ITEM CONTENT HELPS THE LONG TAIL BUT HURTS POPULAR ITEMS IN SEQUENTIAL RECOMMENDATION UNDER A LEXICAL ENCODER: FREQUENCY-GATED CONTENT INJECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Item content is widely used to help sequential recommenders rank long-tail items, yet its effect on popular items is rarely isolated. On three Amazon Reviews 2023 categories, uniformly injecting item titles and categories into a SASRec backbone through a fixed lexical content encoder raises long-tail Recall@10 by 10.6–17.8 points but lowers head Recall@10 by 1.7–4.5 points. This head cost is specific to the lexical encoders we test: with a frozen semantic encoder at 768 dimensions, it disappears in all three categories. Frequency-conditioned fusion weights have been explored in recent multimodal and long-tail recommenders; we study a minimal single-scalar form, Frequency-Gated Content Injection (FGCI), where a per-item scalar α_i scales the content embedding and is learned jointly with the backbone from the item's frozen, standardized log interaction count. FGCI improves head Recall@10 over the no-content baseline in all three categories, while long-tail Recall@10 stays 8.4–15.8 points above it. A natural explanation is that FGCI simply injects more content; we rule this out. A constant scale fixed a priori at the head-item level misses FGCI by 5.1–6.0 points (unpaired $p < 0.01$ in all three categories). Nor can the gate be fitted after the backbone has been trained. An ablation on Health and Household, one of the three categories, shows the same for training-time gating alone: it drives head Recall@10 23.8 points below the no-content baseline. Together, these controls show that the gain is not reproduced by any unconditional, post-hoc, or single-sided variant we test. What, then, does the gate learn? In all three categories the learned α_i is largest for head items and smallest for long-tail items, inverting the intuition that the items with the least interaction history need the most content. That inversion does not explain the recovery either: head items keep 85–88% of their content weight under the gate, yet a constant scale at that value does not recover head performance at all. This rules out the most natural static-weight reading of the gate and indicates that the recovery requires conditioning to be per item and learned jointly, rather than residing in the specific gate values. Why this conditioning helps remains an open question that our controls have narrowed but not resolved.

1 INTRODUCTION

A sequential recommender is usually evaluated by its average performance over the whole catalogue, and by that measure, the case for item content appears settled. Long-tail items have too few interactions for co-occurrence alone to rank them well. Their titles and categories are available regardless of how many users have interacted with them, and fusing this text into a sequence encoder improves aggregate accuracy (Zhang et al., 2019; Zhou et al., 2020; Hou et al., 2022; Li et al., 2023). Content-enhanced sequential recommendation is now widely used, and the gains it reports are real.

These gains, however, are unevenly distributed. We measure what unconditional content injection does to each part of the popularity distribution, rather than to the average. On three categories of Amazon Reviews 2023, adding item titles and categories to a SASRec backbone (Kang & McAuley, 2018) raises Recall@10 on the long-tail stratum by 10.6 to 17.8 percentage points, but *lowers* Re-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

call@10 on the head stratum—the popular items the backbone already ranks well—by 1.7 to 4.5 percentage points. The effect reverses, and it reverses the same way in all three categories (Section 4.3). This head cost appears specific to a fixed lexical content encoder: with a semantic encoder at dimension 768 it disappears in all three categories, and in two of the three the head gain turns positive (Section 4.4).

This reversal sharpens the design question. If one operating point helps the tail and hurts the head, then the question is not whether to inject content, but how much each item should receive. A single global weight cannot answer this: the two strata require opposite settings. Nor is a better constant enough (Section 4). What is needed is a weight that varies per item.

A per-item content weight is not new: recent long-tail and multimodal recommenders condition it on item popularity (Wang et al., 2025; Zhang et al., 2026) or on the sequence context (Glisovic et al., 2026). Our goal is not to propose a new fusion architecture: we adopt a minimal single-scalar form, Frequency-Gated Content Injection (FGCI), precisely so that its gain can be isolated from the design. Each item i carries a scalar $\alpha_i \in (0, 1)$ produced by a small gate network from a frozen, standardized log-count of the item's interaction count. The item representation becomes $\mathrm{rep}(i) = \mathrm{item_emb}(i) + \alpha_i P(c_i)$, where P projects the item's content features c_i. The gate is trained jointly with the backbone and applied at both training and inference. The design is intentionally minimal: one scalar per item, one input, and no change to the backbone architecture.

FGCI recovers the head while keeping most of the tail. Head Recall@10 rises above the no-content baseline in all three categories (+1.24, +1.74 and +4.06 percentage points), and long-tail Recall@10 remains 8.4 to 15.8 percentage points above that baseline, 1.9 to 2.9 percentage points below what unconditional injection reaches.

We then ask what produces this recovery, and the answer is not obvious. Two controls narrow it, and a third observation sharpens the puzzle. First, the recovery is not a dose effect: constant-scale controls fixed a priori at 0.85 fall back to the unconditional level in every category, missing FGCI by 5.1 to 6.0 percentage points. Second, neither side of the gate is sufficient: gating only during training drives head Recall@10 23.8 percentage points below the no-content baseline, and fitting a gate afterwards on a frozen unconditional backbone also fails to recover the head. Third, the gate's learned behavior contradicts our initial expectation. The head stratum's mean α is the largest of the three strata, reversing the pattern one would expect if the least-interacted items were the ones that need content most. That reversal does not account for the recovery either. Head items retain most of their content weight under the gate, yet constant-scale controls, which apply a single value to every item, recover nothing. The gain therefore cannot be attributed to a global content scale; it requires conditioning that is per item and learned jointly with the backbone. The mechanism behind the recovery remains open: the controls rule out dose and single-sided conditioning, but leave the mechanism itself unresolved.

These results define the scope of our claims. Every experiment uses a SASRec backbone and title-and-category content, the latter encoded by a fixed bag-of-words hash; the encoder is held identical across all arms, so that differences between arms are attributable to the injection alone. We do not benchmark against existing content-based models: the backbone, data, and content are held fixed and only the injection is varied, so that its effect can be read without confounding from the encoder or the scoring function. The full ablation chain that isolates the conditioning requirement runs on one of the three categories, with the cross-category evidence covering three arms.

