Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Adaptive Lagrangian Control of Long-Tail Exposure in Graph Recommendation

Puskar Adhikari

puskar.adhikari@alumni.ait.asia

Asian Institute of Technology

Chantri Polprasert

Asian Institute of Technology

Research Article

Keywords: Graph neural networks, Recommender systems, Item-side exposure fairness, Long-tail exposure, Constrained optimization, Primal–dual optimization

Posted Date: November 13th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11241269/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Adaptive Lagrangian Control of Long-Tail Exposure in Graph Recommendation

Puskar Adhikari[1*] and Chantri Polprasert[1*]

[1*]Faculty of Advanced Science and Technology, Asian Institute of Technology, Khlong Luang, 12120, Pathum Thani, Thailand.

*Corresponding author(s). E-mail(s): puskar.adhikari@alumni.ait.asia; chantri@ait.asia;

Abstract

Graph neural network (GNN) recommenders can amplify popularity bias, limiting long-tail exposure. Existing in-training remedies rely on a penalty weight that must be tuned to achieve a desired exposure level, and the same weight can produce very different exposure levels across datasets. We propose Adaptive Lagrangian Fairness (ALF), a primal–dual controller that makes the desired top-$\boldsymbol{K}$ share of reachable long-tail items the control variable of GNN training: a dual variable driven by the measured validation tail share sets the strength of a separate, interchangeable differentiable surrogate. We instantiate it with a softmax tail-mass proxy (ALF-S) and the pairwise popularity-bias loss PBiLoss (ALF-P). On MovieLens-100K, MovieLens-1M, and Yelp2018 with a LightGCN backbone, fixed weights transfer poorly: the same softmax-penalty weight yields tail exposure differing up to elevenfold across datasets, and the same PBiLoss weight up to 2.6-fold. ALF instead attains the requested share closely, with median errors of **0.002**–**0.004** for ALF-S on MovieLens-1M and Yelp2018 and for ALF-P on MovieLens-100K within its range. Against their static counterparts over shared exposure ranges, the ALF variants have higher common-range hypervolume in five of six comparisons (up to **13%**), with 95% bootstrap intervals excluding zero in four. The surrogates differ in range: ALF-P saturates near tail shares of **0.55**, **0.59**, and **0.21**, whereas ALF-S remains responsive across the full tested range. Cosine scoring also raises unconstrained long-tail exposure over inner-product scoring on all three datasets.

Keywords: Graph neural networks, Recommender systems, Item-side exposure fairness, Long-tail exposure, Constrained optimization, Primal–dual optimization

1 Introduction

Graph neural network (GNN) recommenders such as LightGCN [18], Neural Graph Collaborative Filtering (NGCF) [36], and GraphSAGE [16] propagate collaborative signals over user–item graphs. This propagation can amplify popularity bias: heavily interacted items receive stronger representations than sparsely interacted long-tail items [7, 47], so head items dominate top-K lists and long-tail exposure shrinks [1, 24].

Post-hoc reranking, such as explicit query aspect diversification (xQuAD) [2, 29], FA*IR [45], and calibrated recommendation [32], needs no retraining but is bounded by the base model's candidates and scores. **In-training regularization** [1, 19, 37] changes the learned representations through a penalty weight, but the resulting exposure depends strongly on that weight and must be searched again whenever the dataset or desired operating point changes. With the weight fixed at $\lambda = 0.15$, the static regularizer we evaluate attains a tail share of 0.07, 0.30, and 0.73 on MovieLens-100K, MovieLens-1M, and Yelp2018 (Fig. 1a). The pairwise PBiLoss [27] is more consistent across the MovieLens catalogues, but $w = 2$ yields 0.52, 0.51, and only 0.20 on Yelp2018 (Fig. 1b).

Related lines of work each cover part of what is needed. FairCo [26] uses feedback to steer exposure through the ranking policy, whereas ALF feeds measured exposure back into the training objective of the GNN. FairDual [40] adapts fairness pressure during training, but for group-level objectives. Differentiable top-K fairness methods [46] make top-K exposure trainable but expose a tuned fairness–utility trade-off parameter rather than a target-conditioned controller. We are not aware of prior work that combines end-to-end GNN training with an explicit top-K long-tail exposure target and dual feedback from the measured exposure.

We make the exposure target the control variable of training. The practitioner specifies a target such as "tail exposure of at least 0.10," and a dual variable driven by violations of that target sets the strength of the training intervention. The key design choice is a *metric–surrogate separation*: the dual update uses the discrete top-K exposure measured on validation users, which need not be differentiable, while a separate differentiable surrogate supplies the gradient. The surrogate determines how the model can move; the measured exposure determines where it should move. We call the controller **Adaptive Lagrangian Fairness (ALF)**. We do not claim novelty in Lagrangian optimization; the contribution is using dual feedback to turn a discrete exposure target into the operating parameter of GNN training, with the surrogate interchangeable. We evaluate two instances: **ALF-S**, with a softmax tail-mass proxy, and **ALF-P**, with the pairwise PopPos term of PBiLoss. Item-side long-tail exposure is one operational fairness objective, not a general fairness criterion.

We also control a confound: inner-product scoring lets degree-correlated embedding norms concentrate popularity [9, 15], and switching to cosine scoring raises unconstrained tail exposure on all three datasets (§5.1). We therefore use cosine-scored BASE as a stricter reference.

The main contributions are:

- **Target-conditioned exposure control.** The practitioner specifies the desired top-K long-tail exposure of a GNN recommender; the strength of the intervention is

set from measured violations, with controller settings shared across datasets apart from surrogate-specific scaling and calibration.

- **Metric–surrogate separation.** Discrete validation TailExp drives the dual feedback and a separate differentiable surrogate drives the model; the same controller operates with a softmax proxy (ALF-S) and the PopPos term of PBiLoss (ALF-P).
- **Matched evaluation.** On MovieLens-100K, MovieLens-1M, and Yelp2018 we compare each instance with a weight sweep of its own surrogate and with post-hoc reranking, reporting target attainment, reachable range, and range-matched frontiers. A supporting finding is that the scoring convention changes baseline exposure and should be reported.

2 Related Work

Graph recommenders and popularity.

NGCF [36] introduced graph convolution for collaborative filtering and LightGCN [18] reduced it to neighborhood aggregation; later work added self-supervision (Self-supervised Graph Learning, SGL [38]), knowledge graphs (Knowledge Graph Attention Network, KGAT [35]), and inductive aggregation (GraphSAGE [16]). Propagation can amplify popularity bias [7, 47, 49], and interventions usually modify the objective or the final ranking rather than the encoder. ALF modifies the objective and is encoder-agnostic.

Post-hoc reranking.

Item-side fairness concerns how exposure is distributed across items or groups [6, 7], in absolute terms or relative to relevance [30]. xQuAD [29] was adapted to popularity bias by Abdollahpouri et al. [2]; FA*IR [45] imposes ranking constraints, calibrated recommendation [32] matches a target distribution, personalized xQuAD [22] adds user-specific preferences, and Marras et al. [25] rerank towards operator-specified exposure. All act after representation learning.

In-training debiasing.

Popular items can be penalized directly [1] or fairness cast as statistical independence [19]. Model-Agnostic Counterfactual Reasoning (MACR) [37] and Disentangling Interest and Conformity with Causal Embedding (DICE) [48] use counterfactual or disentangled representations; SGL [38], Simple Graph Contrastive Learning (SimGCL) [42], and its extension XSimGCL [43] improve representation geometry and can raise long-tail coverage; HetroFair [13] makes aggregation fairness-aware, and FairDA [41] targets demographic fairness. PBiLoss [27] penalizes scoring a user's popular items above less popular ones with a fixed weight. None imposes an absolute long-tail exposure target; we use SimGCL and PBiLoss as baselines and PBiLoss also as a surrogate inside ALF. Popularity also acts through embedding norms, addressed by normalization [15, 20], learned temperatures [8], and spectral analysis [21]; hence our cosine baseline.

Exposure allocation.

Fair exposure has been formulated as a constrained ranking policy [30, 31], amortized over rankings [4], measured as expected exposure [12], and controlled by feedback [26]. These approaches operate at the ranking-policy or exposure-allocation level; ALF instead feeds exposure feedback into representation learning.

Differentiable ranking.

Top-K selection gives no gradient, so SoftRank [33], ApproxNDCG [28], NeuralSort [14], optimal-transport sorting and top-K [11, 39], and isotonic ranking [5] relax the ranking operator, and recent work relaxes top-K fairness directly [46]. ALF-S needs only the aggregate tail mass of the top-K set, so it relaxes set membership over a sampled candidate pool at $O(B \cdot S)$ cost.

Constraint-based learning.

Fair classification has used saddle-point reductions [3], covariance constraints [44], and proxy-Lagrangians for non-decomposable constraints [10]. Recent methods adapt fairness pressure from training uncertainty [23] or a Lagrangian multiplier [40], use ADMM duals for a computational constraint [34], or relax top-K fairness with a tuned trade-off parameter [46]. The adaptive dual update is therefore not new; ALF applies it at a different interface, with an absolute tail-exposure target, feedback from the discrete metric, and an interchangeable surrogate.

