UDC: 004.8: [004.421:004.65] (045) https://doi.org/10.15407/jai2026.02.143

A. Shyraliiev[1], **I. Pyshnograiev**[2]
[1,2]National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ukraine
Beresteiskyi Ave, 37, Kyiv, 03056
[1]anarshyraliiev@gmail.com
[2]pyshnograiev@gmail.com
[1]https://orcid.org/0009-0007-4140-5476
[2]https://orcid.org/0000-0002-3346-8318

ARCHITECTURAL LIMITATIONS IN INTENSITY-AWARE SEQUENTIAL RECOMMENDATION: A NEGATIVE RESULT AND MECHANISM ANALYSIS

Abstract. Sequential recommenders based on self-attention discard a rich supervised signal: per-interaction intensity (a 5-star rating, hours played, dwell time, purchase value). We hypothesised that re-introducing intensity into the attention computation should improve top-k accuracy in proportion to how informative the underlying signal is. We test the hypothesis with three drop-in modifications of SASRec (IA-SASRec-Add, IA-SASRec-Mul, IA-SASRec-Val), corresponding to the three algebraic positions where intensity can be threaded into the attention expression. Each variant adds a single learnable scalar λ per attention layer, designed so that $\lambda = 0$ recovers vanilla SASRec exactly: the model is a strict super-set of the baseline. We evaluate across seven datasets (MovieLens×2, Amazon×2, Steam×3) with five seeds per configuration, under a dual-criterion significance protocol with a per-seed stability check (seed-level paired t and per-user paired bootstrap with $B = 2{,}000$). Under this protocol, no variant achieves a robust improvement on NDCG@10, Recall@10, or MRR@10 anywhere. The pattern inverts the hypothesis: Steam (the dataset with the richest intensity signal) hosts the largest significant degradations (up to -9.5% NDCG@10, $p < 0.05$). Mechanism analysis through the learned λ explains the failure: λ adapts freely on datasets where intensity is not informative (down to 0.07 on ML-1M) but stays at 0.55–0.99 on Steam, where the variants are at their worst. We argue that a single per-layer scalar gate is too limited a control for intensity injection, and outline three concrete designs that target this limitation.

Keywords: sequential recommendation, self-attention, SASRec, side information fusion, intensity-aware attention, reproducibility, negative result, mechanism analysis.

1 Introduction

Sequential recommenders treat a user's interaction history as a token stream, and self-attention models such as SASRec [1] and BERT4Rec [2] dominate the leaderboards. In the standard formulation, every interaction is collapsed to a binary "happened": item i_j was in the user's history at position j, and that is the totality of the signal that reaches the attention layer. The underlying intensity of each interaction (a 5-star rating on MovieLens, hours played on Steam, dwell time or purchase value in industrial logs) is discarded. From a Shannon standpoint this amounts to discarding substantial supervised information, and the discard is more notable the richer the intensity signal is: a binary signal carries one bit; a continuous heavy-tailed playtime distribution can carry several.

A natural hypothesis follows: re-introducing per-interaction intensity into the attention computation should help, and should help in proportion to how informative the underlying signal is. Steam playtime (continuous, heavy-tailed, several bits per interaction) should help more than discrete 1–5 ratings on MovieLens, whose entropy is bounded above by $\log_2 5 \approx 2.3$ bits and is much lower in practice given the well-known rating bias toward 4 and 5.

We test this hypothesis with three drop-in modifications of SASRec (IA-SASRec-Add, IA-SASRec-Mul, and IA-SASRec-Val), corresponding to the three algebraic positions in the attention expression where intensity can be threaded in (pre-softmax additive bias, pre-softmax multiplicative scale, post-softmax attention reweighting). Each variant adds exactly one learnable scalar λ per attention layer, initialised to 1.0, designed so that $\lambda = 0$ collapses the variant bit-for-bit to vanilla SASRec. The model is therefore a strict super-

set of the baseline: if intensity is not informative, a fair learner can recover SASRec exactly.

Across seven datasets, spanning the discrete-rating and continuous-playtime regimes, and five random seeds per (dataset, model) cell, the hypothesis fails. Not one of the 63 tested cells (3 variants × 3 metrics at $k = 10$ across 7 datasets) shows a robust improvement under our dual-criterion significance protocol: a seed-level paired t and a per-user paired bootstrap that must both reject, guarded by a per-seed CI sign-majority on the bootstrap. The single positive paired-t cell (IA-SASRec-Add on Amazon-Office MRR@10, $+3.6\%, p = 0.025$) is suppressed by the per-user bootstrap and the per-seed sign agreement. Worse, the degradation pattern is the opposite of the intensity-richness prediction. Steam, the dataset with the richest signal, hosts the largest and cleanest significant Losses (-9.5% to -7.3% NDCG@10/Recall@10/MRR@10 across Steam-3k Mul and Steam-15k Mul/Val, all robust).

The architecture was designed with an explicit fallback ($\lambda = 0$ recovers SASRec), so the failure has a sharp mechanism question attached, which we answer through an analysis of the learned per-layer λ. The learned λ does adapt: it drops to 0.07 on ML-1M Val (and the model recovers SASRec), so the optimisation path is open. But on Steam it stays at 0.55–0.99 across all three variants and all three subset sizes, even as test metrics degrade. The architecture provides a path to disable intensity, yet the optimiser does not take it. A single per-layer scalar is not a sufficient fallback for intensity injection: it cannot down-weight specific intensity values, and it cannot let some attention heads use intensity while others ignore it. It also confounds the signal itself with the way we encoded it: a flat λ could mean intensity carries no information, or that the chosen normalisation suppressed the information it contained.

Contributions:

- Three IA-SASRec variants (Add / Mul / Val) realised as a unified framework with one learnable scalar λ per layer, so that $\lambda = 0$ collapses each variant exactly to SASRec.
- A 7-dataset × 5-seed evaluation under a dual-criterion significance protocol (paired-t at the seed level and a per-user paired bootstrap with $B = 2{,}000$, guarded by a per-seed CI sign-majority on the bootstrap) explicitly designed against the failure modes flagged by prior reproducibility critiques [3, 4].
- A negative result with mechanism evidence: learned λ stays large precisely where the variant hurts the most, and the "off-switch at $\lambda = 0$" fallback is not realised in practice.

2 Related Work

Sequential attention recommenders.

