Problem Formulation over Feature Engineering: Evidence from a Production Reranking System in a C2C Marketplace

Ryo Watanabe[1,*], Yusuke Shido[1], Ryo Tanaka[1], Yuki Yada[1] and Shinya Yaginuma[1]

[1]*Mercari, Inc., Tokyo, Japan*

Abstract

We present a three-round production case study of home-feed item recommendation at Mercari, a Japanese consumer-to-consumer (C2C) marketplace with 23 million monthly active users. In Rounds 1 and 2, a pointwise multilayer perceptron (MLP) reranker was deployed to predict tap events; Round 2 further incorporated image and text embeddings. Although both approaches increased the online session-level tap rate by 5–9%, they reduced the component-attributed first-touch purchase rate by 12–19%, and neither model was deployed. In Round 3, no additional input features were introduced. Instead, the training objective was reformulated using a listwise LambdaRank loss with a graded relevance label spanning the full view-to-purchase funnel. This approach increased the tap rate by 17.6% and the first-touch purchase rate by 19–21% relative to a no-reranker baseline, leading to full production deployment. A controlled offline comparison under matched retrieval conditions isolates the effect of the objective reformulation from a concurrent retrieval upgrade. These results highlight two key findings: (1) in C2C marketplaces, optimizing for tap rate can negatively impact purchase conversion, and (2) problem formulation—specifically the choice of labels and loss functions—can be more influential than feature engineering.

Keywords

learning to rank, reranking, e-commerce, listwise ranking, LambdaRank, production case study, offline–online metric divergence

1. Introduction

Personalized item recommendation on the home feed of a consumer-to-consumer (C2C) marketplace is commonly formulated as a retrieval-and-reranking problem [1], in which a retrieval model [2] generates a candidate set and a lightweight reranker orders items for display. Once such a pipeline is established, a common next step is to improve the reranker by incorporating additional features and increasing model capacity, an approach widely observed in industrial systems, such as YouTube, Airbnb, and Alibaba [1, 3, 4].

Mercari is the largest C2C marketplace in Japan, with 23 million monthly active users and over 4 billion listed items. The home screen serves as a primary discovery surface that (i) deepens users' existing interests by surfacing novel yet relevant items that may trigger impulse purchases, and (ii) broadens user interests by enabling serendipitous exploration of the platform's long-tailed inventory beyond immediate intent. In such settings, engagement signals may reflect heterogeneous user intents, ranging from curiosity-driven browsing to purchase-ready behavior, making the choice of training objective particularly consequential.

This paper reports three online A/B tests of home-feed reranking on Mercari that examine the limits of feature-centric optimization in this setting. In Rounds 1 and 2, a pointwise multilayer perceptron (MLP) reranker trained on a binary tap label—despite an expanded feature set with content embeddings in Round 2—consistently improved the session-level tap rate (5–9%) but degraded component-attributed first-touch purchase outcomes (12–19%); consequently, neither model was deployed. In Round 3, we retained the reranker architecture with no additional input features, but replaced the training objective with a listwise LambdaRank loss defined over a graded-relevance label that orders engagement signals

SIGIR 2026 Workshop on eCommerce (eCom'26), July 24, 2026, Melbourne, Australia

*Corresponding author.

r-b-watanabe@mercari.com (R. Watanabe); shidoy@mercari.com (Y. Shido); ryo.tanaka@mercari.com (R. Tanaka); y-b-yada@mercari.com (Y. Yada); s-yaginuma@mercari.com (S. Yaginuma)

along the view-to-purchase funnel. This modification resulted in improvements in both tap and purchase metrics in the A/B tests, and the model was subsequently deployed to production.

A funnel-based decomposition localizes the observed regression to the like-to-purchase-start transition, suggesting that optimizing for taps increased exposure to items that were more attention-grabbing but less likely to convert into purchase actions. In other words, the pointwise tap objective learned to rank items by tap propensity, which, in a C2C marketplace—characterized by single-unit, seller-posted, and heterogeneously conditioned listings—may be negatively correlated with purchase intent.