3 Method

3.1 Problem Formulation

Let $\mathcal{U}$ and $\mathcal{I}$ be the user and item sets, $|\mathcal{I}| = N$. A graph encoder ϕ_θ produces embeddings $\mathbf{e}_u, \mathbf{e}_i \in \mathbb{R}^d$, and pairs are scored by cosine similarity, $s_\theta(u,i) = \mathbf{e}_u^\top \mathbf{e}_i / (\|\mathbf{e}_u\| \, \|\mathbf{e}_i\|)$, which removes the embedding norm from the score (§5.1). With $\mathcal{T}(u)$ the training history of u, the top-K list is $\mathcal{R}_K(u;\theta) = \mathrm{TopK}_{i \in \mathcal{I} \setminus \mathcal{T}(u)} \, s_\theta(u,i)$. The encoder can be any GNN; we use LightGCN [18] and a heterogeneous GraphSAGE [16] (§4.2).

Tail and exposure.

The head set $\mathcal{H}$ holds the 10% most popular items by training count. The reachable tail $\mathcal{I}_{\text{tail}}$ holds the least popular 20% of items with at least $c_{\min} = 5$ training interactions; rarer items are excluded because they cannot be recommended reliably. Uniform recommendation would give a tail share of $|\mathcal{I}_{\text{tail}}|/|\mathcal{I}| \approx 0.17$, 0.19, and 0.19 on MovieLens-100K, MovieLens-1M, and Yelp2018 (8,698 of 45,503 items on Yelp2018). For users $\mathcal{V}$, the tail share is

$$\mathrm{TailExp}(\theta;\mathcal{V}) = \frac{1}{K|\mathcal{V}|} \sum_{u \in \mathcal{V}} \left| \mathcal{R}_K(u;\theta) \cap \mathcal{I}_{\text{tail}} \right|, \tag{1}$$

a set-level, undiscounted measure. As an optional guardrail we also use the platform entropy $\mathrm{H}(\theta) = -\sum_{i\in\mathcal{I}} p_i(\theta)\log p_i(\theta)$ of the recommendation distribution p_i, normalized as $\bar{\mathrm{H}}(\theta) = \mathrm{H}(\theta)/\log N \in [0,1]$. The ideal problem is

$$\min_{\theta}\ \mathcal{L}_{\mathrm{rank}}(\theta) \quad \text{s.t.} \quad \mathrm{TailExp}(\theta) \geq \tau_{\mathrm{tail}}, \quad \bar{\mathrm{H}}(\theta) \geq \bar{\tau}_{\mathrm{ent}}, \tag{2}$$

where τ_{tail} and the normalized entropy target $\bar{\tau}_{\mathrm{ent}} \in [0,1]$ are the requested operating point. Both constraints depend on discrete top-K lists, so training uses differentiable surrogates $\widetilde{T}$ and $\widetilde{H}$ in the Lagrangian-inspired objective

$$\mathcal{L}(\theta,\boldsymbol{\lambda}) = \mathcal{L}_{\mathrm{rank}}(\theta) + \lambda_{\mathrm{tail}}\big(\tau_{\mathrm{tail}} - \widetilde{T}(\theta)\big)_+ + \lambda_{\mathrm{ent}}\big(\bar{\tau}_{\mathrm{ent}} - \widetilde{H}(\theta)\big)_+, \tag{3}$$

with $\boldsymbol{\lambda} \geq \mathbf{0}$ and $(x)_+ = \max(0,x)$. This is the form used by ALF-S; ALF-P replaces the tail term with a pairwise surrogate (§3.4), and both share the dual update (§3.3).

Static popularity penalty.

The POP baseline rewards soft tail mass with a fixed weight,

$$\mathcal{L}_{\mathrm{POP}}(\theta) = \mathcal{L}_{\mathrm{rank}}(\theta) - \lambda_{\mathrm{POP}}\widetilde{T}(\theta), \tag{4}$$

without a target or rectifier, so it keeps pushing exposure up regardless of the level reached. It is a static popularity regularizer [1], not ALF with a fixed multiplier.

3.2 Softmax Surrogate (ALF-S)

Soft tail mass.

For a batch of B users we draw a candidate pool $\mathcal{S}$ of 5000 item indices uniformly with replacement, redrawn per batch; on MovieLens it covers most of the catalogue, on Yelp2018 it is a subsample. With L2-normalized embeddings, the scores $\mathbf{C} = \tilde{\mathbf{Z}}_U\tilde{\mathbf{Z}}_{\mathcal{S}}^\top$ give $\pi_{u,i} = \exp(C_{u,i}/t_\pi)/\sum_{j\in\mathcal{S}}\exp(C_{u,j}/t_\pi)$ and

$$\widetilde{T}(\theta) = \frac{1}{B}\sum_{u}\sum_{i\in\mathcal{I}_{\mathrm{tail}}\cap\mathcal{S}}\pi_{u,i}, \qquad \Delta_{\mathrm{tail}}(\theta) = \big(\tau_{\mathrm{tail}} - \widetilde{T}(\theta)\big)_+. \tag{5}$$

The training proxy does not mask history items, so it differs from the masked, full-catalogue proxy analysed in §6.3; it is a training signal, not a relaxation of the top-K count. The cost is $O(B\cdot 5000)$ per active batch.

Temperature.

Δ_{tail} gives a gradient only while the proxy is below the target, so t_π is chosen so that the unconstrained model's proxy lies below the targets of interest: 0.10 on MovieLens and 0.03 on Yelp2018 (Online Resource 1, Sect. S1). A target near the unconstrained exposure can leave the primal term inactive (§6.4), and the check should be repeated for very different catalogue sizes.

Entropy guardrail.
With $\bar{\pi}_i = \frac{1}{B}\sum_u \pi_{u,i}$, the surrogate $\widetilde{H} = -\sum_{i\in\mathcal{S}} \bar{\pi}_i \log \bar{\pi}_i / \log|\mathcal{S}| \in [0,1]$ is normalized by the pool's maximum entropy, so that a target cannot exceed what a sampled pool can attain, with violation $\Delta_{\text{ent}} = (\bar{\tau}_{\text{ent}} - \widetilde{H})_+$. It is a training surrogate for the catalogue-normalized $\bar{\text{H}}$: both express entropy as a fraction of its maximum, but over the pool rather than the catalogue, and the dual signal below uses the catalogue-normalized value. Satisfying the pool-normalized surrogate is therefore not taken as exact satisfaction of the catalogue-level entropy constraint. A head-mass surrogate exists in the code but is disabled in all experiments. The ALF-S penalty, $\lambda_{\text{tail}}\Delta_{\text{tail}} + \lambda_{\text{ent}}\Delta_{\text{ent}}$, is applied every $f = 2$ epochs with the multipliers held fixed.

3.3 Adaptive Dual Updates

Every $E_{\text{eval}} = 20$ epochs, the model is evaluated on held-out validation users and the violation signals

$$g_{\text{tail}}^{(t)} = \tau_{\text{tail}} - \text{TailExp}(\theta^{(t)}; \mathcal{D}_{\text{val}}), \qquad g_{\text{ent}}^{(t)} = \bar{\tau}_{\text{ent}} - \widetilde{H}_{\mathcal{I}}(\theta^{(t)}; \mathcal{D}_{\text{val}}) \tag{6}$$

are computed; the tail signal uses the exact top-K statistic, and $\widetilde{H}_{\mathcal{I}}$ is the softmax entropy over the full catalogue normalized by $\log N$, i.e. the smoothed counterpart of $\bar{\text{H}}$. A positive signal raises the multiplier and a negative one lets it fall. Each signal is clipped to $[-1,1]$, turned into the proposal $\lambda_\bullet + \eta_\bullet \bar{g}_\bullet$, and damped and projected:

$$\lambda_\bullet^{(t+1)} = \text{clip}_{[0,\lambda_{\max,\bullet}]}\Big[\beta\lambda_\bullet^{(t)} + (1-\beta)\big(\lambda_\bullet^{(t)} + \eta_\bullet \bar{g}_\bullet^{(t)}\big)\Big], \tag{7}$$

an effective step of $(1-\beta)\eta_\bullet \bar{g}_\bullet$. We use $\beta = 0.6$, $\lambda_{\max,\text{tail}} = 5$, and start dual updates after a warmup of $T_{\text{warm}} = 50$ epochs.

This is the metric–surrogate separation on which the method rests: the surrogate supplies the primal gradient, the discrete validation TailExp supplies the feedback, and evaluation reports the same discrete quantity on test users. Surrogate and metric need not agree numerically, only move together (§6.3). Eq. 3 motivates the penalty, but the multiplier behaves as a feedback controller on the measured metric. Because the objective is non-convex and the signals come from a changing model, we treat the update as an adaptive heuristic and make no saddle-point or convergence claim.

3.4 Pairwise Surrogate (ALF-P)

Because the dual update never differentiates the surrogate, any term whose weight moves tail exposure in one direction can replace it. ALF-P uses the PopPos term of PBiLoss [27]. For each user u in the batch with at least one training positive in $\mathcal{I}_{\text{tail}}$ and one in $\mathcal{H}$ (the set $\mathcal{B}'$), we draw one of each, i and j, and compute

$$\mathcal{L}_{\text{PBi}}(\theta) = \frac{1}{|\mathcal{B}'|}\sum_{u\in\mathcal{B}'} \text{softplus}\big(-[s_\theta(u,i) - s_\theta(u,j)]\big), \tag{8}$$

which asks the model to rank each user's own tail positives above their head positives. ALF-P trains with

$$\mathcal{L}_{\text{ALF-P}}(\theta) = \mathcal{L}_{\text{rank}}(\theta) + \lambda_{\text{tail}}\,\mathcal{L}_{\text{PBi}}(\theta) \tag{9}$$

at every batch, with λ_{tail} updated by Eq. 7 from g_{tail}. There is no rectifier and no entropy guardrail. Static PBiLoss is Eq. 9 with a fixed weight w, so the comparison isolates the controller. Because the PopPos term has a different scale, ALF-P uses $\eta_{\text{tail}} = 1.0$ instead of 0.3, set on MovieLens-100K with one seed and then fixed. The term compares each user's observed tail and head positives and gives no direct gradient on unseen items, so its ability to raise the exposure of unseen tail items is limited (§5.4).