The Transformer [5] brought self-attention to sequence modelling, and SASRec [1] adapted causal self-attention to next-item recommendation; it remains the canonical attention-based baseline. BERT4Rec [2] replaces the causal mask with a bidirectional masked-language-modelling objective in the spirit of BERT [6]. Both treat each historical interaction as a binary presence signal: the item identity and its position are everything the model sees. TiSASRec [7] is the closest prior work; it injects the relative time intervals between every pair of interactions in the sequence directly into the attention computation through a learned bias on the time delta. The architectural surface is the same as IA-SASRec, a learned modifier applied to attention scores, but the injected signal is different: TiSASRec injects temporal spacing, a structural feature of the sequence; IA-SASRec injects interaction strength, which varies across users for the same item-position pair. Earlier non-attention sequential recommenders (GRU4Rec [8], NARM [9], FPMC [10], Caser [11], NextItNet [12]) and the classic implicit-feedback ranker BPR [13] likewise make no use of per-interaction intensity at the architectural level.

Side information in sequence models.

FDSA [14] runs separate self-attention streams over item identities and item-side features (category, brand, attributes) and concatenates their outputs. S^3-Rec [15] brings side information in through pretraining objectives that mask attributes, items, and

segments. Both inject item-side metadata: features that are constant across the population for a given item. IA-SASRec instead injects an interaction-side signal: the same item appears with different intensities in different users' histories, so the injected quantity carries personalisation information that item-side features cannot. At the loss / sampling level, weighted matrix factorisation [16] and one-class collaborative filtering [17] use interaction strength to weight the training objective in non-sequential recommenders. Those approaches modify what is optimised, not what the attention layer sees; IA-SASRec is a strictly more invasive intervention.

Invasive vs. decoupled side-information fusion. A growing body of work has recognised that fusing auxiliary signals into standard self-attention is structurally fragile. NOVA [18] argues explicitly against invasive fusion: mixing side information into item embeddings or attention logits creates semantic interference and degrades the value path; it proposes using side information only to shape attention weights while keeping the value matrix purely semantic. DIF-SR [19] extends this critique to early fusion and shows that auxiliary signals require their own decoupled attention sub-spaces to be effective. CARCA [20] sidesteps self-attention injection entirely by querying context and attribute information through cross-attention, treating them as external key–value pairs rather than perturbations to the causal sequence. Dense routing alternatives reinforce the same point: HGN [21] introduces hierarchical instance and feature gating mechanisms rather than scalar gates, and UniSRec [22] routes heterogeneous textual features through a mixture-of-experts blender. SSE-PT [23] critiques vanilla SASRec's lack of explicit personalisation and injects user identity through stochastic shared embeddings. MEANTIME [24] similarly argues that a single attention head cannot absorb heterogeneous temporal signals and allocates separate heads to different temporal embedding types. By construction, IA-SASRec acts as an invasive, mid-attention, scalar-gated intervention. It sits precisely in the architectural region this literature flags as fragile, setting up a strict empirical test of whether a shared scalar gate can overcome the predicted semantic interference.

Attention pathologies and regularization. A parallel line of work argues that dot-product self-attention itself is an unreliable substrate for sequential recommendation. FMLP-Rec [25] replaces attention with a filter-enhanced MLP to dampen frequency noise; STOSA [26] replaces point-attention with stochastic Gaussian distributions and a Wasserstein-distance ranker; LightSANs [27] decomposes self-attention into a low-rank subspace for efficiency, and the resulting bottleneck also constrains the configurations the optimiser can exploit; and CORE [28] diagnoses a representation inconsistency between sequence encoding and the final dot-product ranker. A second strand of regularization tackles representation degeneration (embeddings collapsing into a narrow cone where small perturbations dominate) through contrastive learning over augmented or intent-clustered sequences [29–31], or through data augmentation alone [32]. These critiques are directly relevant to the design under test: if an invasive scalar-gated injection drives representation degeneration of the kind this literature identifies, the optimiser can settle into a configuration that maximises training-loss fit while damaging held-out generalisation, and a global scalar gate provides no gradient path back out.

Reproducibility and benchmarking critiques. Prior reproducibility critiques [3, 4] have shown that many neural recommender models do not beat well-tuned classical baselines [33] once the protocol is fixed across methods, and that single-seed reporting lets small effects survive significance testing on one seed only to disappear on the next. Sampled top-k metrics misrank methods systematically relative to full-vocabulary scoring [34], and even widely cited baselines such as BERT4Rec [2] are sensitive to implementation choices that go unreported in the original publications [35]. Our dual-criterion significance protocol is designed against exactly these failure modes: it requires both a seed-level

paired t-test and a per-user paired bootstrap to reject, with a per-seed sign-majority guarding against single-seed dominance, and we score against the full item vocabulary rather than a random negative pool. The IA-SASRec-Add Amazon-Office MRR@10 "win" we observe is a textbook example of the failure mode this protocol catches.

3 IA-SASRec: Intensity-Aware Self-Attention

3.1 Background: SASRec attention

SASRec [1] encodes a user's left-padded history $s = [i_1, \dots, i_T]$ (sequence length T) as embeddings $E \in \mathbb{R}^{T\times d}$, where d is the embedding dimension. Each transformer block projects E to queries, keys, and values $Q, K, V \in \mathbb{R}^{T\times d}$, and computes scaled dot-product self-attention with a causal mask $M \in \{0, -\infty\}^{T\times T}$:

$$\mathrm{Att}(Q,K,V) = \mathrm{softmax}\big(QK^\top/\sqrt{d} + M\big)V. \quad (1)$$

Only an item's identity (via the embedding lookup) and its position (via the positional embedding) enter equation (1); the strength of each interaction (a 5-star rating, hours played, or dwell time) is not used. IA-SASRec re-introduces that signal.

Let $w \in \mathbb{R}^T$ be the per-position intensity vector for the user's history (after the normalisation step described below). Define the intensity matrix $M_W \in \mathbb{R}^{T\times T}$ with $M_W[q,k] = w_k$, broadcasting intensity along the key axis, which is the axis the softmax normalises.

3.2 Three injection points

The attention expression (1) has three algebraic positions where M_W can be threaded in: as a pre-softmax additive bias, as a pre-softmax multiplicative scale, or as a post-softmax attention reweighting. We instantiate all three. Each adds one learnable scalar $\lambda \in \mathbb{R}$ per attention layer, initialised to 1.0, so that $\lambda = 0$ collapses the variant exactly to SASRec (1). This learnable gate is the central design discipline of IA-SASRec.

IA-SASRec-Add (additive logit bias). Intensity enters as an explicit bias on the raw attention logits:

$$\mathrm{Att}_{\mathrm{Add}} = \mathrm{softmax}\big(QK^\top/\sqrt{d} + \lambda M_W + M\big)V. \quad (2)$$

A high-intensity key k inflates column k of the logit matrix and is forced to receive large attention probability, irrespective of semantic similarity to the query. As $\lambda \to \infty$ the variant approaches a hard intensity-argmax.