Section 2 positions the work against content-enhanced and long-tail recommendation. Section 3 defines FGCI and the ablation arms. Section 4 reports the experiments, and Section 5 discusses what the controls leave open.

Contributions.

- A stratified, paired measurement protocol for content injection. It zeroes the content term in the same backbone, on the same seed and against the same negatives, and reports the signed per-stratum difference, exposing an effect that the aggregate hides: on three Amazon Reviews 2023 categories, under our lexical encoder, content helps the long tail by 10.6 to 17.8 points while costing popular items 1.7 to 4.5.

- A minimal single-scalar realization of frequency-conditioned injection—a 49-parameter gate—that recovers head Recall@10 in all three categories while retaining most of the long-tail gain.
- An isolation of the conditions under which the recovery appears. Removing either side of the gate—training-time or inference-time—removes it, and constant-scale controls do not reproduce it.
- A constraint on what the gate can be doing. The recovery cannot be read off the gate's learned values, nor reduced to a static reweighting of the same content, even at the gate's own average value; the values themselves reverse the pattern the literature would predict.

2 Related Work

2.1 Text-enhanced sequential recommendation

Item text enters sequential recommendation along three routes, which differ in where the text is fused and how much of the encoder it replaces. One keeps item IDs and fuses attribute or text features into the item representation, from attribute-level transition modeling (Zhang et al., 2019) to self-supervised attribute signals (Zhou et al., 2020). A second replaces IDs with text as the item identifier (Ding et al., 2021; Hou et al., 2022; Li et al., 2023): Hou et al. (2022) freeze a language model offline and train a projection on top, while Li et al. (2023) train the encoder jointly with the sequence model at far higher cost. A third folds item text into prompts or language-model embeddings (Geng et al., 2022; Liao et al., 2024). Where fusion is placed matters in its own right (Xie et al., 2022).

What is fused matters as well. Our own content representation is intentionally simple: a hashed bag of title and category tokens rather than a pretrained language-model embedding, and Section 4.4 quantifies why the head-level cost we report is a property of that choice rather than of content injection in general.

2.2 Long-tail items and popularity-conditioned fusion

Content is the standard remedy when history is thin: Volkovs et al. (2017) score cold items from content, and Liu et al. (2024) and Zhang et al. (2026) inject language-model-derived embeddings to help long-tail items. Popularity bias pushes the other way, since head items dominate the training signal and the popularity prior is a weak or inverted cue for rare targets (Abdollahpouri et al., 2019; Steck, 2011).

Beyond how content is represented, a second design axis is how much content each item receives. Wang et al. (2025) call their design multimodal frequency-aware fusion: a min-max standardized log count passes through an MLP that emits per-item weights over modality scoring terms. Zou et al. (2026) gate an LLM-derived channel against an ID channel using ID, language-model and frequency features, and report that the weight on the ID channel rises with interaction frequency. Glisovic et al. (2026) learn a per-item weight from the sequence context rather than from frequency, and compare it against constant scales, finding a per-item advantage of 3.4 to 6.6 percent over the best constant scale in each case on their two benchmarks.

2.3 What is not yet measured

These works establish that making the fusion weight per item—conditioned on frequency or on the sequence context—is beneficial. What they do not provide is a signed, per-stratum account of the trade-off. Zou et al. (2026) compare against competing methods within item-frequency groups, and Glisovic et al. (2026) study how visual utility varies with sparsity; both measure a model's advantage over other models, not what content contributes within one backbone. We therefore ask a question these works leave open: whether content helps or hurts a given stratum, and by how much, relative to the same backbone with the content term removed. Whether the gate must be learned jointly with the backbone, or can be fitted afterwards, is also unresolved.

The gate itself is not our contribution; our measurements are. We report a paired, stratum-wise difference against the same backbone with content zeroed, and an ablation separating training-time

from inference-time gating, which together isolate the conditions under which the gain appears. One difference in direction is worth flagging: Zou et al. (2026) report that the ID channel weight rises with interaction frequency, whereas the gate we study shifts toward content. Section 5 takes this up.

3 METHOD

3.1 SETUP AND NOTATION

Our goal in this section is to isolate the effect of content injection from everything else. To that end, we hold the backbone, the content matrix, and the data protocol fixed, and vary only how the content is combined with the backbone. Within a seed, every arm starts from identical initial weights, consumes the same negative-sampling stream, and is evaluated on the same held-out examples. The data protocol is described in Section 4.

Backbone. We use a two-layer SASRec (Kang & McAuley, 2018) with hidden size 64, two attention heads, dropout 0.2, pre-layer normalization, and maximum sequence length 50, trained with AdamW at a learning rate of 10^{-3} for 30 epochs, with no learning-rate schedule, no gradient clipping, and no early stopping. Each position is supervised by a sampled-softmax cross-entropy over one positive and 100 negatives.

Content. Each item i carries a vector $c_i \in \mathbb{R}^{128}$ built from its title and category. We tokenize both fields, hash each token to one of 128 bins with CRC32, accumulate token counts, and normalize every row to unit length. The encoder is deliberately shallow—a fixed lexical bag rather than a learned language representation—so that what we measure is the effect of the injection and not of the encoder; Section 4.4 repeats the comparison with a semantic encoder. The padding row and every item without tokens are all-zero rows, and because the content projection carries no bias, a zero row projects to exactly zero. An item without content therefore falls back to its pure ID representation, and the zeroed-content backbone serves as the reference for every arm.

Strata. We stratify by popularity, measured as an item's interaction count in the training split. Two stratifications appear in this paper, and they are not interchangeable. The first is used in metric tables and stratifies evaluation examples: each held-out target is placed in head, mid, or tail by the terciles of $\log(1+\text{count})$ over the target items of all evaluation examples. The second is used to summarize gate values and stratifies the item set: every item $i \neq 0$ receives equal weight and is placed by the terciles of its own $\log(1+\text{count})$. Metric tables stratify evaluation examples because Recall@10 is computed per example; gate values are summarized over items because α_i is an item property.