These findings point to a broader implication: under conditions of proxy–KPI divergence, problem formulation (specifically, the choice of training labels and loss functions) can have a greater impact on downstream purchase outcomes than feature engineering. Because Round 3 coincided with an upgrade to the retrieval layer, a controlled offline comparison (Section 6) trains pointwise and listwise variants on the same data with identical features, thereby isolating the contribution of the objective reformulation from that of the retrieval change.

Contributions. (i) A three-round production case study of home-feed reranking in a C2C marketplace, covering both unsuccessful and successful iterations. (ii) Empirical evidence, based on server-log A/B testing data, that optimizing for click-based proxies can adversely affect purchase conversion, with a funnel decomposition localizing the regression to the like-to-purchase-start transition (Section 4). (iii) A graded-relevance LambdaRank formulation that aligns offline NDCG@10 with a purchase-correlated engagement funnel without introducing additional input features (Section 5). (iv) A controlled offline comparison that holds data, features, and model capacity constant while varying the training objective (i.e., label and loss jointly), thereby isolating the effect of problem formulation from concurrent changes in retrieval and training scope (Section 6).

2. Related Work

The learning-to-rank literature categorizes ranking losses into pointwise, pairwise, and listwise approaches, with increasingly direct optimization of rank-sensitive metrics across this hierarchy [5]. RankNet [6] introduced pairwise neural ranking, whereas LambdaRank [7, 8] extended this framework by scaling gradients according to the NDCG swap delta, thereby directly optimizing rank-sensitive objectives. Listwise methods include ListNet [9] and ApproxNDCG [10].

Industrial applications of learning-to-rank are now widespread in consumer-facing platforms, including YouTube's deep ranking system [1], Airbnb's embedding-based personalization [3], and Alibaba's Transformer-based reranker [4]. However, these systems are typically optimized with respect to a single dominant engagement proxy. When engagement and conversion objectives are misaligned, industrial solutions have often relied on multi-task learning frameworks, such as YouTube's MMoE [11] and Alibaba's ESMM [12], which jointly model click and post-click outcomes.

In contrast, this paper explores an alternative approach: a single graded-relevance LambdaRank objective, selected in part for its production interpretability and simplicity, as discussed in Section 7. A related line of research treats implicit feedback signals as biased or misspecified, addressing issues such as position bias through counterfactual learning-to-rank methods [13, 14]. Additionally, Netflix has emphasized that engagement-based proxies only partially capture long-term user satisfaction [15], highlighting a tension that we revisit in Section 7.

Most prior case studies focus on business-to-consumer (B2C) search or video recommendation systems, whereas ranking in C2C marketplaces remains comparatively underexplored in the academic literature [16]. Public reports of production C2C reranking deployments remain limited; this paper documents a case in which both the offline tap-based proxy and the online tap rate improve while the attributed purchase metric declines—a central finding presented in Section 4.