IA-SASRec-Mul (multiplicative logit scaling). Instead of an additive bias, the semantic similarity itself is scaled by intensity:

$$\mathrm{Att}_{\mathrm{Mul}} = \mathrm{softmax}(QK^\top/\sqrt{d} \odot (1 + \lambda(M_W - 1)) + M)\,V, \quad (3)$$

with $\odot$ element-wise. The $(w_k - 1)$ parameterisation interpolates between two endpoints: at $\lambda = 0$ the multiplier is 1 and the variant collapses to SASRec; at $\lambda = 1$ the multiplier is exactly w_k, recovering the original hard-gatekeeper formulation. The causal mask is added after scaling so padded keys remain at $-\infty$.

IA-SASRec-Val (post-softmax attention reweighting). The standard softmax computes the attention distribution; intensity then reweights each key position's attention probability before the value mix. Equivalently, since the factor depends only on k, this is algebraically identical to pre-scaling each value vector V_k by its intensity:

$$(A \odot (1 + \lambda(M_W - 1)))\,V = A\,(D_\lambda V),$$

where $D_\lambda = diag(1 + \lambda(w - 1))$.

Hence the name Val: the operation acts at the value-mixing stage of attention.

$$\mathrm{Att}_{\mathrm{Val}} = [\,\mathrm{softmax}\big(QK^\top/\sqrt{d} + M\big) \odot (1 + \lambda(M_W - 1))]\,V. \quad (4)$$

Unlike Mul, which acts on raw logits and can be diluted by the softmax normalisation, Val acts directly on the attention probabilities that mix the values. We deliberately do not renormalise the reweighted distribution: any magnitude inflation introduced by $1 + \lambda(w_k - 1) > 1$ is absorbed by the post-attention layer normalisation [36] in the residual block, while the intensity-driven direction change in the output vector, the part that actually re-ranks candidates, passes through unaffected.

Equation (4) is a breaking change from a previous formulation of Val, which multiplied the post-attention context by query-position intensity. At prediction time only the last query is read out, so every candidate score was

rescaled by the same constant and the ranking was identical to vanilla SASRec: the variant could not reorder candidates and is not a meaningful test of intensity injection.

3.3 The λ fallback

At $\lambda = 0$ each variant equals SASRec bit-for-bit (verified by unit tests on the attention math). A fair learner that finds intensity not informative should therefore drive $\lambda \to 0$ and recover the baseline. This is the load-bearing prediction we test empirically: if the fallback holds, learned λ should be small wherever IA-SASRec fails to improve over SASRec, and large only where it helps.

3.4 Intensity normalisation

Raw intensities are heterogeneous: Steam hours range from 0.1 to 1000 +, while ratings live in $\{1, \ldots, 5\}$. Feeding raw values through a softmax causes severe value collapse, so normalisation is essential rather than cosmetic. We expose four modes as a single hyperparameter intensity_norm selected by Optuna [37]: log1p_minmax (default; per-user $\log(1 + w)$ then divide by the masked max), minmax (per-user with a floor $f = 0.1$ so the weakest real signal stays distinguishable from padding), zscore, and none (sanity-check ablation). The floor f is non-cosmetic: without it the least-intense real item maps to exactly 0 and becomes indistinguishable from padding, silently discarding one interaction per sequence in Mul and Val. After every mode, padding positions are forcibly zeroed.

3.5 Parameter cost

Each variant adds exactly n_{layers} learnable scalars total (2–3 in our configurations), strictly less than any prior side-information injection. Table 1 summarises the three formulations.

4 Experimental Setup

4.1 Datasets

Table 2 lists the seven datasets used. They are chosen to span two regimes of intensity signal: discrete, low-entropy explicit ratings (1–5) on MovieLens-100K and MovieLens-1M [38], Amazon Digital Music 5-core and Amazon Office Products 5-core [39, 40]; and continuous, heavy-tailed implicit hours-played on three Steam Reviews subsamples [41, 42] (3k, 8k, 15k users; fixed seed). All datasets are 5-core filtered ($|\text{interactions}_u|, |\text{interactions}_i| \geq 5$). The split is per-user chronological leave-one-out: the latest interaction forms the test target, the second-latest is held out as validation, the remainder is training.

4.2 Training and hyperparameter optimisation

Every (dataset, model) pair receives an identical hyperparameter budget: 25 Optuna [37] trials with the TPE sampler, optimising validation NDCG@10 with 10 epochs per trial (short HPO). The search space for SASRec and the three IA-SASRec variants is the same on the SASRec hyperparameters ($n_{\text{layers}} \in [1, 3]$, $n_{\text{heads}} \in \{1, 2, 4\}$, hidden size, inner size, dropouts, learning rate), with one extra categorical for IA-SASRec selecting intensity norm (log1p minmax, minmax, zscore, none). The learnable λ's are model parameters, not hyperparameters. The final fit uses each model's HPO-best configuration, trains for up to 50 epochs with early stopping on validation NDCG@10 (patience 5), and is repeated across five seeds. Test metrics come from a single pass using the best-on-validation checkpoint; validation data is never folded back into training.

This protocol gives the IA-SASRec variants strictly more hyperparameter freedom than SASRec (one extra categorical at no extra trial cost), and identical training budget per seed. Under this protocol, an under-tuning explanation would require a configuration outside the shared search space.

Table 1. The three IA-SASRec variants. λ is a learnable scalar per layer (init 1.0); $\lambda = 0$ collapses each variant to SASRec

Variant	*Mechanism*
IA-SASRec-Add	$\text{softmax}\left(\frac{QK^\top}{\sqrt{d}} + \lambda M_W + M\right)V$
IA-SASRec-Mul	$\text{softmax}\left(\frac{QK^\top}{\sqrt{d}} \odot (1 + \lambda(M_W - 1)) + M\right)V$
IA-SASRec-Val	$\left[\text{softmax}\left(\frac{QK^\top}{\sqrt{d}} + M\right) \odot (1 + \lambda(M_W - 1))\right]V$

Table 2. Datasets after 5-core filtering. Density is $|\mathcal{I}|/(|\mathcal{U}| \cdot |\mathcal{V}|)$. Intensity is the per-interaction signal threaded into IA-SASRec attention

Dataset	$\lvert\mathcal{U}\rvert$	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{I}\rvert$	*density*	*intensity*
ML-100K	943	1,682	100,000	6.305%	rating 1–5
ML-1M	6,040	3,706	1,000,209	4.468%	rating 1–5
Amazon-Music	5,541	3,568	64,706	0.327%	rating 1–5
Amazon-Office	4,905	2,420	53,258	0.449%	rating 1–5
Steam-3k	3,000	5,754	39,450	0.229%	hours played
Steam-8k	8,000	8,110	101,210	0.156%	hours played
Steam-15k	15,000	9,446	187,004	0.132%	hours played

4.3 Evaluation: dual-criterion significance with a per-seed stability check

We report NDCG@10, Recall@10, and MRR@10 at the standard $k = 10$ sequential-recommendation cutoff [1, 2]. Under our leave-one-out full-ranking protocol, where exactly one held-out positive per user is ranked against the full item catalogue, Hit@K coincides with Recall@K and Precision@K equals Recall@K/K, so the three metrics we report are the non-redundant trio (set-membership, reciprocal rank, log-discounted rank); Hit and Precision are computed but omitted as they carry no information beyond Recall.