The two nest but do not coincide. On Health and Household, the item-level head contains 21185 items with a median count of 40, while the example-level head covers 2742 distinct items with a median count of 187; the corresponding numbers are $13810/37$ and $2015/141$ on Grocery, and $21060/33$ and $2873/134$ on Beauty. We qualify "head" as example-level or item-level throughout.

3.2 UNCONDITIONAL CONTENT INJECTION

We begin with the simplest way to use content, which adds it to every item. With a bias-free projection P into the backbone's hidden space, the item representation becomes

$$\text{rep}(i) = \text{item_emb}(i) + P(c_i). \tag{1}$$

As Section 4 reports, this moves the two ends of the popularity distribution in opposite directions: long-tail Recall@10 rises by 10.6 to 17.8 percentage points across the three categories, while head Recall@10 falls by 1.7 to 4.5 percentage points. A single global weight cannot serve both ends, since one small enough to spare the head is too small to help the tail; the weight has to vary with the item. The natural way to let it vary is a piecewise-constant weight by frequency bin. That form adds a bin count as a hyperparameter and introduces discontinuities at bin boundaries, so we take the continuous per-item alternative instead. The constant-scale arms of Section 4 are the degenerate one-bin member of that family: they probe the end where every item shares a weight, and what we ask is whether moving off that end does any work beyond a global rescaling.

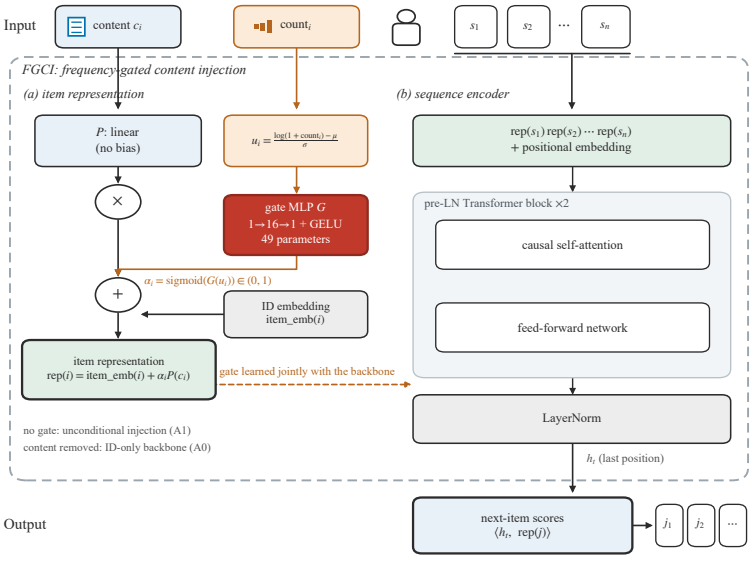

Figure 1: The FGCI injection point. A gate G ($1 \to 16 \to 1$ with GELU, 49 parameters) maps the standardized log-count u_i to $\alpha_i \in (0, 1)$, which scales the projected content $P(c_i)$ before it is added to the item embedding. Disabling the gate takes the same addition path as unconditional injection (A1); removing content gives the ID-only backbone (A0).

3.3 Frequency-gated content injection

A per-item weight needs a signal to condition on. We condition on interaction frequency, the variable that separates head from tail by construction. FGCI replaces the global weight with a per-item one,

$$\text{rep}(i) = \text{item_emb}(i) + \alpha_i P(c_i), \qquad \alpha_i = \text{sigmoid}(G(u_i)) \in (0, 1). \tag{2}$$

The gate reads a frozen scalar summary of the item's training-split interaction count,

$$u_i = \frac{\log(1 + \text{count}_i) - \mu}{\sigma}, \tag{3}$$

where μ and σ are the mean and population standard deviation of $\log(1+\text{count})$ over training items with at least one interaction. Items with no training interactions, padding included, receive $u_i = 0$, the population mean, so that the gate is not conditioned on an absent count; at most two items per category are affected. The gate network and the backbone are trained jointly; the summary u_i is held in a non-trainable buffer.

G is a two-layer MLP—one input scalar, a 16-unit hidden layer with GELU, one output logit—adding 49 parameters to the backbone. Its output layer is initialized with bias 4 and weight standard deviation 0.01, giving $\text{sigmoid}(4) \approx 0.982$ at initialization, so training starts close to unconditional injection rather than near zero.

The gate has separate switches for training and evaluation, and it is active when training and the training switch are both on, or evaluation and the evaluation switch are both on. With both switches on, the configuration is FGCI. With both off, the model takes the same addition path as unconditional injection; it does not multiply by a learned factor that happens to equal one, so the ablation compares mechanisms rather than near misses. Figure 1 summarises the injection point and the two switches.

3.4 Ablation arms

To test each of these design choices, we compare the following arms.

Each arm isolates one design choice: whether the gain comes from the conditioning signal or from the gate's extra capacity (A3), whether a gate fitted after training recovers it (A4), whether the inference side matters (A5), and whether a per-item weight is needed at all (U c).

Table 1: The ablation arms. All arms share one backbone initialization per seed and one negative-sampling stream; A0 zeroes the content matrix and is thus the ID-only model. A3 permutes the gate input across items once, before training, preserving its multiset and leaving the architecture and parameter count unchanged. A4 first trains the backbone with the gate off, then freezes it and fits only the gate for five epochs at a learning rate of 10^{-4}. The two constants were fixed before the main experiments to approximate the scale of the gate values measured in an earlier run under the same protocol, and are shared across all categories and encoders. The $c = 1$ endpoint coincides with A1 and is not listed separately.

Arm	Content term	Gate active	Purpose
A0 (reference)	0	—	No content
A1	$P(c_i)$	—	Unconditional injection
A2 (FGCI)	$\alpha_i P(c_i)$	train + eval	The method
A3	$\alpha_i P(c_i)$, permuted input	train + eval	Conditioning, or capacity?
A4	$\alpha_i P(c_i)$, backbone frozen	two-stage	Joint, or post-hoc?
A5	$\alpha_i P(c_i)$	train only	Both sides?
Uc (constant scale)	$c\,P(c_i)$, $c \in \{0.725, 0.85\}$	—	Constant, or per item?