3.5 Training Algorithm

Each batch uses the multi-negative softmax ranking loss

$$\mathcal{L}_{\text{rank}}(\theta) = -\frac{1}{B} \sum_{(u,i^+,\{i_n^-\})} \log \frac{\exp(s_\theta(u,i^+)/t_{\text{r}})}{\exp(s_\theta(u,i^+)/t_{\text{r}}) + \sum_n \exp(s_\theta(u,i_n^-)/t_{\text{r}})}, \tag{10}$$

with $t_{\text{r}} = 0.1$ and $N_{\text{neg}} = 100$ negatives drawn from a popularity-weighted multinomial (exponent 0.75) that excludes only training-history items. On fairness epochs, ALF-S optimizes

$$\mathcal{L}_{\text{train}} = \mathcal{L}_{\text{rank}} + \lambda_{\text{tail}}\Delta_{\text{tail}} + \lambda_{\text{ent}}\Delta_{\text{ent}}, \tag{11}$$

and $\mathcal{L}_{\text{rank}}$ otherwise, with Adam and gradient clipping. Algorithm 1 summarizes ALF-S; for ALF-P, lines 8–15 are replaced by adding $\lambda_{\text{tail}}\mathcal{L}_{\text{PBi}}$ to every batch and only λ_{tail} is updated. Early stopping uses validation NDCG, and the reported checkpoint follows §4.5. The learning rate and batch size are selected once per (dataset, backbone) and held fixed across seeds and targets (Online Resource 1, Sect. S1).

4 Experimental Setup

4.1 Datasets and Protocol

We use each dataset as released, without additional core filtering: **MovieLens-100K** (943 users, 1,682 items, about 100,000 interactions) and **MovieLens-1M** (6,040 users, 3,706 items, about 1,000,000 interactions) [17], with ratings treated as binary positives, and **Yelp2018** as released with KGAT [35] (45,919 users, 45,538 items, 1,183,610 interactions[1]). Densities are 0.063, 0.045, and 5.7×10^{-4}.

MovieLens uses a per-user temporal leave-one-out split (last interaction for test, second-to-last for validation). Yelp2018 has no timestamps, so we keep the official KGAT test partition and hold out one random training interaction per user for validation. Each Yelp2018 test interaction is a separate ranking query, and other test items of the same user are not masked; this keeps the released partition unchanged, applies to all methods, and makes absolute NDCG incomparable across datasets but not across

[1] Counted from the released training and test files (930,032 and 253,578); the KGAT paper reports 1,185,068.

Algorithm 1 Adaptive Lagrangian training for fairness-aware graph recommendation.

Require: Encoder ϕ_θ; targets $(\tau_{\text{tail}}, \bar{\tau}_{\text{ent}})$; dual rates $(\eta_{\text{tail}}, \eta_{\text{ent}})$; damping β; warmup T_{warm}; evaluation interval E_{eval}; fairness frequency f; candidate-pool size S; epoch budget $E_{\max}$; patience P

Initialize θ and $\boldsymbol{\lambda} \leftarrow \mathbf{0}$
Compute reachable tail set $\mathcal{I}_{\text{tail}}$ from training popularity
for epoch $= 1, \ldots, E_{\max}$ **do**
  for each mini-batch $(u, i^+, \{i_n^-\})$ **do**
    Compute current embeddings $\mathbf{Z}_U, \mathbf{Z}_I$
    Sample N_{neg} popularity-weighted negatives
    Compute $\mathcal{L}_{\text{rank}}$ using Eq. 10
    if epoch mod $f = 0$ **then**
      Sample S item indices uniformly with replacement as pool $\mathcal{S}$
      Compute candidate scores
      Compute $\widetilde{T}$, Δ_{tail}, and Δ_{ent}
      Use $\mathcal{L}_{\text{train}}$ (Eq. 11) as the batch objective
    else
      Use $\mathcal{L}_{\text{rank}}$ as the batch objective
    end if
    Update θ with Adam and gradient clipping
  end for
  if epoch mod $E_{\text{eval}} = 0$ **then**
    Evaluate validation TailExp and smoothed fairness quantities
    if epoch $> T_{\text{warm}}$ **then**
      Update $\lambda_{\text{tail}}, \lambda_{\text{ent}}$ using Eq. 7
    end if
    Update validation early-stopping criterion
  end if
  if early-stopping patience $\geq P$ **then**
    break
  end if
end for
return final training state $(\theta, \boldsymbol{\lambda})$

methods. Exposure metrics use one top-K list per test user on every dataset. Items without a training-side representation are excluded from the evaluation catalogue for all methods.

4.2 Backbones

LightGCN [18] uses $L = 2$ layers, $d = 64$, and mean pooling over layers, with cosine scoring. **GraphSAGE** [16] is a heterogeneous variant with two `SAGEConv` layers, $d = 64$, ReLU, and user, item, and category nodes (movie genres; KGAT business

categories on Yelp2018), with relation outputs summed (Online Resource 1, Sect. S2). Neither backbone has a built-in fairness mechanism.

4.3 Baselines

All methods share splits, negative sampling, encoder, and five seeds; the main comparison uses LightGCN. POP and PBiLoss are the fixed-weight counterparts of ALF-S and ALF-P. We write λ_{tail}, λ_{ent} for the ALF multipliers and λ_{POP}, λ_{xQ}, λ_{cl} for the POP, xQuAD, and SimGCL weights.

BASE trains the backbone with the ranking loss and L_2 regularization only. **POP** adds the static reward of Eq. 4, swept over $\lambda_{\text{POP}} \in \{0.01, 0.05, 0.10, 0.15, 0.20, 0.25, 0.30, 0.40, 0.50, 0.75\}$ on every dataset. **xQuAD** [2], in its Binary and Smooth variants, reranks the top-500 BASE candidates with $\lambda_{\text{xQ}} \in \{0, 0.1, 0.3, 0.5, 0.7, 0.9\}$; $\lambda_{\text{xQ}} = 0$ reproduces BASE (Online Resource 1, Sect. S3). **SimGCL** [42] adds a contrastive loss with $\lambda_{\text{cl}} = 0.05$ on MovieLens and 0.10 on Yelp2018 and gives one operating point per dataset.

PBiLoss [27] adds $w\,\mathcal{L}_{\text{PBi}}$ (Eq. 8, PopPos variant) with $w \in \{0.05, 0.1, 0.2, 0.5, 1.0, 2.0\}$. Our version is an adaptation: it uses cosine scores and the softmax base loss instead of inner products and BPR, so the published weight (0.005) is far too small; even $w = 0.05$ moves MovieLens-100K exposure only from 0.040 to 0.054. The PopNeg variant left TailExp@10 at 0.040–0.041 for every weight in a single-seed MovieLens-100K check (0.037 for BASE) and is not reported.

ALF-S targets are listed in Online Resource 1, Tables S9–S11. **ALF-P** follows the ALF-S targets on MovieLens (0.05–0.65 and 0.05–0.85). On Yelp2018 it uses {0.10, 0.15, 0.20, 0.35, 0.50}: BASE already exceeds 0.05, and ALF-P had saturated by 0.35 (TailExp@10 of 0.196, and 0.214 at 0.50), so higher targets would add no new operating points; the saturated targets are reported.

4.4 Metrics

Test metrics use $K = 10$ and full-catalogue ranking with training history masked: NDCG@10, Recall@10, Coverage@10 (fraction of items appearing in any list), TailExp@10, and exposure Gini. For hypervolume (HV), each method's Pareto front in the TailExp@10–NDCG@10 plane is integrated against a reference NDCG of 0.010, 0.001, and 0.0005 on MovieLens-100K, MovieLens-1M, and Yelp2018. Because methods reach different exposure ranges, HV is computed for a pair of methods (A, B) only over their shared window

$$\left[\max\{\min \text{TailExp}_A, \min \text{TailExp}_B\},\ \min\{\max \text{TailExp}_A, \max \text{TailExp}_B\}\right], \tag{12}$$

integrating the best NDCG@10 attained at or above each exposure level. Intervals use 1000 bootstrap resamples over seeds, with paired resamples for the difference $\text{HV}_A - \text{HV}_B$.

4.5 Checkpoint Selection

Among logged epochs with validation NDCG within 95% of the best (plus the final epoch), we report the one with the highest validation TailExp if it exceeds that of the best-NDCG epoch by at least 0.01, and otherwise the best-NDCG epoch. This rule is used for every method and run except the ALF-S sweeps on MovieLens-1M and Yelp2018.