A standard single-seed table is under-powered for small effect sizes: training stochasticity and test-population sampling are both large at the scale of these datasets [3, 4]. We report a metric difference as robust only when two independent hypothesis tests, each sensitive to a different source of variance, both reject at $\alpha = 0.05$, and a per-seed stability check confirms that the per-user test is not dominated by a single seed:

(i) Seed-level paired t-test on the $n = 5$ per-seed test metrics ($\text{df} = 4$, two-sided $\alpha = 0.05$). Unit of variation: random training seed. Pairing is by seed so the same data split sits on both sides of the comparison.

(ii) Per-user paired bootstrap [43] ($B = 2{,}000$ resamples) on aligned per-user metric differences pooled across seeds, with percentile 95% CI and two-sided $p = 2\min(\Pr[\bar{d} \leq 0], \Pr[\bar{d} \geq 0])$ floored at $1/B$. Unit of variation: test user.

(iii) Per-seed CI sign-agreement majority: a stability check, not an independent hypothesis test. We re-run the bootstrap of (ii) independently within each seed and require ≥ 3 of 5 per-seed 95% CIs to sit entirely on the same side of zero. This guards against (ii) being driven by a single lucky seed.

Cells in our main results table are bolded only when (i) and (ii) both reject at $\alpha = 0.05$ and the stability check holds. For full-vocabulary scoring rather than sampled top-k, see the cautions in [34].

4.4 Implementation and reproducibility

The model is built on top of RecBole [44] (PyTorch [45] back-end), with an additional intensity column threaded through the shared data pipeline so that baselines and intensity-

aware variants run on identical inputs. All artefacts needed to reproduce the analysis (code, checkpoints, per-seed evaluation outputs, and per-user metric tables) are released alongside the paper.

5 Results

Table 3 reports the relative change in NDCG@10, Recall@10, and MRR@10 of each IA-SASRec variant against the SASRec baseline, paired by seed. Stars indicate paired-t significance; cells in bold are robust under the dual-criterion protocol described in the previous section (paired t and per-user bootstrap both reject at $\alpha = 0.05$, with the per-seed sign-majority stability check holding). Figure 1 shows the per-user paired bootstrap CIs for NDCG@10 across every (dataset, variant) pair: nearly every confidence interval either crosses zero or sits to its left.

The headline is that across 63 tested cells (7 datasets × 3 variants × 3 metrics at $k = 10$), no positive cell survives the dual-criterion test with the per-seed stability check. Every robust cell is a significant degradation. Extending the same dual-criterion test to deeper cutoffs $k \in \{20, 50, 100\}$ for the same metric trio adds 189 further cells, 0 of which robustly reject positively; the null verdict is not an artefact of the $k = 10$ choice. We examine the pattern dataset by dataset.

MovieLens (ML-100K, ML-1M). Every cell is non-significant under paired-t at $\alpha = 0.05$. ML-100K shows numerically large drops (-5 to -13%) but with $n = 5$ seeds and the small per-user counts of a 943-user dataset the paired-t cannot reject. ML-1M is essentially flat: -0.9% to $+1.7\%$ relative change across all nine cells. Bounded discrete rating intensity
does nothing measurable, neither helping nor robustly hurting, consistent with the prior assumption that rating-as-side-signal is information-poor.

Amazon (Digital Music, Office Products). Mixed and small. The single positive paired-t cell in the entire campaign (IA-SASRec-Add on Amazon-Office MRR@10, $+3.6\%$, $p = 0.025$) is exactly what a single-seed protocol would publish as a win. Under our dual-criterion test with the per-seed stability check it does not survive: the per-user bootstrap or the per-seed sign majority fails, and the same configuration is non-significant on NDCG@10 ($+2.5\%$) and Recall@10 ($+1.4\%$). Mul on Amazon-Music degrades NDCG@10 (-2.7%, $p = 0.012$) and Recall@10 (-2.9%, $p = 0.021$); Val on Amazon-Office degrades Recall@10 robustly (-6.1%, $p = 0.005$, bold in Table 3). The Amazon cluster shows the value of a dual-criterion protocol with a stability check: it suppresses exactly the cell that would otherwise be the headline win.

Steam (3k / 8k / 15k). The hypothesis-killer cluster. Every variant degrades NDCG@10 across all three subset sizes. Steam-3k Mul is $-9.5\%^{*}$ NDCG@10 and $-10.7\%^{*}$ MRR@10, both robust; Steam-15k Mul lands at $-5.0\%^{**}$ / $-4.5\%^{**}$ / $-5.7\%^{*}$ across NDCG@10/Recall@10/MRR@10 and Steam-15k Val at $-6.2\%^{**}$ / $-5.1\%^{**}$ / $-7.3\%^{*}$: six robust degradations in a single dataset cluster. This is the opposite of what the intensity-richness hypothesis predicts: Steam carries the richest, most continuous interaction signal of the seven datasets, and is exactly where IA-SASRec hurts most.

6 Analysis of the degradation mechanism

The negative result documented above admits a more precise characterisation than a simple hypothesis failure. The variants are designed to be a strict super-set of SASRec: at $\lambda = 0$ each one collapses, layer-by-layer, to the baseline. A fair learner should therefore have driven λ toward zero whenever intensity was a hindrance and recovered SASRec exactly. It did not. This section makes the mechanism precise.

6.1 λ calibration

Table 4 reports the per-(dataset, variant) learned λ averaged across attention layers and 5 seeds, and Figure 2 plots it against the relative NDCG@10 change. Two facts dominate.