Table 2: Recall@10 difference against the A0 baseline of the same seed, in percentage points; $n = 3$ seeds. U 0.85 is one fixed scale applied to every item in every domain; U 0.725 is reported in Table 3. Standard deviations (population, ddof$= 0$): unconditional head 1.06 / 0.03 / 0.86 and overall 0.34 / 0.58 / 0.56; FGCI head 0.25 / 0.73 / 0.93 and overall 0.38 / 0.85 / 0.65; U 0.85 head 0.78 / 0.09 / 1.01.

	unconditional (A1)			FGCI (A2)			constant
Dataset	all	head	tail	all	head	tail	U 0.85 head
Health	$+3.61$	-4.52	$+10.57$	$+5.98$	$\mathbf{+1.24}$	$+8.72$	-4.45
Grocery	$+6.66$	-3.14	$+11.29$	$+8.45$	$\mathbf{+1.74}$	$+8.40$	-3.34
Beauty	$+10.32$	-1.70	$+17.78$	$+12.49$	$\mathbf{+4.06}$	$+15.75$	-1.90

4 Experiments

Setup. We use three domains of the 2023 edition of the Amazon Reviews collection: Health_and_Household (62,727 items, 136,118 users), Grocery_and_Gourmet_Food (40,360 / 70,668) and Beauty_and_Personal_Care (61,737 / 101,396). Each domain is pruned to a 10-core over users and items; every user's last interaction is held out as the test target and ranked against 100 sampled negatives. We run seeds $2026/2027/2028$, report Recall@K and NDCG@K for $K \in \{10, 20\}$, and give every number as a difference against the A0 baseline of the same seed, so that all arm-to-arm comparisons are paired by seed. We compute every derived quantity—gains, repairs, and the differences between them—from unrounded values and round only for display, so a difference quoted in the text can differ from the difference of the two displayed entries by 0.01. The backbone, the content matrix and the stratifications are those of Section 3.

4.1 The gate repairs the head

On Health, where we ran the full arm chain of Table 1, adding content unconditionally (A1) buys $+3.61$ pp overall but costs the head stratum 4.52 pp (-2.49 pp on NDCG@10). Training the gate instead (A2) turns the head into $\mathbf{+1.24}$ pp and raises the overall gain to $+5.98$ pp, a $\mathbf{+5.75}$ pp head repair over A1 (the same quantity is $+5.69$ pp relative to U 0.85). NDCG@10 moves the same way ($-2.49 \rightarrow +2.91$ on the head). The repair is not free: relative to A1, A2 gives up 1.85 pp on the tail, which remains $+8.72$ pp above A0. The trade is therefore about two points of tail for nearly six points of head, and we do not claim the tail is untouched. The head cost is specific to the lexical encoder used here; Section 4.4 quantifies that boundary.

Zeroing item content at inference is a diagnostic, not a separate arm: it zeroes the content term so that the representation reduces to the A0 one, while leaving the trained weights in place. It costs A1 13.2–25.3 pp of overall Recall@10 and A2 48.9–55.0 pp across the three domains, which takes the zeroed A2 model to 6.1–7.8% overall Recall@10, below the 9.9% that random ranking yields when one positive competes against 100 negatives. Neither arm returns to the content-free baseline

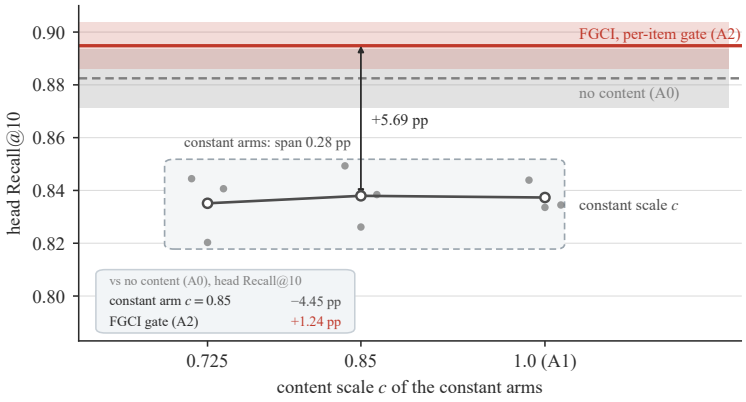

Figure 2: Head Recall@10 for the two constant-scale arms ($c = 0.725$ and $c = 0.85$) against the gate (A2) and the no-content reference (A0). Both constant scales fall below A0 and within 0.3 pp of each other; only the gate sits above A0.

Table 3: Health only, Recall@10 difference against A0, in percentage points, $n = 3$ seeds. Arms are defined in Table 1; all seven share one A0 per seed.

Arm	all	head	mid	tail
A1 unconditional	+3.61	−4.52	+4.89	+10.57
A2 FGCI	**+5.98**	**+1.24**	+8.02	+8.72
A3 permuted gate input	+3.54	−4.61	+4.82	+10.54
A4 frozen backbone, gate tuned only	+3.63	−4.35	+4.88	+10.47
A5 gate off at evaluation	−7.13	−23.77	−18.72	+21.78
U 0.725	+3.14	−4.73	+4.33	+9.93
U 0.85	+3.45	−4.45	+4.72	+10.19

when content is removed: A0 sits at 46.0–52.7%, A1 falls 5.8–22.8 pp below it, and A2 falls below chance.

4.2 A SCALAR CONSTANT IS NOT ENOUGH

Table 3 gives the full Health chain. Two constant-scale controls sit next to the learned gate: U 0.725 and U 0.85 each apply one fixed factor to every item. Neither repairs the head (−4.73 and −4.45 pp); both land on the unconditional level (−4.52 pp), and the two constants and the unconditional arm span only 0.28 pp between them on the head stratum. Figure 2 shows the two constant arms against the gate and against no content on that stratum. A2 is 5.69 pp above U 0.85 on that stratum, against the 5.75 pp that joint training recovers over A1. The two differ by 0.06 pp because U 0.85 is itself only 0.06 pp from A1 on that stratum (−4.45 against −4.52): almost all of the gate's head advantage over a fixed scale is the head repair it delivers over the unconditional arm. Because every arm is trained on the same seeds, we test the contrast by seed: a paired test gives $t = 10.82$ ($\mathrm{df} = 2$, $p = 0.008$) on Health, $t = 9.80$ ($p = 0.0102$) on Grocery and $t = 23.79$ ($p = 0.002$) on Beauty. The paired form has only two degrees of freedom, so we also report the pooled unpaired test, which is the one behind the abstract's $p < 0.01$ claim: it gives $p = 0.0034$, 0.0017 and 0.0002 in the same order (Table 5). Section 6 discusses the limits of three-seed significance.