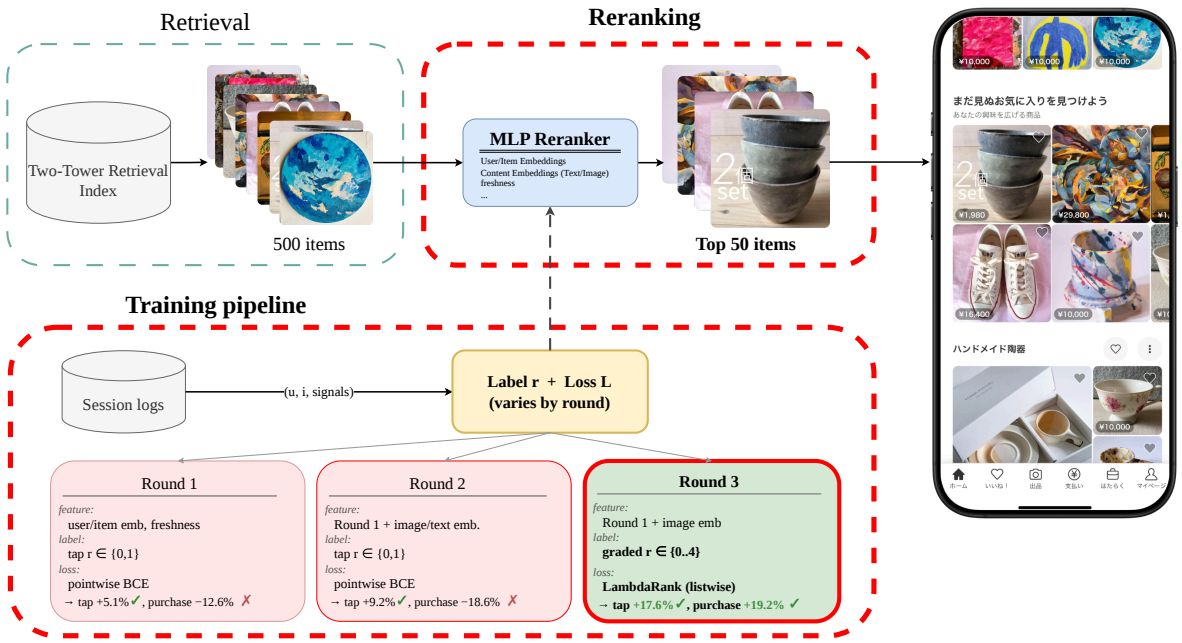

Figure 1: System overview. The serving pipeline retrieves the top 500 candidates per user, reranks them with an MLP, and returns up to 50 items for display. The training pipeline constructs labels from session logs; the primary differences across rounds lie in the label definitions and loss functions (highlighted at the bottom).

3. System and Problem Setup

3.1. Home-Feed itemrec Component

Mercari serves 23 million monthly active users across its mobile applications and web platforms. The home feed consists of a vertical stack of personalized recommendation components. The *item recommendation* component (hereafter referred to as itemrec) is a general-purpose recommender within this feed, drawing from the full marketplace inventory and regenerated once per session.

The primary objective of the component is to promote purchase-correlated browsing on the home feed. Although purchase is the most consequential downstream signal used for evaluation, the component also supports earlier stages of the user journey, including curiosity-driven discovery and pre-purchase deliberation, both of which the component explicitly encourages. Performance is tracked using three online metrics: the *session-level tap rate* (the proportion of home-feed sessions in which the component receives at least one tap, among those in which it is shown), average transactions per user (ATPU), and the *buyer conversion rate* (BCR). ATPU and BCR are reported using first-touch attribution to isolate the component's contribution from other home-feed surfaces. Because a single purchase may be attributed to different components depending on which one directed the user to the item detail page, we explicitly specify the attribution rule throughout.

3.2. Retrieval-and-Reranking Pipeline

Serving is implemented as a two-stage retrieval-and-reranking pipeline (Figure 1). A two-tower retrieval model maps users and items into a shared 64-dimensional embedding space and retrieves the top 500 candidates per user based on inner-product similarity using approximate nearest-neighbor search. These candidates are then passed to a reranker, implemented as an MLP, which scores each user–candidate pair. Item, image, and text embeddings are precomputed by an asynchronous inference pipeline triggered upon item creation or modification, persisted to an online datastore, and retrieved by the reranker at serving time. The top-ranked items are selected and capped at 50 for display on the home feed. The reranker adds approximately 10 ms to serving latency at the 95th percentile, while the full home-feed

itemrec component operates at approximately 250 ms at the same percentile (measured during the Round 3 period). The retrieval model is not held constant across the three rounds. Rounds 1 and 2 use v1 of the two-tower model, whereas Round 3 uses v2. Both versions share the architecture described above and differ only in their training data: v1 builds each user's interaction sequence from purchase events and predicts the immediate next action, whereas v2 broadens this sequence to include views, likes, and purchases, and samples the training target from a multi-day interaction window to capture longer-term interest. Section 6 isolates the effect of this change. In all reported A/B tests, the control condition corresponds to the no-reranker setting (i.e., retrieval outputs are displayed directly), such that each round's measured lift reflects the full incremental contribution of the reranker over retrieval alone.