The ALF-S sweeps on MovieLens-1M and Yelp2018 are the exception: they report the final training state after early stopping, because there the dual variable overshoots and recovers (§6.4). The logged validation histories show that the rule would pick a different epoch in 38 of 45 MovieLens-1M runs and 45 of 50 Yelp2018 runs, typically an early epoch during the overshoot, since the rule maximizes exposure within the NDCG band; for targets from 0.15 to 0.65, validation tail exposure at those epochs exceeds the final state by 0.05–0.07 and 0.02–0.09. The rule was designed for methods without a target, for which more exposure at similar accuracy is preferable; for a controller, it selects the overshoot. The controller does not depend on the exception: ALF-P uses the common rule and attains its targets within 0.004 on MovieLens-1M up to $\tau_{\text{tail}} = 0.35$ (Table 5). Trajectories are shown in Online Resource 1, Fig. S2. For every other method, and for ALF-S on MovieLens-100K, the rule's checkpoint and the final state are close: their validation TailExp differs by at most 0.008 on average and 0.031 in any run (the ablation arms, which run the ALF-S controller under the rule, reach 0.013 and 0.033), against a mean of 0.036 (up to 0.10) for ALF-S on MovieLens-1M and Yelp2018. Every ALF-S run on these two datasets was also evaluated on the test set at the rule's checkpoint: the hypervolume comparisons of §efsec:cross-dataset change by at most one percentage point (ALF-S over POP +9.6% and +13.1%, intervals excluding zero), whereas the target error rises to 0.02–0.09 (Online Resource 1, Table S14). The exception therefore affects target tracking, not the frontier comparisons.

4.6 Implementation Details

Models use PyTorch and PyTorch Geometric, Adam, and gradient clipping at norm 5. ALF-S, ALF-P, and PBiLoss run for at most 1000 epochs with an early-stopping patience of 20 evaluations; BASE and POP converge earlier and use a 500-epoch cap on MovieLens-1M and Yelp2018, and extending them did not materially change their results (Online Resource 1, Sect. S4). Controller settings are identical across datasets, seeds, and targets except the dual step (0.3 for ALF-S, 1.0 for ALF-P) and the ALF-S proxy temperature; all values are in Online Resource 1, Table S2.

5 Main Results

We first fix the reference point (scoring, §5.1; headline operating points, §5.2), then compare frontiers and target control (§5.3) and reachable ranges (§5.4). Results use LightGCN, five seeds, and cosine scoring unless stated otherwise. Figure 1 first shows how each method's control parameter maps to attained exposure on the three datasets; it is discussed with target control in §5.3.

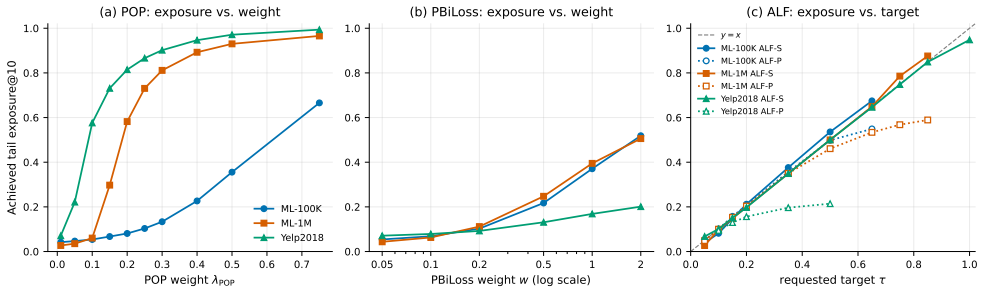


Fig. 1 Portability of the fairness control parameter on MovieLens-100K (circles), MovieLens-1M (squares) and Yelp2018 (triangles). (a) Attained TailExp@10 against the static softmax penalty weight λ_{POP}. (b) Attained TailExp@10 against the PBiLoss weight w (log scale). (c) Attained TailExp@10 against the requested target τ under ALF-S (solid) and ALF-P (dotted, open markers), with the identity line dashed; ALF-P saturates near 0.55, 0.59, and 0.21. Means over five seeds

Table 1 BASE under inner-product versus cosine scoring, retrained from scratch with the scoring convention changed and the corresponding ranking temperature adjusted. Mean ± sd over five seeds ($\{42, 43, 44, 45, 46\}$). Cosine rows are a separate retraining and differ slightly from the BASE rows of Table 2 on MovieLens-1M and Yelp2018.

Dataset	Scoring	NDCG@10	TailExp@10	Coverage@10
MovieLens-100K	Inner product	0.0685 ± 0.0023	0.0074 ± 0.0005	0.3734 ± 0.0054
MovieLens-100K	Cosine	0.0647 ± 0.0018	0.0398 ± 0.0027	0.6847 ± 0.0079
MovieLens-1M	Inner product	0.0292 ± 0.0006	0.0000 ± 0.0000	0.2174 ± 0.0037
MovieLens-1M	Cosine	0.0326 ± 0.0010	0.0260 ± 0.0006	0.8122 ± 0.0066
Yelp2018	Inner product	0.0116 ± 0.0001	0.0000 ± 0.0000	0.0069 ± 0.0001
Yelp2018	Cosine	0.0113 ± 0.0001	0.0658 ± 0.0013	0.8417 ± 0.0015

5.1 Scoring Function as an Experimental Control

BASE is retrained with inner-product or cosine scoring, keeping everything else fixed except the ranking temperature, which must differ: cosine scores lie in $[-1, 1]$, so $t_{\mathrm{r}} = 1$ would leave the softmax in Eq. 10 nearly uniform, whereas unbounded inner products are well scaled at $t_{\mathrm{r}} = 1$. We therefore use $t_{\mathrm{r}} = 1.0$ for inner product and 0.1 for cosine, and treat the comparison as a protocol-level diagnostic of the scoring and temperature convention rather than a causal, scoring-only comparison (Table 1).

Under inner-product scoring, recommendations collapse onto a few head items: no tail item is recommended on MovieLens-1M or Yelp2018, coverage is 0.37, 0.22, and 0.007, and Gini is 0.85, 0.95, and 0.999. NDCG changes little and not in a consistent direction (-5.5%, $+12\%$, -2.8% with cosine), so the difference is not a weaker model. Cosine BASE exposure (0.040, 0.026, 0.066) is still far below the uniform reference, and cosine BASE is the reference for all remaining results.

Table 2 Headline operating point per dataset. ALF-S and ALF-P use the dataset-specific target; each baseline is shown at the grid point closest to the attained TailExp@10 of the ALF instance it is paired with (POP and xQuAD-S with ALF-S, PBiLoss with ALF-P). ΔNDCG and ΔTail are relative to BASE and are computed from unrounded means; Yelp2018 NDCG is shown to five decimals so that the deltas can be reproduced. Means over five seeds.

Dataset	Method	NDCG@10	ΔNDCG	TailExp@10	ΔTail
ML-100K	BASE	0.0647	–	0.0398	–
	ALF-S (τ=0.10)	0.0587	−9.2%	0.0817	+105%
	POP (λ=0.20)	0.0588	−9.1%	0.0809	+103%
	xQuAD-S (λ=0.1)	0.0626	−3.2%	0.0912	+129%
	ALF-P (τ=0.10)	0.0624	−3.5%	0.1013	+154%
	PBiLoss (w=0.2)	0.0597	−7.6%	0.1026	+157%
ML-1M	BASE	0.0333	–	0.0273	–
	ALF-S (τ=0.10)	0.0308	−7.7%	0.0982	+260%
	POP (λ=0.10)	0.0319	−4.3%	0.0602	+121%
	xQuAD-S (λ=0.1)	0.0324	−2.6%	0.0594	+118%
	ALF-P (τ=0.10)	0.0309	−7.3%	0.1000	+267%
	PBiLoss (w=0.2)	0.0309	−7.3%	0.1117	+309%
Yelp2018	BASE	0.01136	–	0.0632	–
	ALF-S (τ=0.15)	0.01001	−11.9%	0.1509	+139%
	POP (λ=0.05)	0.00901	−20.7%	0.2214	+250%
	xQuAD-S (λ=0.1)	0.01091	−4.0%	0.1242	+97%
	ALF-P (τ=0.15)	0.01051	−7.5%	0.1299	+106%
	PBiLoss (w=0.5)	0.01041	−8.4%	0.1310	+107%

5.2 Headline Operating Points

Table 2 uses $\tau_{\text{tail}} = 0.10$ on the MovieLens datasets and 0.15 on Yelp2018, whose higher BASE exposure (0.063) would make 0.10 only a 1.6× lift; 0.15 gives 2.4×, comparable to MovieLens. The Yelp2018 point at 0.10 (0.1009, −4.2% NDCG) is in Online Resource 1, Table S11. Baselines are compared at attained exposure, not at matching parameter values.

ALF-S raises TailExp@10 on all three datasets at NDCG costs of 9.2%, 7.7%, and 11.9%. ALF-P lands within 0.002 of the target on both MovieLens datasets at lower cost (3.5% and 7.3%) but reaches only 0.130 of 0.15 on Yelp2018. At almost the same exposure, ALF-P is more accurate than the nearest PBiLoss weight on MovieLens-100K (0.0624 against 0.0597) and equal on MovieLens-1M, and ALF-S and POP are nearly identical on MovieLens-100K (0.0587 against 0.0588 at 0.082 and 0.081). Elsewhere the nearest grid points reach different exposure levels; xQuAD-Smooth costs less NDCG than ALF-S on all three datasets but, on MovieLens-1M, at lower exposure (0.059). The trade-off is therefore compared over full ranges next.

