semantic clusters.

Datasets		Tmall						Diginetica						Retailrocket					
Metrics		Accuracy		Long-tail				Accuracy		Long-tail				Accuracy		Long-tail			
SBR Models	Comparisons	HR	MRR	tHR	tMRR	tCov	Tail	HR	MRR	tHR	tMRR	tCov	Tail	HR	MRR	tHR	tMRR	tCov	Tail
STAMP	+ HID	**28.26**	**15.84**	**28.35**	**15.93**	73.65	78.19	**50.39**	**17.58**	**48.09**	**17.28**	93.05	69.24	**52.38**	27.99	**52.09**	28.34	56.02	72.59
	+ HID (w/o attr.)	28.03	15.60	28.08	15.76	**73.71**	**78.29**	50.12	17.44	47.79	17.13	**93.17**	**69.31**	52.24	**28.09**	51.97	**28.42**	**56.38**	**72.71**
SRGNN	+ HID	**28.38**	**14.66**	**28.13**	**14.50**	66.40	78.12	**52.09**	**18.26**	49.79	**17.25**	96.22	70.05	**53.45**	29.47	**52.61**	29.51	**55.75**	**73.54**
	+ HID (w/o attr.)	28.22	14.51	28.00	14.36	**66.71**	**78.31**	52.01	18.19	**49.82**	17.18	**96.39**	**70.18**	53.33	**29.55**	52.46	**29.59**	55.68	73.48

D.4 Metrics.

To evaluate the recommendation accuracy and long-tail performance, we employ three widely used accuracy metrics, which are HR@K, and MRR@K. Following previous works on long-tail issue (Abdollahpouri, Burke, and Mobasher 2019; Liu and Zheng 2020a; Yang et al. 2023), we introduce some long-tail metrics which are tHR@K, tMRR@K, tCov@K, and Tail@K. tNDCG@K, HRt@K, and MRRt@K calculate the Normalized Discounted Cumulative Gain, Hit Ratio, and Mean Reciprocal Rank of sessions whose next-item (i.e, ground turth item) belongs to the tail items. For the other two long-tail metrics, we give clear definitions as follows:

tCov@K (Tail Coverage) (Liu and Zheng 2020a; Yang et al. 2023) measures how many different tail items ever appear in the top-K recommendations, which can be formulated as: $tCov@K = \frac{|\cup_{u \in U} L_K^T(u)|}{|V|}$, where $L_K^T(u)$ is the set of long tail items within the top-K recommendations of session u.

Tail@K (Liu and Zheng 2020a; Yang et al. 2023) measures how many long-tail items in the top-K for each recommendation list. This metric can be formulated as: $Tail@K = \frac{1}{|U|} \sum_{u \in U} \frac{|L_K^T(u)|}{K}$, where U is the set of sessions.

D.5 Implementation Details

For general settings, the embedding size is 100, the batch size is 256 for Tmall and Diginetica. The scale parameter ϵ is set to 0.2, while the temperature coefficient σ is set to 0.14. Additionally, the penalty threshold η is set to 0.2 and the penalty scale λ_p is set to 0.3. For the hybrid intent learning, the number of clusters n of the spectral clustering is set to 300. About the training process, we adopt the Adam optimizer and set the initial learning rate and L_2 regularization to be 0.001 and 10^{-5}, respectively, and utilize a StepLR scheduler whose decay rate is 0.6 for each epoch to schedule the learning rate. Considering the training epoch, we set the maximum number of epochs to 20, and stopped training when the model did not show any performance improvement after 3 epochs.

All the parameters are initialized by sampling from a Gaussian distribution. Apart from the above settings, we adopt the best hyperparameters reported in the original papers for all SBR models and comparison methods.

D.6 Replacing Attribute With Semantic Clusters

To enhance the applicability of HID across more scenarios, we design experiments to investigate the performance of HID that does not rely on the real attributes of items. Specifically, we replaced the attributes (which depend on additional information) in the original two-stage process with the results of semantic clustering on item embeddings as the preliminary intent, thereby forming another two-stage clustering method (semantic clustering + spectral clustering) to eliminate reliance on attribute labels.

However, this design renders the pre-storage of hybrid intent infeasible under the original approach, as updates to embeddings alter the mapping between items and preliminary intents—consequently affecting the input to spectral clustering. To address this, we first perform spectral clustering with items as nodes to pre-establish the mapping from items to spectral clusters. During training, we compute the semantic clustering of items in each epoch to update the mapping from items to preliminary intents (i.e., semantic clusters). Subsequently, we retrieve the mapping from items to topological clusters, replace items with preliminary intents, and finally obtain the embedding of hybrid intent through average pooling.

The accuracy and long-tail performance of HID and HID (w/o attr.) are given in Table 2. Here we set the semantic cluster of HID (w/o attr.) to be as the same as the attribute number of HID on each dataset. The results show that HID and HID (w/o attr.) achieve comparable overall performance, confirming that HID outperforms base models in both accuracy and long-tail metrics regardless of additional attribute availability. This validates the effectiveness and generalizability of ICLoss. Notably, HID (w/o attr.) exhibits superior long-tail performance, which may due to the initial meaningless item embeddings. Preliminary intents derived from semantic clustering enable HID (w/o attr.) to explore more item combinations at the initial training stage, thereby enhancing long-tail performance.