3.3. Training Data and Labels

A weekly data pipeline processes server logs of home-feed item impressions and filters sessions to those containing at least one tap on the component. In Rounds 1 and 2, the training data consisted of the top four recommended items from each tap-positive session over a one-week period. In Round 3, the candidate set was expanded to include all impressed items per session, still restricted to tap-positive sessions. Each session was then subsampled to four candidates by retaining all high-engagement instances (tap, like, purchase) and filling the remaining positions with randomly selected negative samples. This process was conducted over a three-week window, with an additional seven-day buffer to allow purchase labels to mature.

Each $(\text{session}, \text{candidate})$ pair is labeled with multiple downstream signals, including a binary tap indicator, a binary engaged-view indicator (defined as a tap followed by a dwell time of at least three seconds), a like indicator, a "see more" action on the item detail page (expanding the folded description), a tap on any recommendation widget within the item detail page, and a purchase indicator within seven days. Rounds 1 and 2 trained pointwise models using the binary tap label, whereas Round 3 introduces a graded label $r(u, i) \in \{0, \ldots, 4\}$, defined in Section 5.1. The offline comparison in Section 6 uses a four-samples-per-session subsample from the Round 3 pipeline, comprising 44.3M rows with a 30.8% tap rate, 25.1% engaged-view rate, and 1.28% 7-day purchase rate.

3.4. Evaluation Metrics

Offline evaluation uses NDCG@10 computed over the graded-relevance labels, along with per-label NDCG@10 to identify potential misalignment between pointwise and listwise objectives. Online evaluation reports the session-level tap rate and first-touch attributed purchase metrics (ATPU and BCR), using consistent definitions across all three rounds to ensure comparability.

4. The Paradox of Tap-Optimized Reranking

Rounds 1 and 2 share the same binary tap label and pointwise cross-entropy loss but differ in their feature sets. In both rounds, the component's objective (Section 3.1) was approximated by maximizing the session-level tap rate, based on the working assumption that improving the earliest stage of the funnel would translate into gains in downstream engagement and purchase. Although both rounds increased the tap metric in online A/B tests, they resulted in declines in downstream purchase metrics; consequently, neither model was deployed. This paired outcome suggests that the regression is attributable to problem formulation rather than to a specific feature set: the tap proxy fails to capture the full view-to-purchase funnel, and optimizing it shifts exposure toward items that are easy to tap but difficult to convert.

4.1. Round 1: Pointwise Binary Tap Prediction

The first reranker is an MLP that takes as input the 64-dimensional user and item embeddings produced by the v1 two-tower retrieval model, along with a scalar freshness feature. The training label is a binary

tap indicator derived from the top four items in tap-positive sessions, and the model is trained using binary cross-entropy loss.

The two-week online A/B test included approximately 2.1M users per arm. The session-level tap rate increased by 5.1%; however, the first-touch ATPU attributed to the component decreased by 12.6%, and the first-touch BCR declined by 11.5% (Table 1). Based on these results, the team decided not to deploy the Round 1 model.

4.2. Round 2: Feature Expansion Does Not Suffice

Following Round 1, the working hypothesis attributed the observed degradation to limited model capacity: the restricted feature set might not sufficiently distinguish between items that are attractive for taps and those that are likely to convert into purchases. To address this, Round 2 introduced a 32-dimensional image embedding (obtained via PCA projection of a pretrained SigLIP tower [17]) and a 32-dimensional text embedding derived from a Mercari-fine-tuned ruri-small-v2 Japanese sentence encoder [18, 19]. Offline evaluation showed that the best feature combination improved recall at 1 from 0.2335 to 0.2517 ($+7.8\%$). However, the two-week online A/B test (approximately 2.9M users per arm) reproduced the pattern observed in Round 1 with greater magnitude: the session-level tap rate increased to $+9.2\%$, while the first-touch ATPU attributed to itemrec declined by -18.6%, and BCR decreased by -15.5% (Table 1). These results indicate that richer features improved performance on the optimized objective but further exacerbated the decline in purchase-related outcomes.