Table 3. Relative change (%) of IA-SASRec variants vs SASRec across 7 datasets and 5 seeds. Stars: paired t-test, $^{*}p < 0.05$, $^{**}p < 0.01$. Bold cells are robust: both the paired-t at the seed level and the per-user paired bootstrap (B=2000) reject at $\alpha = 0.05$, and the per-seed CI sign-majority stability check holds

Dataset	IA-SASRec-Add			IA-SASRec-Mul			IA-SASRec-Val		
	NDCG @10	Recall @10	MRR @10	NDCG @10	Recall @10	MRR @10	NDCG @10	Recall @10	MRR @10
ML-100K	-5.4%	-4.3%	-6.1%	-3.2%	-3.5%	-2.5%	-8.2%	-3.2%	-13.1%
ML-1M	+1.0%	+0.3%	+1.4%	-0.2%	-0.9%	+0.4%	+1.1%	+0.1%	+1.7%
Amazon-Music	-1.2%	**-3.9%***	+1.8%	-2.7%*	-2.9%*	-2.6%	+1.0%	-1.5%	+3.7%
Amazon-Office	+2.5%	+1.4%	+3.6%*	+0.9%	+2.0%	-0.1%	-2.3%*	-6.1%**	+1.9%
Steam-3k	-6.9%	-7.8%	-6.1%	**-9.5%***	-7.6%	**-10.7%***	-8.9%	-4.0%	-12.5%
Steam-8k	-1.4%	-0.1%	-2.5%	-1.9%	-2.4%	-1.3%	-3.5%	-2.6%	-4.3%
Steam-15k	-2.5%	-1.4%	-3.6%	**-5.0%****	**-4.5%****	**-5.7%***	**-6.2%****	**-5.1%****	**-7.3%***

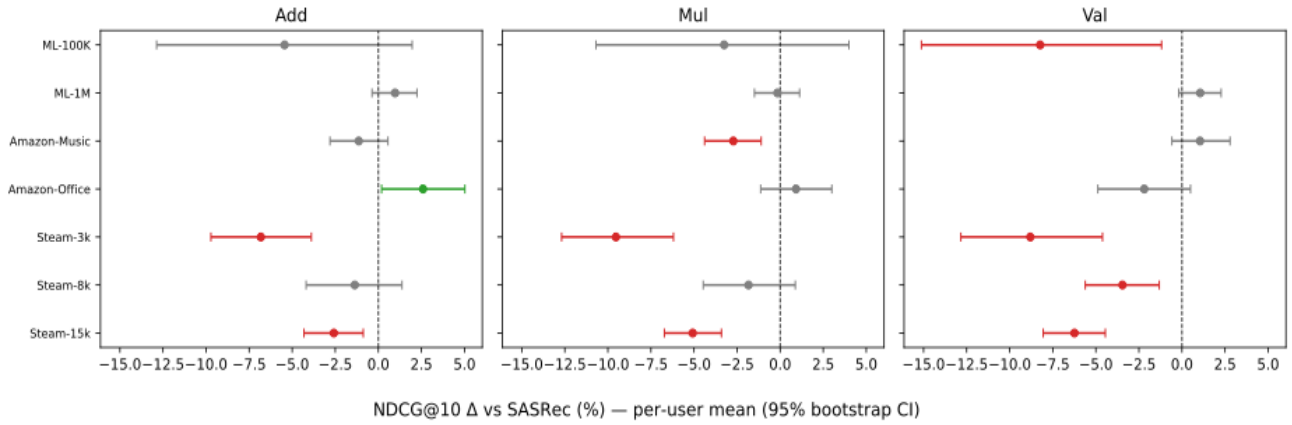

Fig. 1. Per-user paired bootstrap ($B = 2000$) on aligned NDCG@10 differences vs SASRec, pooled across five seeds. Three panels, one per variant; within each panel every row is one dataset and rows share the common Δ x-axis. Center = mean per-user Δ, whiskers = 95% percentile CI. Red markers have CI entirely below zero; green entirely above; grey cross zero. No robust positive cell exists for NDCG@10

λ does adapt away from its initialisation. The learned values span 0.07 to 0.99, two orders of magnitude of variation around the 1.0 initialisation. On ML-1M, Val drops to $\lambda \approx 0.07$ and Mul to $\lambda \approx 0.15$, and on ML-100K Mul reaches $\lambda \approx 0.21$. Wherever the gradient signal favours shrinking the intensity channel, the optimiser can do so: the gradient path is open.

λ stays high precisely where the variant hurts. On Steam-3k all three variants converge to $\lambda \in [0.98, 0.99]$ (effectively at the $\lambda = 1$ initialisation), and on Steam-8k and Steam-15k they sit at 0.56–0.75. These are exactly the cells with robust negative deltas in Table 3. The "fallback" that the design promised is not realised in practice: the optimiser does not steer λ toward 0 when intensity is harmful. There is no positive correlation between λ and Δ in Figure 2; if anything, the slope is negative.

The two observations together rule out the simple counter-narrative "the optimiser is stuck at $\lambda = 1$". It is not stuck; it moves freely on ML-1M and ML-100K. On Steam it makes an active choice to keep λ large, presumably because per-step cross-entropy loss locally rewards the intensity-injected attention pattern even though the resulting model generalises worse on the held-out test ranking. We did not run a controlled $\lambda \equiv 0$ ablation (the training-budget discipline did not allow an extra pinned-λ run);

Table 4. Learned λ per (dataset, variant), averaged over attention layers and 5 seeds (mean±std). Values are the per-layer learnable scalar gating the intensity signal; $\lambda = 0$ collapses to SASRec

Dataset	IA-SASRec-Add	IA-SASRec-Mul	IA-SASRec-Val
ML-100K	0.61±0.19	0.21±0.06	0.86±0.02
ML-1M	0.43±0.07	0.15±0.02	0.07±0.01
Amazon-Music	0.52±0.06	0.34±0.54	0.63±0.09
Amazon-Office	0.72±0.03	0.99±0.11	0.74±0.03
Steam-3k	0.98±0.01	0.99±0.00	0.98±0.01
Steam-8k	0.75±0.05	0.75±0.02	0.69±0.03
Steam-15k	0.68±0.04	0.73±0.04	0.56±0.04