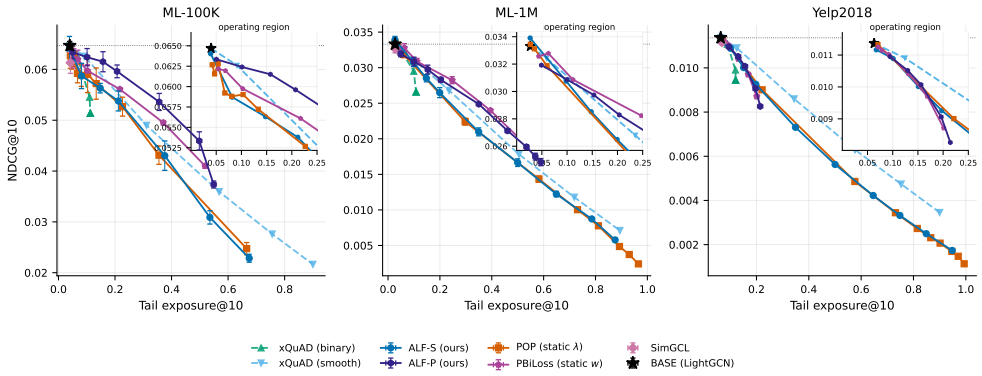


Fig. 2 Accuracy–exposure frontiers on the three benchmarks. Each panel plots NDCG@10 against TailExp@10 for ALF-S (circles), ALF-P (hexagons), the static softmax penalty grid POP (squares), the PBiLoss weight sweep (pentagons), xQuAD-Binary (upward triangles), xQuAD-Smooth (downward triangles), SimGCL (diamond), and the unconstrained cosine-scored backbone BASE (star). Error bars are standard errors over five seeds; the dotted line marks BASE accuracy, and the inset magnifies TailExp@10 ≤ 0.25

5.3 Accuracy–Exposure Trade-off and Target Control

Frontiers.

Figure 2 shows all frontiers (full values in Online Resource 1, Tables S9–S12). On the MovieLens datasets the pairwise frontiers (ALF-P, PBiLoss) lie above the softmax frontiers (ALF-S, POP) over most of their shared range; on Yelp2018 the four in-training frontiers nearly coincide below a tail share of 0.2. Fixed grids also resolve the operating range unevenly: adjacent POP weights jump from 0.30 to 0.58 exposure on MovieLens-1M and from 0.22 to 0.58 on Yelp2018, and the six PBiLoss weights span only 0.07–0.20 on Yelp2018.

Common-range hypervolume.

Table 3 compares each ALF instance with the static use of its own surrogate, which isolates the controller, and the two instances with each other. ALF-S exceeds POP on all datasets, with the paired 95% bootstrap interval excluding zero on MovieLens-1M (+10.5%) and Yelp2018 (+13.2%) but not on MovieLens-100K (+5.4%). ALF-P exceeds PBiLoss on MovieLens-100K (+8.8%) and, narrowly, Yelp2018 (+1.2%), both with intervals excluding zero, and is slightly below it on MovieLens-1M (−1.3%, interval including zero). Target control thus has the higher point estimate in five of six comparisons, with intervals excluding zero in four; with five seeds, these intervals are descriptive rather than formal tests. Over ALF-P's reachable window, ALF-P exceeds ALF-S by 29%, 18%, and 2.3%. These comparisons are unchanged within one percentage point when ALF-S on MovieLens-1M and Yelp2018 is evaluated at the common rule's checkpoint (§4.5).

Table 3 Common-range hypervolume ($\times 10^3$) for pairs of methods, with bootstrap 95% confidence intervals (1000 resamples over seeds). The window is the tail-exposure interval attained by both methods of the pair; the last column is the paired difference $\mathrm{HV}_A - \mathrm{HV}_B$.

Dataset	A vs. B	Window	HV_A	HV_B	Difference
ML-100K	ALF-S vs. POP	[0.042, 0.666]	18.23 [17.38, 19.38]	17.30 [16.48, 18.23]	+0.93 [−0.44, +2.27]
	ALF-P vs. PBiLoss	[0.054, 0.519]	19.99 [19.36, 20.69]	18.38 [18.10, 18.56]	+1.61 [+0.99, +2.38]
	ALF-P vs. ALF-S	[0.050, 0.549]	21.02 [20.35, 21.72]	16.32 [15.44, 17.33]	+4.70 [+3.36, +5.77]
ML-1M	ALF-S vs. POP	[0.027, 0.876]	13.31 [13.09, 13.52]	12.04 [11.79, 12.31]	+1.27 [+0.92, +1.58]
	ALF-P vs. PBiLoss	[0.049, 0.506]	10.95 [10.84, 11.07]	11.10 [10.94, 11.22]	−0.14 [−0.33, +0.05]
	ALF-P vs. ALF-S	[0.049, 0.589]	12.35 [12.22, 12.47]	10.49 [10.29, 10.70]	+1.86 [+1.61, +2.07]
Yelp2018	ALF-S vs. POP	[0.071, 0.948]	4.159 [4.139, 4.178]	3.674 [3.632, 3.706]	+0.485 [+0.446, +0.531]
	ALF-P vs. PBiLoss	[0.095, 0.201]	0.984 [0.979, 0.990]	0.972 [0.962, 0.979]	+0.012 [+0.003, +0.024]
	ALF-P vs. ALF-S	[0.095, 0.214]	1.087 [1.078, 1.092]	1.062 [1.053, 1.071]	+0.025 [+0.013, +0.035]

Table 4 NDCG@10 at fixed tail-exposure levels, by linear interpolation along each method's empirical frontier. A dash marks a level outside the method's attained range. Best value per row in bold.

Dataset	TailExp	ALF-S	POP	ALF-P	PBiLoss	xQuAD-S
ML-100K	0.10	0.0580	0.0590	**0.0624**	0.0599	0.0621
	0.20	0.0543	0.0540	**0.0599**	0.0567	0.0559
	0.35	0.0447	0.0435	**0.0539**	0.0504	0.0470
	0.50	0.0336	0.0346	**0.0456**	0.0421	0.0394
ML-1M	0.10	0.0307	0.0303	0.0309	**0.0313**	0.0311
	0.20	0.0265	0.0263	0.0284	**0.0292**	0.0279
	0.35	0.0210	0.0209	0.0249	**0.0253**	0.0230
	0.50	0.0167	0.0167	0.0199	**0.0203**	0.0180
Yelp2018	0.10	0.01090	0.01088	0.01089	0.01090	**0.01109**
	0.20	0.00922	0.00934	0.00890	0.00874	**0.01011**
	0.35	0.00730	0.00750	–	–	**0.00854**
	0.50	0.00563	0.00575	–	–	**0.00705**

Matched exposure.

Table 4 interpolates NDCG at fixed exposure levels. On MovieLens-100K, ALF-P is the most accurate method at every level, ahead of PBiLoss by 0.0025–0.0035 and of xQuAD-Smooth by 0.0003–0.0069. On MovieLens-1M, PBiLoss leads ALF-P by 0.0004–0.0008, consistent with the tied hypervolume, and both lead xQuAD-Smooth from 0.20 upward. On Yelp2018, xQuAD-Smooth is most accurate at every level. ALF-S and POP are within 0.0012 everywhere. Reranking is therefore strongest on the sparsest catalogue, and a pairwise surrogate trained into the model is strongest at moderate and high exposure on the denser ones.

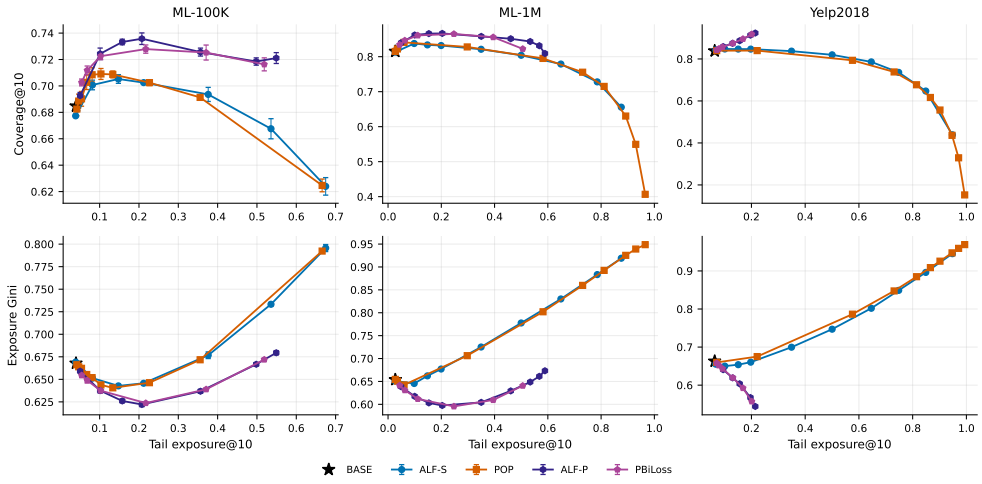


Fig. 3 Catalogue Coverage@10 (top row) and Exposure Gini (bottom row) along the ALF-S (circles), POP (squares), ALF-P (hexagons), and PBiLoss (pentagons) frontiers, plotted against attained TailExp@10; the star marks BASE

Exposure spread.

At the same tail share, the pairwise methods spread exposure more broadly (Fig. 3): at TailExp@10 = 0.20, Gini is 0.62, 0.60, and 0.56 for ALF-P against 0.65, 0.68, and 0.66 for ALF-S, and coverage is 0.74, 0.87, and 0.92 against 0.70, 0.83, and 0.85; PBiLoss and POP behave like ALF-P and ALF-S. A plausible reading is that the softmax surrogate rewards tail mass wherever it falls, whereas the pairwise term raises each user's own tail positives, which differ across users. For the softmax methods, coverage falls and Gini rises at high exposure (POP coverage 0.41 and 0.15 on MovieLens-1M and Yelp2018 at the largest λ), and xQuAD-Smooth loses coverage as its weight grows ($0.68 \to 0.44$, $0.82 \to 0.61$, $0.84 \to 0.55$ at $\lambda = 0.9$).

Target control.

