DAAM: Dual Alignment with Adaptive Margins Framework for Popularity-Aware Fair Recommendation

Wenxuan Xie[1], Jian Cao[1,*], Shiyou Qian[1], Ziyi Huang[1]
[1]School of Computer Science, Shanghai Jiao Tong University, Shanghai, China
{xiewenxuan, cao-jian, ziyi.huang}@sjtu.edu.cn, qshiyou@cs.sjtu.edu.cn

***Abstract*—Recommendation systems frequently suffer from severe popularity bias, which significantly constrains the exposure of unpopular items and degrades overall recommendation quality. Existing debiasing methods, primarily based on causal inference or graph contrastive learning, have achieved progress in mitigating bias. However, they predominantly rely on imposing global constraints to suppress the influence of popular items. This paradigm overlooks users' heterogeneous preferences regarding popularity and fails to effectively bridge the representation separation between popular and unpopular items within the embedding space, leaving unpopular items with insufficient semantic learning. To address these challenges, we propose the *Dual Alignment with Adaptive Margins (DAAM)* framework, which reframes popularity as a personalized user preference. *DAAM* optimizes recommendations via a dual alignment mechanism, achieving intent alignment by matching user popularity expectations with item reality, and representation alignment by transferring collaborative signals from popular anchors to enhance the learning of unpopular items. Furthermore, a popularity-based dynamic margin adjustment mechanism is incorporated to adaptively regulate training difficulty. Experiments on three datasets demonstrate that *DAAM* outperforms mainstream baselines in both accuracy and fairness, verifying the rationality and effectiveness of our approach. Our code is available at https://github.com/sjtuxwx/DAAM.**

***Index Terms*—Popularity Bias, Heterogeneous Preferences, Intent Alignment, Representation Alignment, Dynamic Margin Adjustment.**

I. Introduction

Recommendation systems serve as a core technology for addressing information overload [1]–[4], playing a pivotal role in e-commerce [5]–[7] and content distribution platforms [8]–[10]. However, traditional Collaborative Filtering (CF) algorithms [11] heavily rely on historical interaction data, making systems highly susceptible to popularity bias. This leads to excessive recommendations of popular items while neglecting unpopular items [12], damaging personalized user experiences and harming the interests of unpopular item providers [13].

Researchers worldwide have conducted in-depth studies on this issue, proposing various solutions. Preprocessing [14], [15] and post-processing algorithms [16], [17] focus on resampling and re-ranking, respectively. In-processing algorithms, however, directly intervene in model architecture and the training phase, achieving higher performance ceilings and garnering significant attention. Common in-processing algorithms primarily draw from regularization methods [18], [19], causal learning [20], [21], distributionally robust optimization [22], [23], and contrastive learning [24], [25].

In recent years, Graph Contrastive Learning (GCL) has garnered significant attention as a mainstream paradigm for mitigating popularity bias [26], [27]. By leveraging the InfoNCE loss [28] to promote a uniform distribution of item embeddings, GCL aims to improve the recommendation quality for unpopular items. However, this strategy of global uniformization often comes at the expense of personalized matching accuracy. It overlooks the inherent heterogeneity of user preferences, which range from conformist to depopulist tendencies, implying that enforced global uniformity fails to satisfy diverse user needs. Furthermore, existing research [25] indicates that excessive uniformization tends to induce a severe representation separation between popular and unpopular items. This separation impedes the effective transfer of collaborative signals from the head to the tail, leaving unpopular items with insufficient semantic enhancement.

To address the limitations of non-personalized uniformity in GCL, we propose the *Dual Alignment with Adaptive Margins (DAAM)* framework. Unlike global de-biasing strategies, *DAAM* treats popularity as a personalized dimension rather than a bias to be removed. It introduces a dual-alignment framework with adaptive margins, comprising three key modules: (1) **Personalized Popularity Preference (PPP)**: Serving as the first alignment, this module explicitly aligns the user's latent popularity preference with the candidate item's actual popularity. By modeling the divergence in user needs, it ensures that the recommended items precisely match the specific popularity levels the user expects. (2) **Popularity-Aware Bayesian Personalized Ranking (PA-BPR)**: To improve discrimination for unpopular items, we introduce popularity-aware dynamic margins. By incorporating adaptive margins based on item popularity into the loss function, this mechanism dynamically increases the learning difficulty for unpopular positive samples, forcing the model to learn more robust features. (3) **Explicit Representation Alignment (ERA)**: Constituting the second alignment, this module bridges the representation gap between popular and unpopular items. By enforcing alignment between these distinct groups within the same user sequence, it facilitates semantic information transfer from popular items to unpopular items, effectively mitigating

* Corresponding author.

the representation isolation issue.

In summary, the core contributions of this paper are fourfold:

- We propose the PPP module to explicitly model heterogeneous user preferences. By treating popularity as a personalized matching dimension, it ensures a precise alignment between item popularity and user intent.
- We design the PA-BPR loss to dynamically adjust learning difficulty. This mechanism adaptively intensifies the focus on unpopular positive samples, enhancing the model's discriminative capability.
- We introduce the ERA module to align popular and unpopular items within user sequences. This bridges the semantic gap and facilitates information transfer from popular to unpopular items.
- We conduct extensive experiments on real-world datasets. Results show that *DAAM* outperforms other methods by **2.89%** in quality and **6.24%** in fairness, achieving a dual improvement.