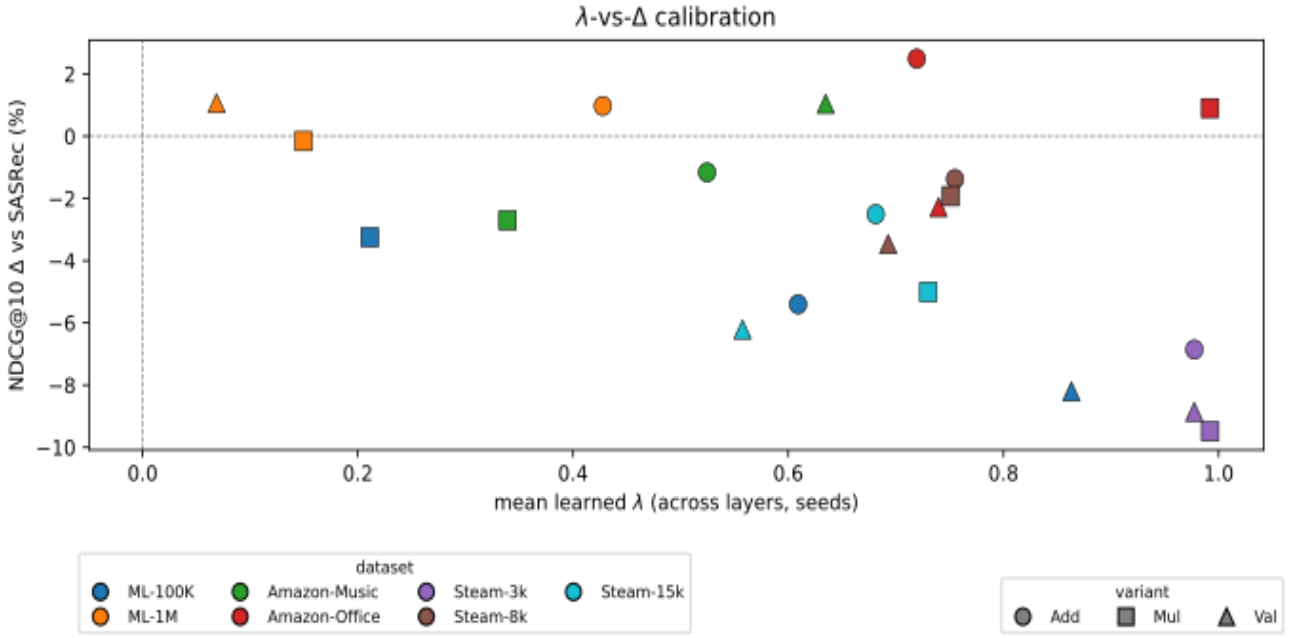

Fig. 2. λ-vs-Δ calibration: each marker is one (dataset, variant) cell. x-axis = mean learned λ across attention layers and seeds (Table 4); y-axis = mean test-NDCG@10 relative Δ vs SASRec. Top-right would be the predicted "model uses intensity and gains"; observed: empty. Bottom-right is "model uses intensity and loses", where Steam clusters

the falsification here therefore rests on the observed dynamics rather than a paired counterfactual. We return to this caveat in the Discussion.

6.2 The Steam paradox

Steam has the strongest, most continuous intensity signal of the seven datasets, yet hosts the worst outcomes. Two non-exclusive explanations are consistent with the data. Distribution shape: Steam playtime is heavy-tailed (a 1000 -hour outlier coexisting with many ~ 1-hour interactions). Under the default log-min-max normalisation the mass collapses into a narrow upper band, so the intensity channel carries less information than its raw range suggests while still adding optimisation variance. Popularity confound: high-playtime titles are also popular ones, and item-embedding learning already captures popularity; the intensity channel then adds redundancy rather than information. A definitive diagnosis between the two would require a per-dataset Spearman correlation between intensity rank and popularity rank and a histogram of post-normalisation intensities; we flag this as future analytical work, not a fix.

6.3 Implications for IA-SASRec design

A single learnable scalar λ per layer is too limited a control to be a useful fallback. Three structural shortcomings emerge from the analysis: (i) λ cannot down-weight a specific tail of the intensity distribution (e.g. mute the 100-hour outliers but keep the moderate range); (ii) λ is shared across attention heads in our implementation, so it cannot let some heads use intensity while others ignore it; (iii) the choice of normalisation is entangled with the architectural claim: a null result could mean intensity carries no information, or that the normalisation we picked flattened the information that was there. Future intensity-injection designs should target these three axes directly; the Conclusion outlines three concrete alternatives.

7 Discussion

7.1 Empirical validation of the decoupled-fusion hypothesis

Our results directly confirm the structural warnings raised by the decoupled-fusion literature (NOVA [18], DIF-SR [19]). Because IA-SASRec forces an invasive, mid-attention fusion through a shared scalar gate, it sits in the architectural region these works predict will fail. The empirical verdict matches the prediction: across the 252 (variant, metric, dataset, cutoff) cells we test (3 variants × 3 metrics × 7 datasets at $k \in \{10, 20, 50, 100\}$), not a single cell yields a robust improvement, and the largest, cleanest degradations appear on Steam, the dataset with the strongest and most continuous intensity signal, where representation degeneration of the kind flagged by [26, 30] is most likely to bite.

7.2 Intensity is not free supervision

Much of the side-information literature for sequential recommenders operates under an implicit additivity assumption: any extra channel threaded into the model can only add information, since the model "can ignore it." Our results rule out that assumption for per-interaction intensity under scalar gating. Once the optimiser has settled into a non-zero λ, the gradient with respect to training-set cross-entropy does not point back toward $\lambda = 0$, even when held-out generalisation degrades. The channel carries negative information in the sense that fitting it as instructed makes the model worse. "The model can ignore it" is a claim about expressivity, not about what the optimiser will actually do.

7.3 Lesson for benchmarking practice

Under a single-seed protocol without per-user confidence intervals, the headline of this paper would have been "IA-SASRec-Add improves MRR@10 on Amazon-Office Products by 3.6% ($p = 0.025$)." The same configuration is non-significant on the other two metrics of the same dataset and on every other dataset, but no single-seed table would have surfaced that. Our dual-criterion protocol flags the cell as non-robust automatically; this is exactly the failure mode that prior reproducibility critiques [3, 4] have argued is common in neural recommender benchmarking. We consider this a working example of the protocol doing its job, and we argue that it should be the default rather than the exception when reporting small-sample sequential-recommender results.

7.4 Threats to validity

HPO budget. We use the same 25-trial Optuna budget for SASRec and for each IA-SASRec variant, with the same backbone search space and an extra categorical for the normalisation choice on IA-SASRec. The variants therefore receive strictly more hyperparameter freedom than SASRec at no extra cost. The selected configurations are not systematically deeper or wider for SASRec than for the variants, so "under-tuned" is not a plausible explanation of the degradations.

Normalisation family. Optuna selects a single normalisation per (dataset, model). A richer parametric normalisation (e.g. learned bin boundaries on the playtime quantile) might rescue Steam; we leave this to future work and flag it as the most likely route to "intensity helps after all" on heavy-tailed signals.