ALF-S tracks the target closely over the interior of the MovieLens-1M and Yelp2018 sweeps (median absolute errors 0.0018 and 0.0016; Table 5, Fig. 4). On MovieLens-100K its errors are larger (median 0.0183): it falls short at the two lowest targets, where at $\tau_{\text{tail}} = 0.10$ all runs reach the epoch budget with λ_{tail} still rising, and overshoots from 0.35. ALF-P tracks more closely there (median 0.0041 up to 0.50) and within 0.004 up to 0.35 on MovieLens-1M; beyond that, and above 0.10 on Yelp2018, it saturates (§5.4). Across-seed standard deviations are mostly 1–9 $\times 10^{-3}$, and with five seeds most differences between targets are within sampling noise. The systematic exceptions are the saturated ALF-P targets, where all seeds reach the same ceiling (standard deviation at most 2×10^{-3} except on MovieLens-100K at 0.65), and ALF-S on Yelp2018, whose variation is largest at mid-range targets and smallest where exposure is pinned by an inactive constraint or near-saturation. A static weight, by contrast, maps to very different exposure across datasets (Fig. 1): POP at $\lambda = 0.15$

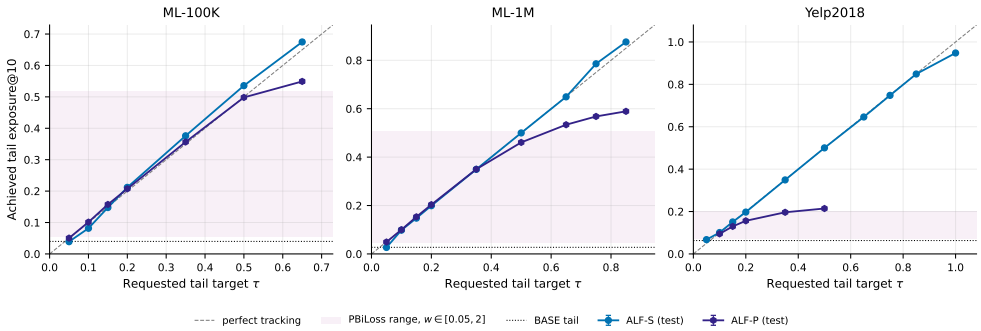


Fig. 4 Constraint satisfaction per dataset. Attained TailExp@10 against the requested target τ for ALF-S (circles) and ALF-P (hexagons), with the identity line dashed and the unconstrained backbone's tail exposure marked by a dotted horizontal line. The shaded band is the exposure range covered by the PBiLoss weight sweep ($w \in [0.05, 2]$). Error bars are standard errors over five seeds; terminal dual variables are listed in Online Resource 1, Tables S12 and S13

Table 5 Target-control error: absolute difference between the requested target τ_{tail} and the mean attained TailExp@10 over five seeds, $\pm$ the standard deviation of the attained TailExp@10 across seeds, both $\times 10^{-3}$. A dash marks a target not run for that instance.

	ML-100K		**ML-1M**		**Yelp2018**	
τ_{tail}	ALF-S	ALF-P	ALF-S	ALF-P	ALF-S	ALF-P
0.05	11.0 ± 2.6	0.4 ± 2.9	23.3 ± 1.6	0.9 ± 2.7	17.4 ± 1.0	–
0.10	18.3 ± 5.3	1.3 ± 5.4	1.8 ± 5.6	0.0 ± 4.6	0.9 ± 2.7	5.3 ± 3.6
0.15	2.3 ± 1.9	7.5 ± 2.7	2.1 ± 8.5	3.3 ± 5.6	0.9 ± 3.3	20.1 ± 2.2
0.20	11.9 ± 8.6	7.4 ± 5.2	0.7 ± 9.1	3.6 ± 3.9	2.6 ± 5.6	43.8 ± 2.1
0.35	26.0 ± 8.1	6.4 ± 2.3	0.6 ± 6.3	0.1 ± 3.3	0.9 ± 7.2	153.6 ± 0.6
0.50	35.7 ± 4.1	1.8 ± 6.6	0.1 ± 8.7	39.3 ± 5.6	0.3 ± 7.3	285.7 ± 1.2
0.65	24.8 ± 11.3	100.9 ± 13.3	1.0 ± 7.2	116.2 ± 1.4	3.9 ± 3.3	–
0.75	–	–	35.6 ± 2.6	182.0 ± 1.2	2.0 ± 1.7	–
0.85	–	–	25.8 ± 2.6	260.9 ± 1.9	1.2 ± 1.2	–
1.00	–	–	–	–	52.0 ± 0.6	–

gives 0.067, 0.297, and 0.731, and PBiLoss at $w = 2$ gives 0.519, 0.506, and 0.201. The PBiLoss weight transfers between the MovieLens catalogues, a practical strength, but not to Yelp2018.

5.4 Reachable Exposure Range

ALF-S follows the identity line up to the highest targets tested (0.65, 0.85, and 1.00). ALF-P tracks the target up to about 0.50, 0.35–0.50, and 0.10 on the three datasets and then saturates near 0.55, 0.59, and 0.21. This is not a failure of the dual update: λ_{tail} keeps rising (to 2.6, 3.3, and 2.8, below the clamp of 5), but extra weight buys little

Table 6 BASE, SimGCL, and ALF-S (headline target) on the same LightGCN backbone. Best value per dataset and column is shown in bold. Means over five seeds.

Dataset	Method	NDCG@10	TailExp@10	Coverage@10
ML-100K	BASE	**0.0647**	0.0398	0.685
	SimGCL	0.0613	0.0423	0.696
	ALF-S	0.0587	**0.0817**	**0.701**
ML-1M	BASE	**0.0333**	0.0273	0.815
	SimGCL	0.0327	0.0271	0.818
	ALF-S	0.0308	**0.0982**	**0.838**
Yelp2018	BASE	**0.0114**	0.0632	0.838
	SimGCL	0.0112	0.0678	**0.855**
	ALF-S	0.0100	**0.1509**	0.847

exposure, and high-target runs stop early (about 470–550 epochs) as validation NDCG declines. Static PBiLoss ends at almost the same exposure on Yelp2018 (0.201 at $w = 2$), so the ceiling belongs to the surrogate, not the controller; a plausible, untested mechanism is that PopPos acts directly only on items in each user's history, which on a sparse catalogue cover little of the tail. The instances are therefore complementary: ALF-P gives the better frontier and broader spread within its reach on MovieLens, and ALF-S serves high targets and is the only option above 0.21 on Yelp2018, with the same interface.

6 Additional Validation

These analyses support the main results. The ablation and proxy analyses concern ALF-S; the dual-variable analysis covers both instances.

6.1 Comparison with SimGCL

SimGCL changes tail exposure only slightly relative to BASE (+0.0025, −0.0002, and +0.0046) with slightly lower NDCG@10 (Table 6); it offers no way to request an exposure level.

6.2 Constraint Ablation

Table 7 compares the tail constraint alone (A1), the entropy guardrail alone (A2), and both (A3), with the same dual machinery, at $\tau_{\text{tail}} = 0.15$ on MovieLens-100K and Yelp2018 and 0.10 on MovieLens-1M; the Yelp2018 entropy target is $\bar{\tau}_{\text{ent}} = 0.97$ (≈ 10.4 nats over the full catalogue). The tail constraint produces the exposure gain: entropy alone leaves TailExp near BASE (0.040, 0.026, 0.065). On the MovieLens datasets the entropy multiplier stays at zero, so A3 equals A1; on Yelp2018 it is active but changes NDCG, TailExp, and Gini by less than one standard deviation. The guardrail is thus no substitute for the tail constraint and has a negligible effect at

Table 7 Constraint ablation: tail only (A1), entropy only (A2), and the combined objective (A3), all at the validation-rule checkpoint. Mean ± sd over five seeds. On ML-100K and ML-1M, A3 equals A1 on every seed (up to 3×10^{-5} in Gini) because the entropy multiplier remains zero.

Dataset	Cond.	NDCG@10	TailExp@10	Gini	Coverage@10
ML-100K (τ_{tail}=0.15)	A1	0.0563 ± 0.0021	0.1477 ± 0.0019	0.6427 ± 0.0038	0.7052 ± 0.0073
	A2	0.0632 ± 0.0054	0.0404 ± 0.0036	0.6676 ± 0.0068	0.6813 ± 0.0113
	A3	identical to A1 on every metric and every seed			
ML-1M (τ_{tail}=0.10)	A1	0.0303 ± 0.0008	0.1108 ± 0.0123	0.6459 ± 0.0023	0.8379 ± 0.0033
	A2	0.0330 ± 0.0012	0.0262 ± 0.0010	0.6519 ± 0.0020	0.8127 ± 0.0030
	A3	identical to A1 on every metric and every seed			
Yelp2018 (τ_{tail}=0.15)	A1	0.0098 ± 0.0002	0.1703 ± 0.0058	0.6692 ± 0.0029	0.8386 ± 0.0030
	A2	0.0113 ± 0.0002	0.0653 ± 0.0018	0.6586 ± 0.0028	0.8401 ± 0.0025
	A3	0.0097 ± 0.0001	0.1695 ± 0.0051	0.6685 ± 0.0016	0.8385 ± 0.0011

Table 8 Agreement between the soft tail-mass proxy $\widetilde{T}$ and discrete validation TailExp, computed separately within each seed over all logged epochs of the headline run and summarized as mean ± sd across the five seeds; n is the number of logged epochs per seed. Online Resource 1, Fig. S3, shows the pooled data.