Two more arms bracket what the gate is doing. A3 permutes the gate's frequency input and reproduces the unconditional arm almost exactly (+3.54 / −4.61 pp), so the gain is not an artefact of having extra parameters in the content path. A4 freezes the backbone at its A1 weights and tunes only the gate: it is also indistinguishable from A1 (+3.63 / −4.35 pp, a head difference of +0.17 against A1's −4.52). A5 trains with the gate but evaluates with it off, and collapses — overall, head and mid fall together (−7.13 / −23.77 / −18.72 pp) while the tail rises to +21.78 pp. Neither half of the gate works in isolation.

The gate's own α is worth reporting directly. Averaged over items with equal weight — the item-level stratification of Section 3 — the mean α is 0.854 / 0.681 / 0.641 for Health, 0.861 / 0.707 /

Table 4: Recall@10 difference against each configuration's own A0, in percentage points, $n = 3$ seeds. *fix* is A2−A1 on the head stratum. U 0.85 is the same constant in every row.

Representation	Dataset	A1 head	U 0.85 head	A2 head	fix	A1 all
crc32 lexical @ 128	Health	-4.52	-4.45	$+1.24$	$+5.75$	$+3.61$
bow lexical @ 768	Health	-3.76	-3.94	$+1.02$	$+4.78$	$+5.17$
semantic @ 768	Health	-0.14	-0.46	$+2.18$	$+2.32$	$+6.93$
semantic @ 768	Grocery	$+0.39$	$+0.28$	$+2.32$	$+1.93$	$+9.08$
semantic @ 768	Beauty	$+2.51$	$+2.03$	$+4.58$	$+2.07$	$+13.59$

0.630 for Grocery and 0.880 / 0.742 / 0.722 for Beauty across head / mid / tail, with the ordering head $>$ mid $>$ tail in all 21 gated-arm runs (seven encoder–domain configurations, three seeds each). The ordering of the stratum means is therefore stable, and it survives both the standard item-level terciles ($9/9$) and a 97-point sweep of the head–tail cut point ($873/873$ run–cut combinations); Figure 3 in the appendix shows the underlying gate curve.

It is not monotone pointwise. Local violations of non-decreasingness appear in 6 of the 9 runs for which we dumped the per-item gates, with maximum decreases of 2.6×10^{-3} to 6.0×10^{-2}, three to five orders of magnitude above the 10^{-6} numerical tolerance. The violations sit in the low-frequency region and do not disturb the head–tail separation.

4.3 Cross-domain behaviour

Table 2 repeats the comparison on Grocery and Beauty. All three domains agree on the qualitative picture: the unconditional arm hurts the head (-4.52 / -3.14 / -1.70 pp), the gate lifts it to ($+1.24$ / $+1.74$ / $+4.06$ pp), and the constant does not (-4.45 / -3.34 / -1.90 pp). Two observations refine the picture of Section 1. First, the repair over the unconditional arm is roughly the same size everywhere ($+5.75$ / $+4.87$ / $+5.76$ pp) even though the harm it repairs shrinks monotonically with the content signal. Second, A2 pays a tail cost relative to A1 in all three domains (-1.85 / -2.89 / -2.03 pp); Grocery is still $+8.40$ pp above A0 on the tail, and A2 remains above A0 in every stratum of every domain (Table 8). We return to the first observation in Section 5.

4.4 Encoder and dimension

The head cost above is measured with the crc32 lexical encoder at 128 dimensions, and it is fair to ask whether injection itself or this particular lexical encoder is responsible. Table 4 holds the model fixed and varies the representation along two axes. Holding the encoder type and raising the dimension to 768 moves the head cost from -4.52 to -3.76 pp; holding the dimension at 768 and switching to a frozen semantic encoder moves it from -3.76 to -0.14 pp. Dimension accounts for 0.75 pp, representation type for 3.62 pp, both measured on Health only.

With the semantic encoder the head cost is gone in every domain: -0.14 pp on Health (standard deviation 0.82), $+0.39$ pp on Grocery (standard deviation 1.22) and $+2.51$ pp on Beauty (standard deviation 0.38), the last the only one clearly above seed noise. The boundary lies in the encoder, not in the dimension. The head effect is positive in 11 of 12 domain $\times$ metric cells, against 4 of 12 under crc32 (Table 9); this is a cell count rather than an independent test, since the four metrics within a domain are correlated. We therefore attribute the head cost to the lexical encoder rather than to content injection. The gate's repair over the unconditional arm is smaller in the semantic setting ($+2.32$ / $+1.93$ / $+2.07$ pp) but keeps its sign and remains significant under the paired test in all three domains ($p = 0.004$ / 0.038 / 0.004, Grocery the marginal one at three seeds), and its head advantage over the constant-scale control survives the encoder swap ($+2.64$ / $+2.03$ / $+2.55$ pp, $p < 0.05$ in all three). The head $>$ mid $>$ tail ordering of α narrows without reversing (Health: $0.214 \to 0.171 \to 0.103$ for crc32@128, bow@768 and semantic@768). This section bounds the head-cost claim rather than re-evaluating the gate: it ran only the arms of Table 4.

5 Discussion

Taken together, the arms of Table 1 narrow the recovery to a specific set of conditions: per-item, frequency-conditioned, jointly learned gating. It is absent under unconditional injection (A1), con-

stant scales at the values we swept ($U\,0.725$, $U\,0.85$), a permuted conditioning input (A3), a post-hoc gate (A4), and gating at training but not at evaluation (A5). The last is the most severe: removing the gate at test time costs more than the gate ever bought and drops the model below the no-content baseline—the same collapse that zeroing the content term at inference produces (Section 4).