II. Related Work

A. Popularity Bias

Popularity bias in recommendation systems refers to the phenomenon where recommendation outcomes deviate from an item's true utility, with traffic significantly skewed toward top-ranked popular items, resulting in vast numbers of unpopular items failing to receive fair exposure [13], [29]. The causes of popularity bias can be examined from both data and algorithmic perspectives. Due to the long-tail effect in observed data [12], interaction omissions are non-random, masking users' genuine interest in unpopular items. Furthermore, mainstream models exhibit bias amplification mechanisms [30], [31], where models disproportionately fit high-frequency features. This reinforces the Matthew effect through feedback loops, trapping recommendation systems in a cycle of homogenization.

B. Strategies for Mitigating Popularity Bias

Prior work has proposed a variety of strategies to mitigate popularity bias, which can be roughly grouped into the following categories.

Reweighting based methods [18], [19], [32] mitigate popularity bias by assigning higher importance weights to unpopular items during training. However, their inherent high variance often hinders a robust trade-off between recommendation fairness and accuracy.

Causal decoupling methods [20], [21], [33] utilize causal modeling to disentangle item popularity into conformity and quality-driven components. By eliminating the impact of conformity, these approaches enhance the exposure of unpopular items. However, their effectiveness heavily relies on the precision of causal assumptions, which can lead to overcorrection and a subsequent loss of recommendation accuracy.

Representation-based methods [24]–[27] attribute recommendation disparities to the semantic separation between popular and unpopular items. Graph Contrastive Learning (GCL) [26], [27] bridges this gap by promoting embedding uniformity, facilitating collaborative signal transfer to unpopular items. However, this global uniformization strategy overlooks the heterogeneity of user popularity preferences. Such rigid constraints can distort item embeddings and amplify noise, ultimately compromising personalized recommendation accuracy and leading to representation isolation.

Existing de-biasing methods struggle to balance effectiveness, accuracy, and stability, often overlooking user popularity preferences. Leveraging GCL, our approach preserves personalized preferences, adaptively tunes training signals via popularity fit, and integrates popularity-aware margins with representation alignment for stable long-tail learning and bias mitigation.

III. Preliminaries

A. Collaborative Filtering

Given a user set $\mathcal{U}$ ($|\mathcal{U}| = M$) and an item set $\mathcal{V}$ ($|\mathcal{V}| = N$), collaborative filtering models preferences through an interaction matrix $\mathbf{Y} \in \{0,1\}^{M\times N}$, where $Y_{u,i} = 1$ indicates an interaction between user u and item i exists, and $Y_{u,i} = 0$ otherwise. The recommendation model $\mathcal{R}$ maps user u and item i to embeddings $\mathbf{p}_u, \mathbf{q}_i \in \mathbb{R}^d$. It predicts scores using the dot product $\tilde{s}_{u,i} = \mathbf{p}_u^\top \mathbf{q}_i$ and generates a Top-k recommendation list based on these scores.

The optimization phase employs Bayesian Personalized Ranking (BPR) loss [34] to ensure positive items i are ranked higher than negative items j:

$$\mathcal{L}_{BPR} = -\sum_{(u,i,j)\in\mathcal{D}} \ln \sigma(\tilde{s}_{u,i} - \tilde{s}_{u,j}), \tag{1}$$

where $\sigma(\cdot)$ is the sigmoid function and the training set $\mathcal{D} = \{(u,i,j) \mid Y_{u,i} = 1, Y_{u,j} = 0\}$ consists of triplets formed by positive item i and negative item j.

B. Graph Contrastive Learning

To mitigate popularity-induced representation degradation, we employ graph contrastive learning [26] to encourage embedding uniformity. Specifically, we generate two augmented graph views via random perturbations and maximize the mutual information between them. The InfoNCE losses [28] for items and users are defined as:

$$\mathcal{L}_{cl}^{item} = \sum_{i\in\mathcal{V}} -\log \frac{\exp(\mathbf{q}_i'^\top \mathbf{q}_i''/\tau)}{\sum_{k\in\mathcal{V}} \exp(\mathbf{q}_i'^\top \mathbf{q}_k''/\tau)}, \tag{2}$$

$$\mathcal{L}_{cl}^{user} = \sum_{u\in\mathcal{U}} -\log \frac{\exp(\mathbf{p}_u'^\top \mathbf{p}_u''/\tau)}{\sum_{k\in\mathcal{U}} \exp(\mathbf{p}_u'^\top \mathbf{p}_k''/\tau)}, \tag{3}$$

where $\mathbf{q}_i', \mathbf{q}_i''$ and $\mathbf{p}_u', \mathbf{p}_u''$ denote the embeddings of item i and user u generated from the two distinct perturbed views, respectively, and τ is the temperature hyperparameter. The total self-supervised loss is $\mathcal{L}_{cl} = \mathcal{L}_{cl}^{item} + \mathcal{L}_{cl}^{user}$. However, strictly pursuing uniformity may inadvertently widen the semantic gap between popular and unpopular items, hindering collaborative signal transfer and leaving tail items under-optimized.