Dataset	n	Pearson r	Spearman ρ	Kendall τ	Proxy/metric ratio
ML-100K	50	0.942 ± 0.012	0.970 ± 0.013	0.876 ± 0.030	2.10 ± 0.04
ML-1M	49–50	0.961 ± 0.006	0.962 ± 0.016	0.874 ± 0.034	2.08 ± 0.02
Yelp2018	32–40	0.997 ± 0.001	0.960 ± 0.029	0.876 ± 0.054	1.12 ± 0.01

these targets. All ablation arms use the common checkpoint rule, so A1 and A3 show the controller's overshoot on MovieLens-1M and Yelp2018 (§4.5).

6.3 Proxy Agreement with the Discrete Metric

Because epochs within a run are autocorrelated, agreement is computed within each seed (Table 8). The masked, full-catalogue validation proxy tracks discrete TailExp closely (mean Pearson r of 0.942, 0.961, 0.997; Spearman ρ of 0.970, 0.962, 0.960; no seed below 0.90), but is higher in level (ratios 2.10, 2.08, 1.12), and the ratio varies along the sweep: on MovieLens-1M it is about 3.8 near BASE exposure, about 1 near 0.35, and below 1 at 0.50–0.65. The proxy therefore preserves the direction and ordering of changes well enough to train on, but is not a calibrated estimate of the metric, which is why the dual update uses the metric itself.

6.4 Cross-Seed Stability of the Dual Variable

We examine cross-seed stability, not formal convergence (Online Resource 1, Tables S12–S13 and Fig. S4). ALF-S multipliers rise with the target on MovieLens-100K and Yelp2018; on MovieLens-1M they are roughly flat over 0.05–0.20 and rise

thereafter. Across targets, the mean coefficient of variation over seeds is 4.0%, 2.7%, and 1.0% for ALF-S and 2.6%, 1.7%, and 2.3% for ALF-P, whose multipliers rise strictly with the target and are larger (at $\tau_{\text{tail}} = 0.20$: 0.47, 0.39, 0.76 against 0.40, 0.13, 0.05), reflecting the surrogate's scale. No multiplier reaches the clamp of 5, so neither the results nor the ALF-P ceiling reflect clamp saturation. Two edge cases illustrate the headroom condition of §3.2: on Yelp2018 at 0.05, BASE already exceeds the target and $\lambda_{\text{tail}} = 0$; on MovieLens-1M at 0.05, the metric stays below the target while the proxy stays above it, so λ_{tail} grows but the primal term never activates, and the training trajectory equals that of the entropy-only ablation.

6.5 Computational Cost

Inference is unchanged and needs no reranking stage. ALF-S adds $O(B \cdot S)$ score computations ($S = 5000$) every $f = 2$ epochs, ALF-P adds $O(B\,d)$ per batch using index tensors built once, and the dual update is a scalar operation per evaluation. Shared-GPU contention makes timings noisy, but in the logs ALF-S takes about 1.2–1.9× the time of POP on MovieLens-1M and Yelp2018 (which use a 500-epoch cap for POP), and ALF-P about 0.8× that of ALF-S. A frontier still needs one run per target, as a static method needs one per weight.

7 Discussion

The results separate two questions that fixed-weight methods conflate: which surrogate shapes the frontier, and how the operating point on it is chosen. The surrogate largely determines the frontier: on the MovieLens datasets the pairwise PopPos term, static or under ALF, gives higher NDCG and lower Gini than the softmax term at matched exposure (Table 4, Fig. 3), while on Yelp2018 the in-training surrogates are close and post-hoc reranking is ahead. The controller determines how the operating point is chosen: for the same surrogate, target control had the higher hypervolume in five of six comparisons, with intervals excluding zero in four (Table 3), without rediscovering the weight per dataset (Fig. 1). ALF borrows its dual update from constraint-based learning [3, 10, 44]; its contribution is the interface, which lets it sit on top of an existing debiasing loss, as ALF-P does for PBiLoss, rather than compete with it.

7.1 Practical Implications

ALF-S remained responsive across the full tested target range, although tracking error grew at the highest targets, whereas ALF-P saturates at a dataset-dependent ceiling (§5.4). This suggests the pairwise instance for moderate targets on catalogues where users' histories contain both head and tail items, and the softmax instance for high targets or sparse catalogues; the interface is the same either way. Post-hoc reranking remains competitive: xQuAD has the highest NDCG at matched exposure on Yelp2018, but falls behind the pairwise in-training methods at moderate and high exposure on the MovieLens datasets and loses coverage at high weights. Sparsity is not a simple guide: at $\tau_{\text{tail}} = 0.10$ ALF-S costs 9.2%, 7.7%, and 4.2% NDCG on MovieLens-100K, MovieLens-1M, and Yelp2018, largest on the densest catalogue,

although sparsity does appear to cap the pairwise surrogate. With three datasets these are hypotheses, not established trends.

7.2 Limits to Validity

Statistics and training. Five seeds give limited power, and bootstrap intervals over seeds are not formal tests. Convergence was checked at selected operating points only: the MovieLens-100K ALF-S runs at $\tau_{\text{tail}} = 0.10$ end at the epoch budget with the dual variable still rising, and high-target runs on the larger datasets end by NDCG-based early stopping. The soft proxy is correlated with, but not calibrated to, discrete TailExp.

Protocol. The ALF-S sweeps on MovieLens-1M and Yelp2018 report the final training state rather than the validation-rule checkpoint used for everything else, because that rule tends to select transient overshoot epochs of the dual variable (§4.5). Their comparisons therefore combine two checkpoint protocols; evaluated at the rule's checkpoint instead, the hypervolume results are unchanged within one percentage point, while target tracking degrades. Negative sampling excludes only training-history items, which may introduce occasional false negatives for all methods.

Scope. The controller was tested with two surrogates using different dual step sizes, the ALF-P step having been set on one dataset and seed. Our PBiLoss is an adaptation (cosine scores, softmax base loss, our head and tail sets), so its absolute numbers are not comparable with the original publication, and the mechanism behind the ALF-P ceiling is a hypothesis. The ablation, proxy, scoring, SimGCL, and GraphSAGE analyses cover ALF-S only; the baselines and ALF-P were run with LightGCN only. The head constraint is inactive, items with fewer than five training interactions are outside the tail, and the study covers three implicit-feedback datasets and item-side exposure only.

8 Conclusion

We introduced Adaptive Lagrangian Fairness (ALF), a primal–dual controller that adjusts the weight of a differentiable surrogate until a requested top-K long-tail exposure is met on validation data. Because the dual update uses the discrete metric, the surrogate is interchangeable; we instantiated it with a softmax tail-mass proxy (ALF-S) and the PBiLoss pairwise objective (ALF-P). On MovieLens-100K, MovieLens-1M, and Yelp2018, fixed weights for both surrogates mapped to different exposure on different datasets, whereas both ALF instances followed the requested target with controller settings shared apart from surrogate-specific scaling and calibration. Against a weight sweep of the same surrogate, target control gave a higher common-range hypervolume in five of six comparisons, with bootstrap intervals excluding zero in four. The surrogate determined the rest: the pairwise surrogate gave the more accurate and more evenly spread frontier on MovieLens but saturated at a dataset-dependent ceiling, while the softmax surrogate remained responsive across the full tested range. Post-hoc reranking remained the most accurate option on the sparsest catalogue, and the scoring convention changed baseline exposure enough that it should be reported.

The main limitations are the small numbers of seeds, datasets, and surrogates, and high-target dual dynamics that need not settle within a fixed budget (§7.2). Future work should analyse the dual dynamics, extend the reach of pairwise surrogates, combine surrogates within one controller, and extend ALF to group-level and position-aware exposure.

Acknowledgements. We thank Dr. Nuwee Wiwatwattana for helpful comments. The Asian Institute of Technology provided the computational resources used in this work.

Statements and Declarations

Funding. No funding was received to assist with the preparation of this manuscript.

Competing interests. The authors have no competing interests to declare that are relevant to the content of this article.

Ethics approval and consent to participate. Not applicable. This study uses only publicly available, anonymized benchmark datasets and involves no human participants.

Consent for publication. Not applicable.

Data availability. This study uses three publicly available benchmark datasets: MovieLens-100K and MovieLens-1M (https://grouplens.org/datasets/movielens/), and the Yelp2018 release distributed with the KGAT benchmark. No new data were generated.

Code availability. The implementation is publicly available at https://github.com/puskr99/Adaptive-Lagrangian-Fairness.

Author contributions. P.A. conceived the method, implemented the experiments, analyzed the results, and drafted the manuscript. C.P. supervised the work, advised on the experimental design and evaluation, and critically revised the manuscript. Both authors read and approved the final version.