The direction of the learned gate, however, is the opposite of what this literature predicts. Content helps the tail in every configuration we ran; but where Zou et al. (2026) report that the weight on the ID channel rises with interaction frequency, the α we learn is largest on the head and smallest on the tail, in all seven encoder–domain configurations, with a head−tail gap of 0.21 / 0.23 / 0.16 across the three crc32 runs. On Health the gap narrows from 0.21 to 0.17 to 0.10 as the encoder changes from crc32@128 to bow@768 to semantic@768. This ordering is the reverse of the one we registered (tail $>$ mid $>$ head), so we report it as an empirical observation rather than as a predicted effect; our controls vary whether conditioning exists and how it is trained, not what it is conditioned on, so we cannot explain it either.

Two natural explanations for the repair also fail. The inversion does not account for it, and neither does the value α takes: the constant arm is flat across the range we swept, and the gate's evaluation-side head level falls inside that range, so matching the gate's average is not enough. A more informative pattern is that the repair does not track the damage. The head damage varies by a factor of 2.7 across the three domains (-4.52 / -3.14 / -1.70 pp), so a mechanism that undid it in proportion would repair Health most and Beauty least; instead the three repairs span less than 0.9 pp ($+5.75$ / $+4.87$ / $+5.76$ pp). Whatever the gate does on the head, it does not scale with the damage it removes; why its size is nearly constant across domains remains open alongside the mechanism itself.

These open questions bound what we can claim in two further ways. First, the conditioning signal is an item-level count, so two items of similar popularity but different content utility receive the same scale; frequency is attractive because it is leak-free and requires no extra annotation, but it is not a proxy for content quality. Second, the tail benefit is not universal: Glisovic et al. (2026) report the opposite for the sparsest items, where removing the visual features improves Hit@10 for 0.31 of low-gate, low-interaction items against 0.04 of high-gate, high-interaction ones. Content helps the tail where it carries information the co-occurrence signal lacks, not by virtue of being content; the `description` control in Appendix C is the same boundary seen from inside our own setup. Why per-item, jointly learned conditioning recovers the head is therefore a question our controls have narrowed but not resolved.

6 Conclusion and Limitations

We asked what content injection does to head items in isolation. On three categories, under the lexical encoder we study, it costs the head while helping the tail, and a per-item, frequency-conditioned gate learned jointly with the backbone removes that cost while keeping most of the tail gain. An unconditional, post-hoc, training-only, or constant variant we test reproduces none of it, so within our controls the conditioning must be per-item, frequency-conditioned, and learned jointly with the backbone; why that suffices remains open.

Two caveats bound that conclusion. The head cost is a property of the hashed lexical representation rather than of content injection: under a frozen language-model encoder it is gone or reversed in all three categories (-0.14 / $+0.39$ / $+2.51$ points, against -4.52 / -3.14 / -1.70), and the gap between representations (3.62 points) is roughly 4.8 times the gap from changing dimension alone (0.75 points). With three seeds our paired tests have two degrees of freedom, so we report the tests themselves—our $p < 0.01$ claim uses the unpaired $df = 4$ test—and read the resulting p-values as indicative rather than as effect sizes; the weakest head comparison under the paired test (Grocery, $t = 9.80$) clears the 5% critical value but not the 1% one.

A third caveat bounds the experiment. The seven-arm chain is measured on Health and Household alone; the constants we sweep (0.725, 0.85) sit near the head level and leave the range below 0.725 untested. All experiments use a single SASRec backbone, so architecture generality is left to future work.

AI USE STATEMENT

We used generative AI tools to assist with drafting and polishing the text, and to help debug the experiment code. The AI tools were not used for research ideation, experimental design, or data analysis. All AI-assisted content was reviewed, verified, and edited by the authors. We take full responsibility for the final content of this work.

ETHICS STATEMENT

This work uses publicly available Amazon Reviews 2023 data and does not involve human subjects or personal data. We do not foresee any ethical concerns.

REPRODUCIBILITY STATEMENT

We report the data split, model configuration, and training details in the main text. Code and evaluation scripts will be released upon publication.

REFERENCES

Himan Abdollahpouri, Masoud Mansoury, Robin Burke, and Bamshad Mobasher. The unfairness of popularity bias in recommendation. *arXiv preprint arXiv:1907.13286*, 2019.

Hao Ding, Yifei Ma, Anoop Deoras, Yuyang Wang, and Hao Wang. Zero-shot recommender systems. *arXiv preprint arXiv:2105.08318*, 2021.

Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. Recommendation as language processing (RLP): A unified pretrain, personalized prompt & predict paradigm (P5). In *Proceedings of the 16th ACM Conference on Recommender Systems*, pp. 299–315, 2022. doi: 10.1145/3523227.3546767.

Natalija Glisovic, Danica Kragic, and Martin Tegner. Deciding when to rely on visual information: Gated multimodal fusion in sequential recommendation. In *CARS Workshop at ACM RecSys*, 2026. arXiv:2608.10700.

Yupeng Hou, Shanlei Mu, Wayne Xin Zhao, Yaliang Li, Bolin Ding, and Ji-Rong Wen. Towards universal sequence representation learning for recommender systems. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 585–593, 2022. doi: 10.1145/3534678.3539381.

Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *2018 IEEE International Conference on Data Mining (ICDM)*, pp. 197–206. IEEE, 2018.

Jiacheng Li, Ming Wang, Jin Li, Jinmiao Fu, Xin Shen, Jingbo Shang, and Julian McAuley. Text is all you need: Learning language representations for sequential recommendation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1258–1267, 2023. doi: 10.1145/3580305.3599519.

Jiayi Liao, Sihang Li, Zhengyi Yang, Jiancan Wu, Yancheng Yuan, Xiang Wang, and Xiangnan He. LLaRA: Large language-recommendation assistant. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1785–1795, 2024. doi: 10.1145/3626772.3657690.

Qidong Liu, Xian Wu, Yejing Wang, Zijian Zhang, Feng Tian, Yefeng Zheng, and Xiangyu Zhao. LLM-ESR: Large language models enhancement for long-tailed sequential recommendation. In *Advances in Neural Information Processing Systems*, 2024.