4.3. Diagnosis: What the Two Rounds Have in Common

To understand why feature expansion failed to address the issue, we analyzed the types of items surfaced by the reranker. Both rounds preferentially surfaced items priced at ¥10,000 or more ($+35\%$ relative under Round 1, $+18\%$ under Round 2), in "bad" condition ($+48\%$ / $+37\%$), and historically less likely to sell within seven days of listing (-10% / -10%). These results confirm that feature enrichment did not mitigate the exposure biases.

Overall, the reranker preferentially surfaced items that attracted more taps but were more expensive, of lower condition quality, and less likely to be purchased. Consistent with the funnel decomposition described in Section 1, upstream engagement signals (e.g., dwell time and like rate) remained stable or showed slight improvement, indicating that the regression is concentrated at the like-to-purchase-start transition rather than distributed across the funnel. Specifically, among users who liked an item from the home-feed slot, 3.69% in the control proceeded to initiate a purchase flow, compared with 2.99% under the Round 1 treatment, representing a 19% relative decrease concentrated at this transition.

Round 2 did not resolve this issue because the tap label itself is fundamentally misaligned with the desired outcome, rather than the feature set being insufficient. A model that improves recall at 1 for taps simply promotes more tap-attractive items to the top of the ranking. As argued in Section 5, the appropriate remedy is to modify the training objective rather than the feature set.

5. From Tap Prediction to Graded-Relevance Ranking

The diagnosis in Section 4.3 identifies the *training objective*—the combination of label and loss—as the key design choice requiring reconsideration. Given the component's objective described in Section 3.1, a natural question is why Rounds 1 and 2 did not incorporate purchase signals into the training label, despite purchase being the most consequential outcome in the funnel. The primary constraint is sparsity: the 7-day purchase label occurs approximately 24 times less frequently than the tap label (1.28% vs. 30.8%; Section 3.3), so a pointwise model trained on purchase alone is dominated by negative examples. Rounds 1 and 2 therefore relied on the denser binary tap signal, a narrow proxy that, as documented above, misaligned with the funnel.

In contrast, Round 3 employs a graded label that spans the full view-to-purchase engagement funnel under a listwise objective. This formulation provides a denser supervision signal while concentrating

Table 1
Online A/B results for the three reranker rounds, reported as first-touch attributed lift versus the no-reranker control (retrieval-only baseline). ATPU = avg. transactions per user; BCR = buyer conversion rate. Each arm includes 2.1–2.9 M users; all reported lifts are statistically significant at $p < 0.001$ (Welch's t-test on per-user averages).

Round / variant	Tap rate	ATPU	BCR
Round 1, pointwise BCE on tap	$+5.1\%$	-12.6%	-11.5%
Round 2, pointwise BCE + features	$+9.2\%$	-18.6%	-15.5%
Round 3, LambdaRank on graded relevance	$\mathbf{+17.6\%}$	$\mathbf{+19.2\%}$	$\mathbf{+21.0\%}$

ranking pressure on purchase-correlated signals, rather than directly optimizing sparse purchase events. Although the label and loss are conceptually separable, they are interdependent in practice: a graded label is not effectively utilized under pointwise cross-entropy, and a listwise loss provides limited signal when applied to a binary label. In Round 3, the MLP architecture from Round 2 is retained, the image embedding feature is reused, and no additional input features are introduced. The following subsections describe these changes and present the corresponding online results.

5.1. Graded-Relevance Label Design

We define the graded relevance $r(u, i) \in \{0, 1, 2, 3, 4\}$ of an item i for a user u within a session as the maximum over a set of engagement indicators: 4 if the user purchases i within 7 days; otherwise 3 if the user likes i or interacts with recommended content on its item detail page; otherwise 2 if the user expands the item description; otherwise 1 if the user has an engaged view of i (defined as a tap followed by dwell time ≥ 3 s); and 0 otherwise.

The ordering of grades reflects the progression of the engagement funnel from initial interaction to purchase. The training objective is therefore neither pure tap-rate maximization (the failure mode observed in Rounds 1 and 2) nor isolated purchase prediction, but rather a graded engagement signal that emphasizes purchase while retaining broader engagement coverage. Purchase is assigned the highest grade (4) as the most critical outcome, whereas lower grades (e.g., like, description expansion, engaged view) capture earlier and denser stages of the funnel.