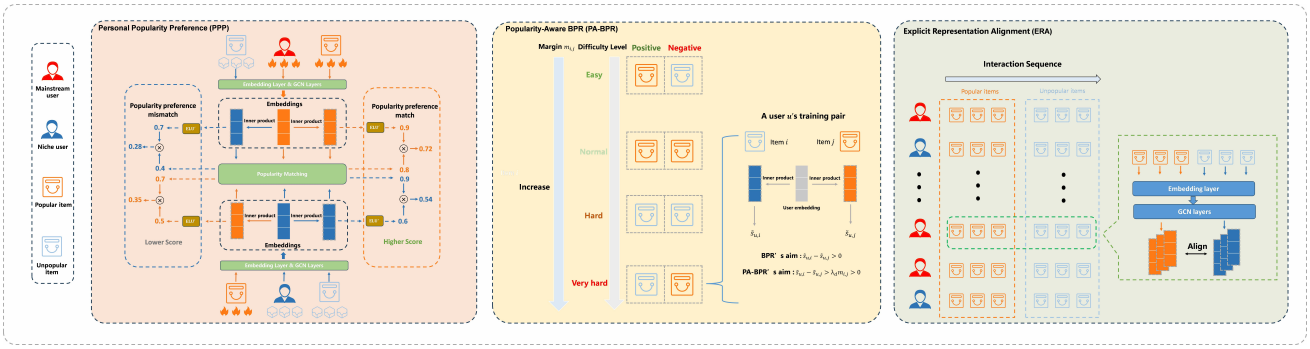


Fig. 1. An Illustration of our proposed *DAAM* framework, comprising the PPP, PA-BPR, and ERA modules. The PPP module achieves intent-level alignment by matching user popularity expectations with actual item popularity, explicitly modeling that niche users prefer unpopular items while mainstream users favor popular ones. The PA-BPR module enhances discrimination for hard samples via adaptive difficulty adjustment. Finally, the ERA module bridges the semantic gap by leveraging popular items to facilitate the representation learning of unpopular items.

IV. The Proposed Model

To mitigate popularity bias in recommendation systems, we propose a popularity matching training algorithm based on graph contrastive learning. As shown in Fig. 1, our model consists of three components: PPP, PA-BPR and ERA. Specifically, PPP and ERA achieve complementary intent and representation alignment, respectively, while PA-BPR coordinates them by adaptively modulating training difficulty for tail-item learning.

A. Personalized Popularity Preference

From the user's perspective, we model popularity preference as a latent user attribute. To align user u with their popularity preference intent, we define a learnable transformation matrix $\mathbf{W} \in \mathbb{R}^{d\times d}$ and a learnable user-specific global bias b_u. Formally, given a training triplet (u, i, j), where i is the positive item and j is the negative item, the user u's ideal popularity score $\beta_{u,i}$ for item i is predicted as:

$$\beta_{u,i} = \sigma(\mathbf{p}_u^\top \mathbf{W}\mathbf{q}_i + b_u), \tag{4}$$

where $\mathbf{p}_u$ and $\mathbf{q}_i$ denote the embeddings of user u and item i, respectively.

Simultaneously, we compute the global popularity statistics. Let $\mathcal{V}$ denote the set of all items in the dataset. We define the global count set as $\mathcal{C} = \{c_i \mid i \in \mathcal{V}\}$, where c_i represents the occurrence count of item i. Then, for the specific positive item i, we perform smoothing normalization to yield its smoothed popularity h_i:

$$h_i = \left(\frac{c_i}{\max(\mathcal{C})}\right)^\gamma, \tag{5}$$

where $\gamma \in [0, 1]$ is the smoothing factor.

To measure the conformity between user u's ideal preference and item i's actual popularity, we employ a Gaussian kernel. The matching score between user u and item i, denoted as $\delta_{u,i}$, is calculated as:

$$\delta_{u,i} = \exp\left(-\frac{(\beta_{u,i} - h_i)^2}{2\rho^2}\right), \tag{6}$$

where ρ is the bandwidth parameter controlling the kernel sharpness. A higher $\delta_{u,i}$ indicates that the item's popularity aligns closely with the user's expectation.

We postulate that high popularity conformity should enhance the interaction probability. Therefore, we calibrate the user's preference score using this conformity weight. However, since the original predicted score $\tilde{s}_{u,i}$ can be negative, directly multiplying it by $\delta_{u,i}$ may cause ranking inversion. To address this, we use a modified activation function $\mathrm{ELU}'(\cdot)$ to map the raw score to a strictly positive domain before weighting [35]. The calibrated score $s_{u,i}$ is defined as:

$$s_{u,i} = \delta_{u,i} \cdot \mathrm{ELU}'(\tilde{s}_{u,i}), \tag{7}$$

where the scalar function $\mathrm{ELU}'(\cdot)$ is defined as:

$$\mathrm{ELU}'(\tilde{s}_{u,i}) = \begin{cases} \tilde{s}_{u,i} + 1 & \text{if } \tilde{s}_{u,i} \geq 0 \\ \exp(\tilde{s}_{u,i}) & \text{if } \tilde{s}_{u,i} < 0 \end{cases}. \tag{8}$$