References

[1] Abdollahpouri H, Burke R, Mobasher B (2017) Controlling popularity bias in learning-to-rank recommendation. In: Proceedings of the 11th ACM Conference on Recommender Systems. ACM, pp 42–46, https://doi.org/10.1145/3109859.3109912

[2] Abdollahpouri H, Burke R, Mobasher B (2019) Managing popularity bias in recommender systems with personalized re-ranking. In: Proceedings of the 32nd International FLAIRS Conference. AAAI Press, pp 413–418

[3] Agarwal A, Beygelzimer A, Dudík M, et al (2018) A reductions approach to fair classification. In: Proceedings of the 35th International Conference on Machine Learning. PMLR, pp 60–69

[4] Biega AJ, Gummadi KP, Weikum G (2018) Equity of attention: Amortizing individual fairness in rankings. In: Proceedings of the 41st International ACM SIGIR

Conference on Research and Development in Information Retrieval (SIGIR), pp 405–414, https://doi.org/10.1145/3209978.3210063

[5] Blondel M, Teboul O, Berthet Q, et al (2020) Fast differentiable sorting and ranking. In: Proceedings of the 37th International Conference on Machine Learning (ICML), pp 950–959

[6] Burke R (2017) Multisided fairness for recommendation. In: Workshop on Fairness, Accountability, and Transparency in Machine Learning (FAT/ML)

[7] Chen J, Dong H, Wang X, et al (2023) Bias and debias in recommender system: A survey and future directions. ACM Transactions on Information Systems 41(3):67:1–67:39. https://doi.org/10.1145/3564284

[8] Chen J, Wu J, Wu J, et al (2023) Adap-τ: Adaptively modulating embedding magnitude for recommendation. In: Proceedings of the ACM Web Conference 2023 (WWW). ACM, https://doi.org/10.1145/3543507.3583363

[9] Choi YJ, Yun WS, Jeong C, et al (2026) GCNs meet long-tail: Embedding norm bias in GCN-based recommendations. Applied Soft Computing 186:114226. https://doi.org/10.1016/j.asoc.2025.114226

[10] Cotter A, Jiang H, Gupta MR, et al (2019) Training well-generalizing classifiers for fairness metrics and other data-dependent constraints. In: Proceedings of the 36th International Conference on Machine Learning. PMLR, pp 1397–1405

[11] Cuturi M, Teboul O, Vert JP (2019) Differentiable ranking and sorting using optimal transport. In: Advances in Neural Information Processing Systems (NeurIPS)

[12] Diaz F, Mitra B, Ekstrand MD, et al (2020) Evaluating stochastic rankings with expected exposure. In: Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM), pp 275–284, https://doi.org/10.1145/3340531.3411962

[13] Gholinejad N, Haghir Chehreghani M (2025) Heterophily-aware fair recommendation using graph convolutional networks. Neurocomputing https://doi.org/10.1016/j.neucom.2025.131956, preprint: arXiv:2402.03365

[14] Grover A, Wang E, Zweig A, et al (2019) Stochastic optimization of sorting networks via continuous relaxations. In: International Conference on Learning Representations (ICLR)

[15] Gupta P, Garg D, Malhotra P, et al (2019) NISER: Normalized item and session representations to handle popularity bias. In: Proceedings of the 1st International Workshop on Graph Representation Learning and its Applications (GRLA@CIKM 2019), URL https://arxiv.org/abs/1909.04276

[16] Hamilton WL, Ying R, Leskovec J (2017) Inductive representation learning on large graphs. In: Advances in Neural Information Processing Systems

[17] Harper FM, Konstan JA (2015) The MovieLens datasets: History and context. ACM Transactions on Interactive Intelligent Systems 5(4):19:1–19:19. https://doi.org/10.1145/2827872

[18] He X, Deng K, Wang X, et al (2020) 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. ACM, pp 639–648, https://doi.org/10.1145/3397271.3401063

[19] Kamishima T, Akaho S, Asoh H, et al (2018) Recommendation independence. In: Proceedings of the 1st Conference on Fairness, Accountability and Transparency. PMLR, pp 187–201

[20] Kim D, Park J, Kim D (2023) Test-time embedding normalization for popularity bias mitigation. In: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM), pp 4023–4027, https://doi.org/10.1145/3583780.3615281

[21] Lin S, Gao C, Chen J, et al (2025) How do recommendation models amplify popularity bias? an analysis from the spectral perspective. In: Proceedings of the 18th ACM International Conference on Web Search and Data Mining (WSDM), https://doi.org/10.1145/3701551.3703579

[22] Liu W, Burke R (2018) Personalizing fairness-aware re-ranking. In: Workshop on Recommendation in Multi-stakeholder Environments (RMSE @ RecSys)

[23] Liu Z, Xu X, Yu J, et al (2024) A self-adaptive fairness constraint framework for industrial recommender system. In: Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM). ACM, https://doi.org/10.1145/3627673.3680099

[24] Mansoury M, Abdollahpouri H, Pechenizkiy M, et al (2020) Feedback loop and bias amplification in recommender systems. In: Proceedings of the 29th ACM International Conference on Information and Knowledge Management. ACM, pp 2145–2148, https://doi.org/10.1145/3340531.3412092

[25] Marras M, Boratto L, Ramos G, et al (2022) Regulating group exposure for item providers in recommendation. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR). ACM, https://doi.org/10.1145/3477495.3531760

[26] Morik M, Singh A, Hong J, et al (2020) Controlling fairness and bias in dynamic learning-to-rank. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), pp 429–438,

https://doi.org/10.1145/3397271.3401100

[27] Naeimi M, Haghir Chehreghani M (2026) PBiLoss: Popularity-aware regularization to improve fairness in graph-based recommender systems. Applied Soft Computing 200. https://doi.org/10.1016/j.asoc.2026.115449, preprint: arXiv:2507.19067

[28] Qin T, Liu TY, Li H (2010) A general approximation framework for direct optimization of information retrieval measures. Information Retrieval 13(4):375–397. https://doi.org/10.1007/s10791-009-9124-x

[29] Santos RLT, Macdonald C, Ounis I (2010) Exploiting query reformulations for web search result diversification. In: Proceedings of the 19th International Conference on World Wide Web. ACM, pp 881–890, https://doi.org/10.1145/1772690.1772780

[30] Singh A, Joachims T (2018) Fairness of exposure in rankings. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pp 2219–2228, https://doi.org/10.1145/3219819.3220088

[31] Singh A, Joachims T (2019) Policy learning for fairness in ranking. In: Advances in Neural Information Processing Systems (NeurIPS)

[32] Steck H (2018) Calibrated recommendations. In: Proceedings of the 12th ACM Conference on Recommender Systems. ACM, pp 154–162, https://doi.org/10.1145/3240323.3240372

[33] Taylor M, Guiver J, Robertson S, et al (2008) SoftRank: Optimizing non-smooth rank metrics. In: Proceedings of the International Conference on Web Search and Data Mining (WSDM), pp 77–86, https://doi.org/10.1145/1341531.1341544

[34] Togashi R, Abe K, Saito Y (2024) Scalable and provably fair exposure control for large-scale recommender systems. In: Proceedings of the ACM Web Conference (WWW). ACM, https://doi.org/10.1145/3589334.3645390

[35] Wang X, He X, Cao Y, et al (2019) KGAT: Knowledge graph attention network for recommendation. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, pp 950–958, https://doi.org/10.1145/3292500.3330989

[36] Wang X, He X, Wang M, et al (2019) Neural graph collaborative filtering. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, pp 165–174, https://doi.org/10.1145/3331184.3331267

[37] Wei T, Feng F, Chen J, et al (2021) Model-agnostic counterfactual reasoning for eliminating popularity bias in recommender systems. In: Proceedings of the 27th

ACM SIGKDD Conference on Knowledge Discovery and Data Mining. ACM, pp 1791–1800, https://doi.org/10.1145/3447548.3467289

[38] Wu J, Wang X, Feng F, et al (2021) Self-supervised graph learning for recommendation. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, pp 726–735, https://doi.org/10.1145/3404835.3462862

[39] Xie Y, Dai H, Chen M, et al (2020) Differentiable top-k with optimal transport. In: Advances in Neural Information Processing Systems (NeurIPS)

[40] Xu C, Li Y, Wang W, et al (2025) Bridging jensen gap for max-min group fairness optimization in recommendation. In: Proceedings of the International Conference on Learning Representations (ICLR), method referred to as FairDual; arXiv:2502.09319

[41] Yang C, Lu K, He J, et al (2026) Dual path fairness optimization for graph neural network based recommendation. Scientific Reports 16. https://doi.org/10.1038/s41598-025-32364-5

[42] Yu J, Yin H, Xia X, et al (2022) 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. ACM, pp 1294–1303, https://doi.org/10.1145/3477495.3531937

[43] Yu J, Xia X, Chen T, et al (2024) XSimGCL: Towards extremely simple graph contrastive learning for recommendation. IEEE Transactions on Knowledge and Data Engineering 36(2):913–926. https://doi.org/10.1109/TKDE.2023.3288135

[44] Zafar MB, Valera I, Rodriguez MG, et al (2017) Fairness constraints: Mechanisms for fair classification. In: Proceedings of the 20th International Conference on Artificial Intelligence and Statistics. PMLR, pp 962–970

[45] Zehlike M, Bonchi F, Castillo C, et al (2017) FA*IR: A fair top-k ranking algorithm. In: Proceedings of the 2017 ACM Conference on Information and Knowledge Management. ACM, pp 1569–1578, https://doi.org/10.1145/3132847.3132938

[46] Zhang B, Hu Q, Sun M, et al (2025) Learning to rank with top-K fairness. Transactions on Machine Learning Research ArXiv:2509.18067

[47] Zhao Z, Chen J, Zhou S, et al (2023) Popularity bias is not always evil: Disentangling benign and harmful bias for recommendation. IEEE Transactions on Knowledge and Data Engineering https://doi.org/10.1109/TKDE.2022.3218994

[48] Zheng Y, Gao C, Li X, et al (2021) Disentangling user interest and conformity for recommendation with causal embedding. In: Proceedings of the Web Conference

2021. ACM, pp 2980–2991, https://doi.org/10.1145/3442381.3449788

[49] Zhu Z, He Y, Zhao X, et al (2021) Popularity-opportunity bias in collaborative filtering. In: Proceedings of the 14th ACM International Conference on Web Search and Data Mining. ACM, pp 85–93, https://doi.org/10.1145/3437963.3441820

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- esm.pdf