5.2. LambdaRank Listwise Loss

We train the model using LambdaRank [8], a pairwise loss in which gradients are weighted by the NDCG swap delta $|\Delta\text{NDCG}_{ij}|$:

$$\mathcal{L}_s = \sum_{(i,j):r_s(i)>r_s(j)} \big|\Delta\text{NDCG}_{ij}\big| \cdot \log\big(1 + \exp\big(-\sigma(s_i - s_j)\big)\big), \tag{1}$$

where s_i denotes the reranker's scalar score for item i, and pairs are constructed within each home-feed session. At inference time, the model outputs a scalar score for each candidate, and items are ranked accordingly.

5.3. Online Results

As in Rounds 1 and 2, the Round 3 A/B test reports the session-level tap rate and first-touch attributed purchase metrics (ATPU and BCR). The deployed model corresponds to the LambdaRank-based treatment, which remains in production. Table 1 summarizes the primary metrics across all three rounds.

Increasing feature richness improved the tap-rate lift ($+5.1\% \rightarrow +9.2\%$) but exacerbated the decline in purchase metrics ($-12.6\% \rightarrow -18.6\%$). In contrast, the label-and-loss reformulation in Round 3 reverses the negative trend, yielding a $+19.2\%$ improvement in purchase metrics and the largest observed tap-rate increase ($+17.6\%$), without introducing new input features beyond the retrieval-v2 upgrade discussed in Section 6.

Table 2
Offline comparison: cross-target NDCG@10. Each row corresponds to a trained model, and each column corresponds to a target label. All models are evaluated on the same held-out test set; the purchase-NDCG column is restricted to the 4.98% of sessions with at least one purchase ($N = 55{,}041$). Bold values indicate the best performance per column; $\pm$ values denote half-widths of 95% bootstrap confidence intervals over sessions ($B = 1000$) [20].

ID	tap NDCG	purchase NDCG†	graded NDCG
E1 (tap BCE)	.749±.002	.719±.002	.731±.002
E2 (purchase BCE)	.700±.002	**.775**±.002	.685±.002
E3 (LambdaRank)	**.755**±.002	.743±.002	**.738**±.002

†Purchase-positive sessions only ($N = 55{,}041$).

5.4. A Confounder: Retrieval-Side Embedding Update

Round 3 was not implemented as a purely isolated change in label and loss. During the same period, the upstream two-tower retrieval model was upgraded from v1 to v2 (the difference between the two versions is described in Section 3.2). Consequently, both the treatment and control arms in the Round 3 A/B test used retrieval-v2 embeddings. While the within-round comparison remains valid—since the retrieval upgrade is shared across both arms and does not affect the treatment–control difference—the comparison across rounds involves differing baseline conditions. Specifically, Rounds 1 and 2 were evaluated against a v1-retrieval control, whereas Round 3 was evaluated against a v2-retrieval control. Therefore, the offline comparisons in Section 6 should be interpreted as evidence that the objective reformulation improves the relevant metrics under a fixed retrieval setup, rather than as a precise estimate of its isolated online impact.

6. Controlled Offline Comparison

This section isolates the effect of the loss-and-label reformulation from the retrieval-side upgrade through three controlled offline training runs that hold the dataset (the Round 3 training set, 44.3M rows), feature set, MLP architecture, and optimizer constant; only the training label and loss differ. We compare three variants: **E1** (pointwise BCE on the tap label, 30.8% positive rate), reproducing the Rounds 1 and 2 objective under the updated feature set; **E2** (pointwise BCE on the 7-day purchase label, 1.3% positive rate); and **E3**, the Round 3 LambdaRank model on the graded relevance label $r \in \{0, \ldots, 4\}$ described in Section 5.2. Performance is evaluated using cross-target NDCG@10: each model is assessed on the same held-out dataset against multiple target labels (Table 2).