Analogously, we compute the calibrated score $s_{u,j}$ for the negative item j following the identical procedure. Finally, the popularity matching loss is formulated based on the calibrated preference margin between the positive and negative items:

$$\mathcal{L}_{PM} = -\sum_{(u,i,j)\in\mathcal{D}} \ln \sigma\left(s_{u,i} - s_{u,j}\right). \tag{9}$$

B. Popularity-Aware Bayesian Personalized Ranking

We categorize the training item pairs for a user u into four distinct difficulty levels based on the actual popularity of the positive item i and the negative item j:

- *Easy*: i is popular and j is unpopular. Ranking i higher aligns with the model's inherent popularity bias, making discrimination trivial.
- *Normal*: Both i and j are popular. With abundant interaction data, the model can easily learn fine-grained preferences to distinguish them.
- *Hard*: Both i and j are unpopular. Despite similar popularity levels, data sparsity for unpopular items makes optimization challenging.
- *Very Hard*: i is unpopular while j is popular. The model must overcome strong popularity bias to correctly rank

the unpopular item i over the popular item j. This combination serves as the key training samples for alleviating the poor recommendation quality of unpopular items.

Based on this difficulty hierarchy, we formally define the popularity discrimination difficulty for a training triplet (u, i, j) as the logarithmic ratio between the negative and positive item popularity:

$$m_{i,j} = \ln\left(\frac{h_j}{h_i}\right), \tag{10}$$

where h_i and h_j are the smoothed popularity of items i and j. Intuitively, $m_{i,j}$ acts as a popularity adaptive margin. A larger value (i.e., $h_j \gg h_i$) corresponds to the *Very Hard* scenario, where the model faces a strong popularity bias from the negative item. Conversely, a smaller value implies an easier task where the popularity signal aids the ranking.

Based on this analysis, we incorporate this difficulty indicator into the standard BPR objective. The resulting Popularity-Aware BPR (PA-BPR) loss is defined as:

$$\mathcal{L}_{PA} = -\sum_{(u,i,j)\in\mathcal{D}} \ln\sigma\left(\tilde{s}_{u,i} - \tilde{s}_{u,j} - \lambda_d \cdot m_{i,j}\right), \tag{11}$$

where λ_d is a hyperparameter scaling the margin influence. By subtracting a dynamic margin, the model is forced to generate a larger preference gap for harder pairs (where $m_{i,j}$ is large) to minimize the loss.

C. Explicit Representation Alignment

The long-tail distribution hinders the representation learning of unpopular items, causing a semantic separation from popular ones. To bridge this, we leverage popular items in user sequences as soft supervision signals to align tail items

Formally, let $\mathcal{V}_u = \{i \in \mathcal{V} \mid Y_{u,i} = 1\}$ denote the positive interaction set of user u, where $\mathcal{V}$ is the item universe. We partition $\mathcal{V}_u$ into two disjoint subsets based on global frequency: the popular set $\mathcal{V}_u^{pop}$ and the unpopular set $\mathcal{V}_u^{unpop}$. This strictly ensures that any item in $\mathcal{V}_u^{pop}$ has higher popularity than those in $\mathcal{V}_u^{unpop}$. Given diverse user interests, blind alignment risks introducing noise. Thus, we design a weighted alignment loss incorporating similarity perception and popularity penalty. The Explicit Representation Alignment (ERA) loss $\mathcal{L}_{ERA}$ is defined as:

$$\mathcal{L}_{ERA} = \sum_{u\in\mathcal{U}} \sum_{i_h\in\mathcal{V}_u^{pop}, i_t\in\mathcal{V}_u^{unpop}} w_{i_h,i_t} \|\mathbf{q}_{i_h} - \mathbf{q}_{i_t}\|_2^2, \tag{12}$$

where i_h and i_t represent a popular item and an unpopular item, respectively. The weight coefficient w_{i_h,i_t} is calculated as:

$$w_{i_h,i_t} = \frac{\max(0, \mathbf{q}_{i_h}^\top \mathbf{q}_{i_t})}{\ln(c_{i_h}+1)\cdot\ln(c_{i_t}+1)}. \tag{13}$$

The numerator utilizes the dot product to filter out negatively correlated noise, ensuring that alignment only occurs between semantically similar items. The denominator introduces logarithmic popularity to suppress the dominant influence of highly popular items, preventing the representation of tail items from being overwhelmed by popular items.

D. Model Optimization

To achieve a balanced optimization across all modules, we define the total objective function as a weighted combination of the aforementioned losses:

$$\mathcal{L} = \mathcal{L}_{PA} + \lambda_1\mathcal{L}_{PM} + \lambda_2\mathcal{L}_{ERA} + \lambda_3\mathcal{L}_{cl} \tag{14}$$

where λ_1 and λ_2 are tunable hyperparameters, while λ_3 is a predefined weight to incorporate the graph contrastive learning signal.

V. Experiments

In this section, we conduct experiments and evaluations on *DAAM*, aiming to address the following research questions:

- **RQ1:** How does *DAAM* perform compared to existing bias mitigation methods?
- **RQ2:** How do the key components of *DAAM* perform?
- **RQ3:** How do different hyperparameters affect *DAAM*'s performance?
- **RQ4:** Why is *DAAM* effective?