Harald Steck. Item popularity and recommendation accuracy. In *Proceedings of the Fifth ACM Conference on Recommender Systems*, pp. 125–132, 2011. doi: 10.1145/2043932.2043957.

Maksims Volkovs, Guang Yu, and Tomi Poutanen. DropoutNet: Addressing cold start in recommender systems. In *Advances in Neural Information Processing Systems*, 2017.

Yuhao Wang, Junwei Pan, Xinhang Li, Maolin Wang, Yuan Wang, Yue Liu, Dapeng Liu, Jie Jiang, and Xiangyu Zhao. Empowering large language model for sequential recommendation via multimodal embeddings and semantic ids. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pp. 3209–3219, 2025. doi: 10.1145/3746252.3761169.

Yueqi Xie, Peilin Zhou, and Sunghun Kim. Decoupled side information fusion for sequential recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1611–1621, 2022. doi: 10.1145/3477495.3531963.

Tingting Zhang, Pengpeng Zhao, Yanchi Liu, Victor S. Sheng, Jiajie Xu, Deqing Wang, Guanfeng Liu, and Xiaofang Zhou. Feature-level deeper self-attention network for sequential recommendation. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 4320–4326, 2019. doi: 10.24963/ijcai.2019/600.

Zhi Zhang, Xiudan Yang, Wenjie Luo, and Huan Liang. LLM2Emb: A two-stage LLM-based embedding framework for long-tail sequential recommendation. *Expert Systems with Applications*, 331:133375, 2026. doi: 10.1016/j.eswa.2026.133375.

Kun Zhou, Hui Wang, Wayne Xin Zhao, Yutao Zhu, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. S3-Rec: Self-supervised learning for sequential recommendation with mutual information maximization. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*, pp. 1893–1902, 2020. doi: 10.1145/3340531.3411954.

Xinchi Zou, Tongzhenzhi Su, Jianjun Li, Yuan Fu, Chang Liu, Zhiying Deng, and Zhiwei Shen. DCGL: Dual-channel graph learning with large language models for knowledge-aware recommendation. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2633–2643, 2026. doi: 10.1145/3805712.3809546.

A Statistics for the gate–constant contrast

Table 5: Statistics for the gate-versus-constant contrast on the head stratum. Each row compares FGCI (A2) against U 0.85 on the same seed; Δ is the head Recall/NDCG difference in percentage points (positive means the gate is better). Paired test: $\mathrm{df} = 2$; unpaired: pooled three-seed samples, $\mathrm{df} = 4$, and this is the test behind the $p < 0.01$ claim in the abstract.

Domain	Metric	Δ (pp)	t (paired)	p (paired)	t (unpaired)
Health	R@10	+5.69	10.82	0.0084	6.24
Health	R@20	+4.02	8.99	0.0122	6.68
Health	N@10	+5.36	9.46	0.0110	5.68
Health	N@20	+4.94	9.05	0.0120	5.73
Grocery	R@10	+5.08	9.80	0.0102	7.48
Grocery	R@20	+3.52	11.05	0.0081	9.12
Grocery	N@10	+5.07	8.83	0.0126	7.22
Grocery	N@20	+4.68	8.91	0.0124	7.44
Beauty	R@10	+5.96	23.79	0.0018	13.24
Beauty	R@20	+3.98	23.53	0.0018	12.08
Beauty	N@10	+6.05	16.53	0.0036	10.56
Beauty	N@20	+5.55	16.10	0.0038	10.23

B Per-run gate diagnostics

Table 6: Per-run gate diagnostics for the nine runs with dumped per-item gates (three categories $\times$ three seeds, arm A2). $\bar{\alpha}$: item-level stratum means. "non-dec.": is α non-decreasing along $\log(1 + \text{count})$; "max dec.": largest local decrease; "sweep": head–tail cut points (of 97 equally spaced quantiles of the item-level log-count) at which the head mean exceeds the tail mean; "min gap": smallest head-minus-tail margin over that sweep.

Run	Seed	$\bar{\alpha}_{\mathrm{h}}$	$\bar{\alpha}_{\mathrm{m}}$	$\bar{\alpha}_{\mathrm{t}}$	non-dec.	#viol.	max dec.	sweep	min gap
Health	2026	0.8511	0.6764	0.6395	no	8	0.0596	97/97	0.066
Health	2027	0.8486	0.6752	0.6348	yes	0	0	97/97	0.106
Health	2028	0.8633	0.6920	0.6476	no	4	0.0026	97/97	0.095
Grocery	2026	0.8554	0.7055	0.6288	yes	0	0	97/97	0.149
Grocery	2027	0.8680	0.7133	0.6348	no	5	0.0445	97/97	0.142
Grocery	2028	0.8584	0.7027	0.6272	yes	0	0	97/97	0.147
Beauty	2026	0.8797	0.7373	0.7211	no	10	0.0095	97/97	0.036
Beauty	2027	0.8795	0.7468	0.7236	no	8	0.0098	97/97	0.043
Beauty	2028	0.8797	0.7424	0.7217	no	9	0.0145	97/97	0.042