The resulting cross-target performance matrix supports the central claim of this paper. Model E3 (LambdaRank with graded relevance) achieves the highest NDCG@10 for both tap and graded-relevance targets. Model E2 (pointwise training on purchase alone) attains the highest purchase-NDCG ($\mathbf{0.775}$ compared to E3's 0.743), reflecting specialization on the sparse purchase signal. However, E2 performs worst on tap-NDCG (0.700 vs. 0.755) and graded-NDCG (0.685 vs. 0.738), representing a 0.053-point deficit in engagement-oriented ranking for a 0.032-point gain in purchase-NDCG.

Given that the production objective is to promote purchase-correlated browsing—rather than to maximize purchase in isolation, which would be both misaligned with the component's goals and, as discussed in Section 5, impractical due to label sparsity—we selected E3 for deployment. This model achieves the strongest performance on engagement-oriented metrics while maintaining competitive purchase-NDCG, a profile consistent with the observed behavior in the Round 3 online A/B test.

7. Discussion

Problem formulation dominates feature engineering. Across the three rounds, the reranker architecture and training infrastructure are held constant, while the training label and loss function

vary (with the retrieval change between Rounds 2–3 isolated in Section 6). Feature engineering in Round 2 improved the offline tap-prediction metric but did not mitigate the observed regression in online purchase metrics. In contrast, modifying what the model is *trained to predict*—without introducing additional tap-related input features—reversed the direction of the online purchase metric.

A standard industrial alternative is multi-task learning that jointly models click and downstream user signals (e.g., MMoE [11] and ESMM [12]). In this work, we deliberately adopt a graded-relevance LambdaRank formulation: it maintains a single-scalar scoring interface compatible with the existing retrieval-and-reranking pipeline, encodes funnel priorities within a single ordinal label that can be inspected by product teams, and avoids the need to tune auxiliary loss weights under production constraints.

Graded relevance as a legible priority encoding. Neither the graded-relevance label nor LambdaRank [7, 8] is novel. The contribution here is the explicit use of graded relevance as a transparent encoding of funnel priorities (engaged view, description expansion, like, purchase) that can be jointly reviewed and refined by machine learning and product teams. For example, stakeholders can evaluate which engagement signals should be assigned to each grade or how grades should be spaced.

Limitations. Several limitations remain. The cross-round online comparison conflates the label-and-loss reformulation with concurrent training-pipeline changes (Section 3.3); the controlled offline comparison in Section 6 addresses this by holding these factors constant. The training data is restricted to tap-positive sessions, applied consistently across rounds; experiments span ≤ 14 days and therefore do not capture long-term outcomes such as repeat purchases or retention. Results are obtained in a C2C marketplace with long-tailed, ephemeral inventory and may not generalize to B2C catalogs with more stable inventory. Finally, the offline comparison varies the label and loss jointly; isolating their individual contributions would require an intermediate formulation, such as a weighted pointwise objective.

8. Future Work and Conclusion

Future extensions include exploring alternative gain functions, varying the number of relevance levels in the graded label, incorporating explicit negative signals such as quick-back taps, and benchmarking the multi-task alternatives discussed in Section 7 against the graded-relevance formulation.

In the specific context of home-feed item recommendation in a C2C marketplace, the three production reranker iterations demonstrate that problem formulation—the choice of training label and loss function—is not a secondary design decision to be addressed after feature engineering. Rather, it is the primary lever influencing business outcomes. A simple graded-relevance listwise objective is sufficient to transform a negative outcome into a successful production deployment.

Acknowledgments

We thank the members of the Recommendation team at Mercari for their substantial support throughout this work. We are also grateful to the Eliza team and the Search team, in particular Sho Akiyama, Andre Rusli, Aman Singh, Changchi Xian, and Takuma Kinoshita, for developing and providing the inference and serving pipelines for the image and text embeddings used in this study. This work would not have been possible without their contributions.

Declaration on Generative AI

During the preparation of this work, the authors used Claude (Anthropic) for drafting assistance, paraphrasing, and LaTeX scaffolding. After using this tool, the authors reviewed and edited the content as necessary, assuming full responsibility for the content of the publication.

References

[1] P. Covington, J. Adams, E. Sargin, Deep neural networks for YouTube recommendations, in: Proceedings of the 10th ACM Conference on Recommender Systems (RecSys), 2016, pp. 191–198.