A. Experimental Setup

1) Dataset: For this experiment, we selected the Yelp2018[1], MovieLens-1M[2] and BookCrossing[3] datasets. These datasets exhibit long-tail distributions, making fixed-ratio data splitting inadequate for evaluating bias mitigation capabilities. Therefore, drawing from prior work [20], we retain a fixed number of interactions for each item in the test set, accounting for approximately 10% of the entire dataset, while ensuring that items in the test set are present in the training set.

2) Baselines: To better evaluate the effectiveness of our proposed method, we selected LightGCN [36] as the base network and experimented with mainstream bias mitigation methods for comparison. These include: Re-weighting: IPS [18] and IPS-CN [32], Causal disentanglement: MACR [20], DICE [33], Distribution-robust optimization: DRO [23], Graph contrastive learning: SimGCL [26] and PAAC [25].

3) Metrics and common hyperparameters' settings: We conduct full-scale product evaluation on the test set, using Recall, NDCG, and HR [32] to assess recommendation system quality, and employing the Gini coefficient as an equity evaluation metric [37]. Unless otherwise specified, we maintain $\lambda_3 = 0.2$ and $\tau = 0.2$ as default hyperparameter settings across all datasets. Specifically, for the Yelp2018 dataset, γ and ρ are both set to 0.8. For MovieLens-1M and BookCrossing, these parameters are adjusted to $\gamma = 0.2$ and $\rho = 0.2$, respectively.

B. Overall Performance (RQ1)

As shown in Table I, we compared the performance of our proposed *DAAM* model against multiple baseline models across three datasets and documented the experiments;

[1]https://www.yelp.com/dataset

[2]https://grouplens.org/datasets/movielens/1m

[3]https://www.kaggle.com/datasets/somnambwl/bookcrossing-dataset

TABLE I

OVERALL PERFORMANCE OF THE PROPOSED *DAAM* AND BASELINE MODELS ON ALL DATASETS. THE SECOND-BEST RESULTS ARE UNDERLINED. NUMBERS MARKED WITH * DENOTE THAT THE IMPROVEMENT IS STATISTICALLY SIGNIFICANT COMPARED WITH THE BEST BASELINE, DETERMINED BY A T-TEST WITH A p-VALUE < 0.05.

Dataset	Metrics	LightGCN	IPS	IPS-CN	DICE	MACR	DRO	SimGCL	PAAC	*DAAM*
Yelp2018	HR@15↑	0.0091	0.0135	0.0144	0.0083	0.0153	0.0097	0.0444	<u>0.0490</u>	**0.0507***
	Recall@15↑	0.0023	0.0052	0.0059	0.0017	0.0075	0.0027	0.0353	<u>0.0396</u>	**0.0409***
	NDCG@15↑	0.0087	0.0121	0.0128	0.0080	0.0129	0.0091	0.0321	<u>0.0351</u>	**0.0365***
	Gini↓	0.9063	0.8337	0.8143	0.9318	0.7946	0.9135	0.7597	<u>0.7510</u>	**0.7219***
MovieLens-1M	HR@15↑	0.0946	0.0159	0.1113	0.1215	0.1157	0.0148	0.1288	<u>0.1339</u>	**0.1391***
	Recall@15↑	0.0933	0.0156	0.1097	0.1201	0.1141	0.0146	0.1273	<u>0.1323</u>	**0.1377***
	NDCG@15↑	0.0583	0.0096	0.0675	0.0755	0.0702	0.0080	0.0770	<u>0.0801</u>	**0.0826***
	Gini↓	0.8153	0.9331	0.7140	0.7606	<u>0.6705</u>	0.9617	0.6782	0.6789	**0.5929***
BookCrossing	HR@15↑	0.0296	0.0086	0.0505	0.0810	0.0864	0.0479	0.0945	<u>0.1053</u>	**0.1065***
	Recall@15↑	0.0288	0.0083	0.0491	0.0793	0.0848	0.0467	0.0923	<u>0.1030</u>	**0.1044***
	NDCG@15↑	0.0197	0.0054	0.0387	0.0603	0.0638	0.0328	0.0740	<u>0.0845</u>	**0.0860***
	Gini↓	0.9305	<u>0.5577</u>	0.8268	0.8627	0.8216	0.9315	0.7325	0.5591	**0.5394***

- *DAAM* outperformed all comparison algorithms across all metrics on the three datasets, specifically achieving average improvements of 3.56%, 3.7%, and 1.4% on Yelp2018, MovieLens-1M, and BookCrossing respectively, while also achieving lower Gini coefficients than all comparison algorithms.
- IPS and DRO algorithms exhibited significant instability due to sensitivity to sparse data update gradients. IPS-CN mitigated large gradient shocks from unpopular items via truncation and normalization, achieving relatively stable performance, though still lagging behind the causal decoupling-based MACR.
- Among all comparison methods, SimGCL and PAAC demonstrated strong competitiveness. However, *DAAM* still outperformed these robust baselines. Specifically on MovieLens-1M, *DAAM* achieved a Gini coefficient approximately 12.6% lower than SimGCL and PAAC, while maintaining a roughly 3.7% lead in recommendation effectiveness and significantly outperforming other methods. This further validates that *DAAM* enhances recommendation fairness while preserving substantial recommendation performance.