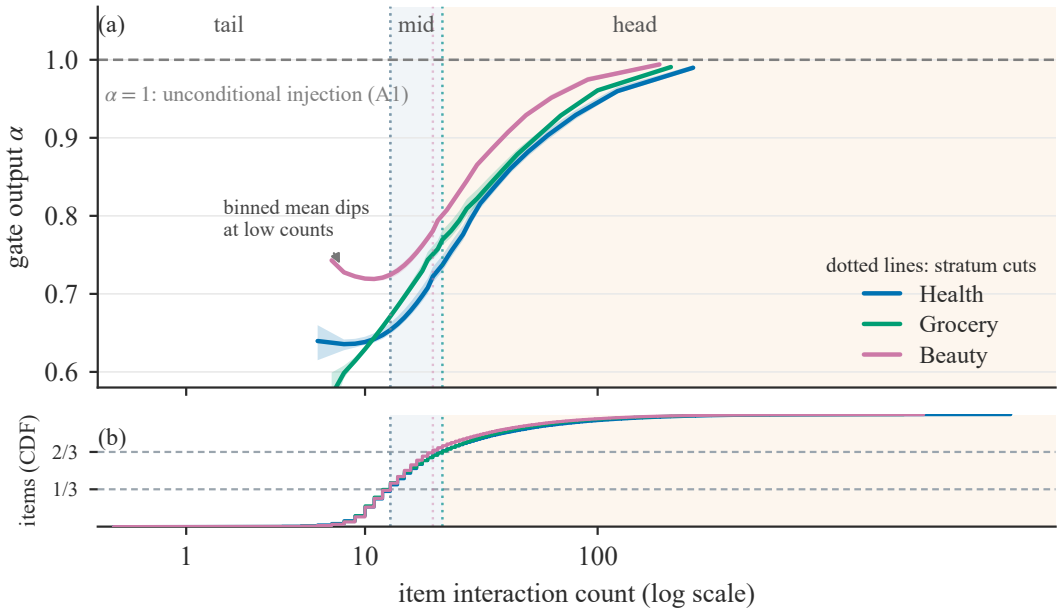

Figure 3: Learned gate output against item frequency. Solid lines are the mean α_i over items in each frequency bin, for the three domains; the shaded band is the range across the three seeds. Dotted vertical lines mark the item-level tercile cuts that define the strata; the lower panel shows the empirical CDF of item counts with the same cuts. The dashed horizontal line is $\alpha = 1$, the unconditional-injection operating point (A1).

C CONTENT-SIGNAL PROBE ON TAIL ITEMS

If the benefit of content is that long-tail items are described rather than remembered, the size of the gain should follow how much a content signal carries beyond co-occurrence. We probe this by reusing the evaluation protocol: the true target and its 100 sampled negatives are scored by a content-only scorer (token overlap between the candidate's title and the user's recent history) and by a co-occurrence baseline (the training-set transition probability from the last item), and we report the AUC gap (content − co-occurrence) on tail items. That gap is 0.108 / 0.123 / 0.141 for Health / Grocery / Beauty, and the unconditional arm's overall Recall@10 gain is +3.61 / +6.66 / +10.32 pp in the same order, with tail gains of +10.57 / +11.29 / +17.78 pp. The two orderings agree in all three domains, though three points cannot establish a trend.

The `description` field serves as a negative control with a known defect: it is absent for just over half of Health. Its tail gap under the same configuration is negative (−0.037 / −0.012 / −0.014). In a single-seed field ablation on Grocery and Beauty, the field with the weakest probe signal also delivers the smallest tail gain: +6.61 and +8.52 pp for `description` against +11.57 and +18.13 pp for `title+category`. The probe's ordering tracks the gain, but a negative probe gap does not null it.

Table 7: Content-signal probe on tail items, per field. "content" is the AUC of scoring the true target against its 100 sampled negatives by token overlap between the candidate's field and the user's history (max over history items); "co-occ." replaces that score by the training-set transition probability from the last item; the gap is their difference. All 100 negatives are scored and a missing-field target receives a content score of zero, so these are the numbers the deployed content matrix delivers; `cov` is the fraction of tail examples whose *history* contains the field at all, which is why it stays near 1 even for a field most items lack. `desc` is the negative control: its item-level missing rate in Health is 50.09% (this is not `cov`, which measures the history side). The text above quotes these rows.

Domain	Field	AUC (content)	AUC (co-occ.)	gap	n	cov.
Health	`title`	0.611	0.503	+0.108	8322	1.000
Health	`category`	0.620	0.503	+0.117	8322	1.000
Health	`desc`	0.466	0.503	−0.037	8322	0.982
Grocery	`title`	0.627	0.504	+0.123	8112	1.000
Grocery	`category`	0.632	0.504	+0.128	8112	1.000
Grocery	`desc`	0.492	0.504	−0.012	8112	0.997
Beauty	`title`	0.645	0.504	+0.141	8325	1.000
Beauty	`category`	0.646	0.504	+0.142	8325	1.000
Beauty	`desc`	0.491	0.504	−0.014	8325	0.974

D Full per-stratum results

Table 8: Recall@10 difference against the A0 baseline of the same seed, in percentage points, for all four strata of all three categories under the crc32@128 representation ($n = 3$ seeds); positive means the arm is better than no content. This is Table 2 extended with the mid stratum and with the full row of the constant-scale control. A2 lies above the A0 baseline in all twelve domain×stratum cells; the constant scale lies below it on the head in every category.

Domain	Arm	all	head	mid	tail
Health	A1	+3.61	−4.52	+4.89	+10.57
	A2	+5.98	+1.24	+8.02	+8.72
	U 0.85	+3.45	−4.45	+4.72	+10.19
Grocery	A1	+6.66	−3.14	+11.83	+11.29
	A2	+8.45	+1.74	+15.04	+8.40
	U 0.85	+6.27	−3.34	+11.26	+10.90
Beauty	A1	+10.32	−1.70	+15.06	+17.78
	A2	+12.49	+4.06	+17.68	+15.75
	U 0.85	+9.88	−1.90	+14.49	+17.22

E Head effect by representation and metric

Table 9: Head-stratum difference against each configuration's own A0 baseline, in percentage points, for every domain and metric ($n = 3$ seeds); these are the cells counted in Section 4.4. The head effect is positive in 4 of the 12 cells under the crc32 lexical encoder and in 11 of the 12 under the frozen semantic encoder. The bow@768 configuration was run on Health and Household only.

Domain	Metric	crc32@128	bow@768	semantic@768
Health	R@10	−4.52	−3.76	−0.14
	R@20	−2.64	−1.75	+0.53
	N@10	−2.49	−1.74	+1.80
	N@20	−2.02	−1.23	+1.96
Grocery	R@10	−3.14	—	+0.39
	R@20	−0.69	—	+1.99
	N@10	−0.43	—	+3.34
	N@20	+0.19	—	+3.74
Beauty	R@10	−1.70	—	+2.51
	R@20	+0.02	—	+2.74
	N@10	+1.11	—	+5.59
	N@20	+1.54	—	+5.65