No controlled $\lambda = 0$ run. We did not pin $\lambda = 0$ as a control, so the falsification of the fallback in our analysis rests on observed dynamics (large learned λ coexists with degraded metrics; the optimiser can drive λ small elsewhere). It does not strictly exclude the counter-narrative " $\lambda = 0$ would also have degraded on Steam"; we phrase the lesson in the Conclusion accordingly.

Conservatism of paired-t at $n = 5$. With $df = 4$ the paired-t has low power. We mitigate by also requiring the per-user bootstrap (effective n in the thousands) and the per-seed sign-majority gate. The cost is that some small-magnitude positive effects may be missed; the benefit is that the cells we do flag as robust are unlikely to be artifacts of seed or test-population sampling.

8 Conclusion

We tested three drop-in intensity-injection variants of SASRec (Add / Mul / Val), each gated by a learnable per-layer scalar λ initialised to 1.0 so that $\lambda = 0$ collapses the variant exactly to the baseline. Across seven datasets and five seeds under a dual-criterion significance protocol with a per-seed stability check, none of the variants delivers a robust improvement, and the failure is largest on Steam, the dataset with the strongest, most continuous intensity signal.

The mechanism admits a more precise characterisation than a simple hypothesis failure. The learned λ adapts freely on the datasets where intensity is not informative (down to 0.07 on ML-1M Val) yet stays at $0.55 – 0.99$ on Steam, where the variants are at their worst. The architectural off-switch predicted at $\lambda = 0$ is therefore available in principle but not realised in practice: cross-entropy gradients do not steer λ back to zero even when generalisation degrades. A single per-layer scalar is too limited a control for intensity injection.

Three concrete directions merit investigation as alternatives to scalar λ gating. First, token-level intensity gating: replace the scalar λ with a per-position learned gate $\lambda(w_k)$, so the model can mute specific intensity values (e.g. the 100-hour Steam tail) while keeping the moderate range. Second, mixture-of-experts on intensity bins: discretise intensity by post-normalisation percentile and route attention through expert heads selected by bin, decoupling "head specialised on high intensity" from "head ignoring intensity." Third, intensity as a side-token: rather than modifying the attention computation, insert an intensity-derived token at each position of the sequence and let standard self-attention decide whether to look at it; this moves the optimisation signal into the embedding space where existing SASRec dynamics are well understood.

References

1. Kang, W.-C., & McAuley, J. (2018). Self-attentive sequential recommendation. *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, 197–206. https://doi.org/10.1109/ICDM.2018.00035
2. Sun, F., Liu, J., Wu, J., Pei, C., Lin, X., Ou, W., & Jiang, P. (2019). BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*, 1441–1450. https://doi.org/10.1145/3357384.3357895
3. Ferrari Dacrema, M., Cremonesi, P., & Jannach, D. (2019). Are we really making much progress? A worrying analysis of recent neural recommendation approaches. *Proceedings of the 13th ACM Conference on Recommender Systems (RecSys)*, 101–109. https://doi.org/10.1145/3298689.3347058
4. Sun, Z., Yu, D., Fang, H., Yang, J., Qu, X., Zhang, J., & Geng, C. (2020). Are we evaluating rigorously? Benchmarking recommendation for reproducible evaluation and fair comparison. *Proceedings of the 14th ACM Conference on Recommender Systems (RecSys)*, 23–32. https://doi.org/10.1145/3383313.3412489
5. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*, 5998–6008. https://doi.org/10.48550/arXiv.1706.03762
6. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of NAACL-HLT 2019*, 4171–4186. https://doi.org/10.18653/v1/N19-1423
7. Li, J., Wang, Y., & McAuley, J. (2020). Time interval aware self-attention for sequential recommendation. *Proceedings of the 13th International Conference on Web Search and Data Mining (WSDM)*, 322–330. https://doi.org/10.1145/3336191.3371786

8. Hidasi, B., Karatzoglou, A., Baltrunas, L., & Tikk, D. (2016). Session-based recommendations with recurrent neural networks. *International Conference on Learning Representations (ICLR)*.
https://doi.org/10.48550/arXiv.1511.06939

9. Li, J., Ren, P., Chen, Z., Ren, Z., Lian, T., & Ma, J. (2017). Neural attentive session-based recommendation. *Proceedings of the 2017 ACM Conference on Information and Knowledge Management (CIKM)*, 1419–1428.
https://doi.org/10.1145/3132847.3132926

10. Rendle, S., Freudenthaler, C., & Schmidt-Thieme, L. (2010). Factorizing personalized Markov chains for next-basket recommendation. *Proceedings of the 19th International Conference on World Wide Web (WWW)*, 811–820.
https://doi.org/10.1145/1772690.1772773

11. Tang, J., & Wang, K. (2018). Personalized top-n sequential recommendation via convolutional sequence embedding. *Proceedings of the 11th ACM International Conference on Web Search and Data Mining (WSDM)*, 565–573.
https://doi.org/10.1145/3159652.3159656

12. Yuan, F., Karatzoglou, A., Arapakis, I., Jose, J. M., & He, X. (2019). A simple convolutional generative network for next item recommendation. *Proceedings of the 12th ACM International Conference on Web Search and Data Mining (WSDM)*, 582–590.
https://doi.org/10.1145/3289600.3290975

13. Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2009). BPR: Bayesian personalized ranking from implicit feedback. *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI)*, 452–461.
https://doi.org/10.5555/1795114.1795167

14. Zhang, T., Zhao, P., Liu, Y., Sheng, V. S., Xu, J., Wang, D., Liu, G., & Zhou, X. (2019). Feature-level deeper self-attention network for sequential recommendation. *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, 4320–4326. https://doi.org/10.24963/ijcai.2019/600

15. Zhou, K., Wang, H., Zhao, W. X., Zhu, Y., Wang, S., Zhang, F., Wang, Z., & Wen, J.-R. (2020). S^3-Rec: Self-supervised learning for sequential recommendation with mutual information maximization. *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM)*, 1893–1902. https://doi.org/10.1145/3340531.3411954

16. Hu, Y., Koren, Y., & Volinsky, C. (2008). Collaborative filtering for implicit feedback datasets. *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM)*, 263–272. https://doi.org/10.1109/ICDM.2008.22

17. Pan, R., Zhou, Y., Cao, B., Liu, N. N., Lukose, R., Scholz, M., & Yang, Q. (2008). One-class collaborative filtering. *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM)*, 502–511. https://doi.org/10.1109/ICDM.2008.16

18. Liu, C., Li, X., Cai, G., Dong, Z., Zhu, H., & Shang, L. (2021). Noninvasive self-attention for side information fusion in sequential recommendation. *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, 4249–4256.
https://doi.org/10.1609/aaai.v35i5.16549