C. Ablation Study (RQ2)

To better illustrate the influence of each component within the *DAAM* model, we conducted ablation experiments. The results are shown in Fig. 2a, where w/o P, w/o A, and w/o E represent three variants of *DAAM* with PPP, PA-BPR, and ERA modules removed, respectively. We observe that:

- The w/o P variant exhibits the most significant decline in recommendation quality, decreasing by approximately 19.7% while increasing the Gini coefficient by 3.7%. This indicates that popularity matching substantially enhances overall recommendation quality while maintaining fairness.
- The w/o A variant suffers the greatest loss in fairness, with its Gini coefficient increasing by approximately 6.8%, and reduced recommendation quality by about 4.0%. This indicates that enabling the recommendation system to learn more robust representations of less popular items helps improve overall fairness and recommendation quality.
- The w/o E variant, both recommendation quality and fairness are inferior to *DAAM*, demonstrating that the alignment module also contributes to enhancing overall recommendation quality while maintaining fairness.

D. Hyperparameter Sensitivities (RQ3)

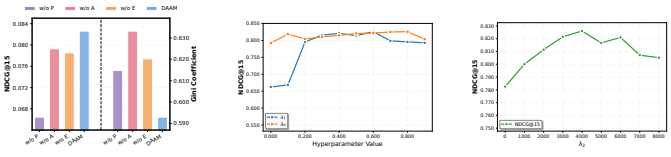
Next, we will investigate how key hyperparameters affect the performance of the *DAAM* model. The results are shown in Fig. 2b and Fig. 2c, revealing that:

- The model performs best when λ_1 is 0.6. When λ_1 is too small, the model struggles to capture users' popularity preferences. When λ_1 is too large, it tends to degrade the recommendation quality of non-mainstream items, thereby reducing overall performance.
- When λ_d is relatively small, its increase enhances the model's attention toward unpopular positive items. However, an excessively large λ_d may cause the model to gradually reduce its focus on popular positive samples, leading to a decline in overall recommendation quality.
- As λ_2 increases, model performance first improves then declines. This occurs because a lower λ_2 reduces the distinction between positive and negative samples, while a higher λ_2 excessively aligns the representations of both groups, unduly affecting their original unique characteristics.

E. Popularity Bias Mitigation (RQ4)

In this subsection, we evaluate *DAAM*'s effectiveness in mitigating popularity bias and analyze the underlying reasons for its superior performance.

First, we classify the top 20% of items by interaction frequency as popular items, designating the remainder as



(a) Ablation study. (b) Effects of λ_1,λ_d. (c) Effect of λ_2.

Fig. 2. Impact of different modules and hyperparameters on *DAAM* performance.

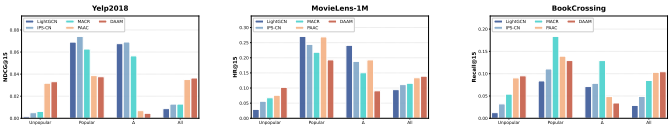
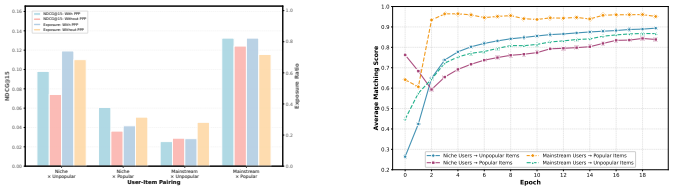
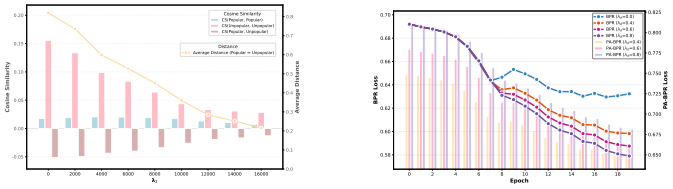


Fig. 3. Item group performance, where Δ denotes the performance gap between popular and unpopular items and All represents the overall performance on the entire dataset.



(a) Group-wise PPP performance. (b) Average matching score evolution.



(c) ERA module's impact. (d) Impact of λ_d on loss.

Fig. 4. Detailed analysis of module effectiveness.

unpopular. The performance comparison based on this classification is shown in Fig. 3. Specifically, *DAAM* significantly narrows the performance gap between popular and unpopular items, while also enhancing overall recommendation quality.

To verify *DAAM*'s capability in capturing polarized user preferences, we categorize users based on the average popularity of their historically interacted items. Specifically, users ranking in the top 20% of this metric are defined as mainstream users, while those in the bottom 20% are defined as niche users. As shown in Fig. 4a, PPP implements a differentiated recommendation strategy. Remarkably, the PPP module achieves a comprehensive optimization: it simultaneously enhances both the exposure ratio and ranking quality for unpopular items among niche users as well as popular items among mainstream users, thereby effectively satisfying the specific interests of both groups. Particularly for niche users, while the frequency of popular item recommendations decreases, the quality of these recommendations improves, indicating that the PPP module effectively filters out irrelevant popular noise. Additionally, Fig. 4b tracks the evolution of average matching scores. Crucially, the score for mainstream users and popular items remains dominant throughout, demonstrating that PPP preserves genuine interests without oversuppressing popular items. Simultaneously, for niche users, the match with unpopular items ascends remarkably, eventually surpassing popular items. This dual trend confirms PPP's precise personalized alignment: effectively correcting bias for niche users while maintaining high-quality recommendations for mainstream users.