[2] X. Yi, J. Yang, L. Hong, D. Z. Cheng, L. Heldt, A. Kumthekar, Z. Zhao, L. Wei, E. H. Chi, Sampling-bias-corrected neural modeling for large corpus item recommendations, in: Proceedings of the 13th ACM Conference on Recommender Systems (RecSys), 2019, pp. 269–277.

[3] M. Grbovic, H. Cheng, Real-time personalization using embeddings for search ranking at Airbnb, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2018, pp. 311–320.

[4] C. Pei, Y. Zhang, Y. Zhang, F. Sun, X. Lin, H. Sun, J. Wu, P. Jiang, J. Ge, W. Ou, D. Pei, Personalized re-ranking for recommendation, in: Proceedings of the 13th ACM Conference on Recommender Systems (RecSys), 2019, pp. 3–11.

[5] T.-Y. Liu, Learning to rank for information retrieval, Foundations and Trends in Information Retrieval 3 (2009) 225–331.

[6] C. J. C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, G. N. Hullender, Learning to rank using gradient descent, in: Proceedings of the 22nd International Conference on Machine Learning (ICML), 2005, pp. 89–96.

[7] C. J. C. Burges, R. Ragno, Q. V. Le, Learning to rank with nonsmooth cost functions, in: Advances in Neural Information Processing Systems 19 (NIPS), 2006, pp. 193–200.

[8] C. J. C. Burges, From RankNet to LambdaRank to LambdaMART: An Overview, Technical Report MSR-TR-2010-82, Microsoft Research, 2010.

[9] Z. Cao, T. Qin, T.-Y. Liu, M.-F. Tsai, H. Li, Learning to rank: From pairwise approach to listwise approach, in: Proceedings of the 24th International Conference on Machine Learning (ICML), 2007, pp. 129–136.

[10] T. Qin, T.-Y. Liu, H. Li, A general approximation framework for direct optimization of information retrieval measures, Information Retrieval 13 (2010) 375–397.

[11] Z. Zhao, L. Hong, L. Wei, J. Chen, A. Nath, S. Andrews, A. Kumthekar, M. Sathiamoorthy, X. Yi, E. H. Chi, Recommending what video to watch next: A multitask ranking system, in: Proceedings of the 13th ACM Conference on Recommender Systems (RecSys), 2019, pp. 43–51.

[12] X. Ma, L. Zhao, G. Huang, Z. Wang, Z. Hu, X. Zhu, K. Gai, Entire space multi-task model: An effective approach for estimating post-click conversion rate, in: Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval, 2018, pp. 1137–1140.

[13] T. Joachims, A. Swaminathan, T. Schnabel, Unbiased learning-to-rank with biased feedback, in: Proceedings of the 10th ACM International Conference on Web Search and Data Mining (WSDM), 2017, pp. 781–789.

[14] Q. Ai, K. Bi, C. Luo, J. Guo, W. B. Croft, Unbiased learning to rank with unbiased propensity estimation, in: Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval, 2018, pp. 385–394.

[15] C. A. Gomez-Uribe, N. Hunt, The Netflix recommender system: Algorithms, business value, and innovation, ACM Transactions on Management Information Systems 6 (2015).

[16] L. Li, Z. A. Din, Z. Tan, S. London, T. Chen, A. H. Daptardar, MerRec: A large-scale multipurpose Mercari dataset for consumer-to-consumer recommendation systems, in: Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), 2025, pp. 2371–2382.

[17] X. Zhai, B. Mustafa, A. Kolesnikov, L. Beyer, Sigmoid loss for language image pre-training, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2023, pp. 11941–11952.

[18] H. Tsukagoshi, R. Sasano, Ruri: Japanese general text embeddings, arXiv preprint arXiv:2409.07737 (2024).

[19] A. Rusli, M. Cao, S. Ishimoto, S. Akiyama, M. Frenzel, Towards better search with domain-aware text embeddings for C2C marketplaces, in: AAAI 2026 Workshop on New Frontiers in Information

Retrieval, 2025. ArXiv:2512.21021.
[20] T. Sakai, Evaluating evaluation metrics based on the bootstrap, in: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2006, pp. 525–532.