19. Xie, Y., Zhou, P., & Kim, S. (2022). Decoupled side information fusion for sequential recommendation. *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1611–1621.
https://doi.org/10.1145/3477495.3531963

20. Rashed, A., Elsayed, S., & Schmidt-Thieme, L. (2022). CARCA: Context and attribute-aware next-item recommendation via cross-attention. *Proceedings of the 16th ACM Conference on Recommender Systems (RecSys)*, 71–80. https://doi.org/10.1145/3523227.3546777

21. Ma, C., Kang, P., & Liu, X. (2019). Hierarchical gating networks for sequential recommendation. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)*, 825–833.
https://doi.org/10.1145/3292500.3330984

22. Hou, Y., Mu, S., Zhao, W. X., Li, Y., Ding, B., & Wen, J.-R. (2022). Towards universal sequence representation learning for recommender systems. *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (KDD)*, 585–593.
https://doi.org/10.1145/3534678.3539381

23. Wu, L., Li, S., Hsieh, C.-J., & Sharpnack, J. (2020). SSE-PT: Sequential recommendation via personalized transformer. *Proceedings of the 14th ACM Conference on Recommender Systems (RecSys)*, 328–337.
https://doi.org/10.1145/3383313.3412258

24. Cho, S. M., Park, E., & Yoo, S. (2020). MEANTIME: Mixture of attention mechanisms with multi-temporal embeddings for sequential recommendation. *Proceedings of the 14th ACM Conference on Recommender Systems (RecSys)*, 515–520.
https://doi.org/10.1145/3383313.3412216

25. Zhou, K., Yu, H., Zhao, W. X., & Wen, J.-R. (2022). Filter-enhanced MLP is all you need for sequential recommendation. *Proceedings of the ACM Web Conference (WWW)*, 2388–2399.
https://doi.org/10.1145/3485447.3512111

26. Fan, Z., Liu, Z., Wang, A., Nazari, Z., Zheng, L., Peng, H., & Yu, P. S. (2022). Sequential recommendation via stochastic self-attention. *Proceedings of the ACM Web Conference (WWW)*, 2036–2047.
https://doi.org/10.1145/3485447.3512077

27. Fan, X., Liu, Z., Lian, J., Zhao, W. X., Xie, X., & Wen, J.-R. (2021). Lighter and better: Low-rank decomposed self-attention networks for next-item recommendation. *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1733–1737.
https://doi.org/10.1145/3404835.3462440

28. Hou, Y., Hu, B., Zhang, Z., & Zhao, W. X. (2022). CORE: Simple and effective session-based

recommendation within consistent representation space. *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1796–1801. https://doi.org/10.1145/3477495.3531955

29. Xie, X., Sun, F., Liu, Z., Wu, S., Gao, J., Zhang, J., Ding, B., & Cui, B. (2022). Contrastive learning for sequential recommendation. *Proceedings of the 38th IEEE International Conference on Data Engineering (ICDE)*, 1259–1273. https://doi.org/10.1109/ICDE53745.2022.00099

30. Qiu, R., Huang, Z., Yin, H., & Wang, Z. (2022). Contrastive learning for representation degeneration problem in sequential recommendation. *Proceedings of the 15th ACM International Conference on Web Search and Data Mining (WSDM)*, 813–823. https://doi.org/10.1145/3488560.3498433

31. Chen, Y., Liu, Z., Li, J., McAuley, J., & Xiong, C. (2022). Intent contrastive learning for sequential recommendation. *Proceedings of the ACM Web Conference (WWW)*, 2172–2182. https://doi.org/10.1145/3485447.3512090

32. Liu, Z., Fan, Z., Wang, Y., & Yu, P. S. (2021). Augmenting sequential recommendation with pseudo-prior items via reversely pre-training transformer. *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1608–1612. https://doi.org/10.1145/3404835.3463036

33. Rendle, S., Krichene, W., Zhang, L., & Anderson, J. (2020). Neural collaborative filtering vs. Matrix factorization revisited. *Proceedings of the 14th ACM Conference on Recommender Systems (RecSys)*, 240–248. https://doi.org/10.1145/3383313.3412488

34. Krichene, W., & Rendle, S. (2020). On sampled metrics for item recommendation. *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)*, 1748–1757. https://doi.org/10.1145/3394486.3403226

35. Petrov, A., & Macdonald, C. (2022). A systematic review and replicability study of BERT4Rec for sequential recommendation. *Proceedings of the 16th ACM Conference on Recommender Systems (RecSys)*, 436–447. https://doi.org/10.1145/3523227.3548487

36. Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*. https://doi.org/10.48550/arXiv.1607.06450

37. Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)*, 2623–2631. https://doi.org/10.1145/3292500.3330701

38. Harper, F. M., & Konstan, J. A. (2015). The MovieLens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems*, *5*(4), 1–19. https://doi.org/10.1145/2827872

39. McAuley, J., Targett, C., Shi, Q., & Hengel, A. van den. (2015). Image-based recommendations on styles and substitutes. *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 43–52. https://doi.org/10.1145/2766462.2767755

40. He, R., & McAuley, J. (2016). Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. *Proceedings of the 25th International Conference on World Wide Web (WWW)*, 507–517. https://doi.org/10.1145/2872427.2883037

41. Pathak, A., Gupta, K., & McAuley, J. (2017). Generating and personalizing bundle recommendations on Steam. *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1073–1076. https://doi.org/10.1145/3077136.3080724

42. Wan, M., & McAuley, J. (2018). Item recommendation on monotonic behavior chains. *Proceedings of the 12th ACM Conference on Recommender Systems (RecSys)*, 86–94. https://doi.org/10.1145/3240323.3240369

43. Efron, B., & Tibshirani, R. J. (1993). *An introduction to the bootstrap*. Chapman & Hall/CRC. https://doi.org/10.1201/9780429246593

44. Zhao, W. X., Mu, S., Hou, Y., Lin, Z., Chen, Y., Pan, X., Li, K., Lu, Y., Wang, H., Tian, C., Min, Y., Feng, Z., Fan, X., Chen, X., Wang, P., Ji, W., Li, Y., Wang, X., & Wen, J.-R. (2021). RecBole: Towards a unified, comprehensive and efficient framework for recommendation algorithms. *Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM)*, 4653–4664. https://doi.org/10.1145/3459637.3482016

45. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., … Chintala, S. (2019). PyTorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems (NeurIPS)*, 8024–8035. https://doi.org/10.48550/arXiv.1912.01703

The article has been sent to the editors 03.06.26.
After processing 05.06.26.
Submitted for printing 30.06.26