As illustrated in Fig. 4c, increasing λ_2 leads to a rapid decrease in the cosine similarity within unpopular items. This indicates that the ERA module effectively enhances the discriminability of unpopular items, thereby mitigating representation collapse. Simultaneously, the cosine similarity between popular and unpopular items increases from negative values towards zero, while their average Euclidean distance significantly decreases. This dual evidence signifies that the model successfully breaks representation isolation by mitigating semantic repulsion and promoting alignment in the latent space. However, an excessively large λ_2 draws these representations indistinguishably close, leading to over-smoothing. Consequently, the unique embedding patterns and intrinsic semantics of popular and unpopular items are erased. This loss of discriminative features impairs the ability of the model to capture fine-grained item characteristics, thereby reducing overall recommendation quality.

As illustrated in Fig. 4d, we visualize training dynamics where bar charts display PA-BPR loss for unpopular-positive/popular-negative pairs. Results indicate that increasing λ_d amplifies this loss, reflecting an enlarged discrimination margin that compels the model to strictly optimize these hard pairs. Consequently, the original BPR loss decreases by approximately 10% under this margin-enhanced regime. This confirms that PA-BPR drives the model to capture strong semantic information inherent in unpopular positive samples, thereby significantly enhancing the discriminability of these challenging pairs.

VI. Conclusion

This study proposes the *DAAM* framework to address the problem of popularity bias. Unlike traditional methods that overlook variations in user preferences, *DAAM* treats popularity as a personalized matching dimension, leveraging the PPP module to model the alignment between a user's ideal preference and an item's actual popularity. By integrating PA-BPR loss with the ERA mechanism, the model significantly enhances long-tail discovery capabilities while retaining effective information from popular items. Experimental results demonstrate that *DAAM* outperforms existing state-of-the-art methods in both accuracy and fairness, achieving an effective balance between the two.

Although these improvements are based on offline observation patterns, we recognize that offline ground truth remains inherently influenced by historical exposure. Future work will

further explore the temporal evolution characteristics of popularity and attempt to extend the framework to more complex scenarios such as sequential recommendation.

REFERENCES

[1] Z. Hu, S. Nakagawa, Y. Zhuang, J. Deng, S. Cai, T. Zhou, and F. Ren, "Hierarchical denoising for robust social recommendation," *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[2] Y. Yang, L. Wu, Z. Wang, Z. He, R. Hong, and M. Wang, "Graph bottlenecked social recommendation," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024, pp. 3853–3862.

[3] X. Wang, L. Wu, L. Hong, H. Liu, and Y. Fu, "Llm-enhanced user–item interactions: Leveraging edge information for optimized recommendations," *ACM Transactions on Intelligent Systems and Technology*, vol. 16, no. 5, pp. 1–24, 2025.

[4] Y. Liao, Y. Yang, M. Hou, L. Wu, H. Xu, and H. Liu, "Mitigating distribution shifts in sequential recommendation: An invariance perspective," in *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2025, pp. 1603–1613.

[5] Y. Chen, Q.-T. Truong, X. Shen, J. Li, and I. King, "Shopping trajectory representation learning with pre-training for e-commerce customer understanding and recommendation," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024, pp. 385–396.

[6] S. Tan, D. Li, R. Jiang, Z. Wang, X. Yu, and M. Okumura, "Taming recommendation bias with causal intervention on evolving personal popularity," in *Proceedings of the 31st ACM SIGKDD conference on knowledge discovery and data mining v. 2*, 2025, pp. 2790–2800.

[7] Y. Li, X. Zhang, L. Luo, H. Chang, Y. Ren, I. King, and J. Li, "G-refer: Graph retrieval-augmented large language model for explainable recommendation," in *Proceedings of the ACM on Web Conference 2025*, 2025, pp. 240–251.

[8] H. Bai, L. Wu, M. Hou, M. Cai, Z. He, Y. Zhou, R. Hong, and M. Wang, "Multimodality invariant learning for multimedia-based new item recommendation," in *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2024, pp. 677–686.

[9] Y. Yang, L. Wu, Y. Liao, Z. He, P. Shao, R. Hong, and M. Wang, "Invariance matters: Empowering social recommendation via graph invariant learning," in *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2025, pp. 2038–2047.

[10] K. Bachiri, A. Yahyaouy, M. Malek, and N. Rogovschi, "Multimodal graph learning and sequential modeling for video recommendation," in *2025 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2025, pp. 1–8.

[11] L. Wu, X. He, X. Wang, K. Zhang, and M. Wang, "A survey on accuracy-oriented neural recommendation: From collaborative filtering to information-rich recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 5, pp. 4425–4445, 2022.

[12] J. Chen, H. Dong, X. Wang, F. Feng, M. Wang, and X. He, "Bias and debias in recommender system: A survey and future directions," *ACM Transactions on Information Systems*, vol. 41, no. 3, pp. 1–39, 2023.

[13] A. Klimashevskaia, D. Jannach, M. Elahi, and C. Trattner, "A survey on popularity bias in recommender systems," *User Modeling and User-Adapted Interaction*, vol. 34, no. 5, pp. 1777–1834, 2024.

[14] X. Chen, W. Fan, J. Chen, H. Liu, Z. Liu, Z. Zhang, and Q. Li, "Fairly adaptive negative sampling for recommendations," in *Proceedings of the ACM Web Conference 2023*, 2023, pp. 3723–3733.

[15] Y. Li, H. Chen, S. Xu, Y. Ge, J. Tan, S. Liu, and Y. Zhang, "Fairness in recommendation: Foundations, methods, and applications," *ACM Transactions on Intelligent Systems and Technology*, vol. 14, no. 5, pp. 1–48, 2023.

[16] Z. Xiang, H. Zhao, C. Zhao, M. He, and J. Fan, "Performative debias with fair-exposure optimization driven by strategic agents in recommender systems," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024, pp. 3507–3517.

[17] C. Xu, S. Chen, J. Xu, W. Shen, X. Zhang, G. Wang, and Z. Dong, "P-mmf: Provider max-min fairness re-ranking in recommender system," in *Proceedings of the ACM Web Conference 2023*, 2023, pp. 3701–3711.

[18] T. Schnabel, A. Swaminathan, A. Singh, N. Chandak, and T. Joachims, "Recommendations as treatments: Debiasing learning and evaluation," in *international conference on machine learning*. PMLR, 2016, pp. 1670–1679.

[19] W. X. Zhao, J. Chen, P. Wang, Q. Gu, and J.-R. Wen, "Revisiting alternative experimental settings for evaluating top-n item recommendation algorithms," in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 2020, pp. 2329–2332.

[20] T. Wei, F. Feng, J. Chen, Z. Wu, J. Yi, and X. He, "Model-agnostic counterfactual reasoning for eliminating popularity bias in recommender system," in *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 2021, pp. 1791–1800.

[21] A. Zhang, J. Zheng, X. Wang, Y. Yuan, and T.-S. Chua, "Invariant collaborative filtering to popularity distribution shift," in *Proceedings of the ACM Web Conference 2023*, 2023, pp. 1240–1251.

[22] S. Sagawa, P. W. Koh, T. B. Hashimoto, and P. Liang, "Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization," *arXiv preprint arXiv:1911.08731*, 2019.

[23] J. Duchi and H. Namkoong, "Variance-based regularization with convex objectives," *Journal of Machine Learning Research*, vol. 20, no. 68, pp. 1–55, 2019.

[24] J. Chen, J. Wu, J. Wu, X. Cao, S. Zhou, and X. He, "Adap-τ: Adaptively modulating embedding magnitude for recommendation," in *Proceedings of the ACM Web Conference 2023*, 2023, pp. 1085–1096.

[25] M. Cai, L. Chen, Y. Wang, H. Bai, P. Sun, L. Wu, M. Zhang, and M. Wang, "Popularity-aware alignment and contrast for mitigating popularity bias," in *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024, pp. 187–198.

[26] J. Yu, H. Yin, X. Xia, T. Chen, L. Cui, and Q. V. H. Nguyen, "Are graph augmentations necessary? simple graph contrastive learning for recommendation," in *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, 2022, pp. 1294–1303.

[27] J. Yu, X. Xia, T. Chen, L. Cui, N. Q. V. Hung, and H. Yin, "Xsimgcl: Towards extremely simple graph contrastive learning for recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 36, no. 2, pp. 913–926, 2023.

[28] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[29] H. Abdollahpouri, M. Mansoury, R. Burke, and B. Mobasher, "The unfairness of popularity bias in recommendation," *arXiv preprint arXiv:1907.13286*, 2019.

[30] Z. Zhu, Y. He, X. Zhao, Y. Zhang, J. Wang, and J. Caverlee, "Popularity-opportunity bias in collaborative filtering," in *Proceedings of the 14th ACM international conference on web search and data mining*, 2021, pp. 85–93.

[31] C. Gao, Y. Zheng, W. Wang, F. Feng, X. He, and Y. Li, "Causal inference in recommender systems: A survey and future directions," *ACM Transactions on Information Systems*, vol. 42, no. 4, pp. 1–32, 2024.

[32] A. Gruson, P. Chandar, C. Charbuillet, J. McInerney, S. Hansen, D. Tardieu, and B. Carterette, "Offline evaluation to make decisions about playlistrecommendation algorithms," in *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 2019, pp. 420–428.

[33] Y. Zheng, C. Gao, X. Li, X. He, Y. Li, and D. Jin, "Disentangling user interest and conformity for recommendation with causal embedding," in *Proceedings of the web conference 2021*, 2021, pp. 2980–2991.

[34] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "Bpr: Bayesian personalized ranking from implicit feedback," *arXiv preprint arXiv:1205.2618*, 2012.

[35] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *arXiv preprint arXiv:1511.07289*, vol. 4, no. 5, p. 11, 2015.

[36] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and 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.

[37] S. Park, M. Yoon, J.-w. Lee, H. Park, and J. Lee, "Toward a better understanding of loss functions for collaborative filtering," in *Proceedings of the 32nd ACM international conference on information and knowledge management*, 2023, pp. 2034–2043.