An Efficient Framework for Whole-Page Reranking via Single-Modal Supervision

Zishuai Zhang[1,2], Sihao Yu[3], Wenyi Xie[3], Ying Nie[3]
Junfeng Wang[3], Zhiming Zheng[1,2], Dawei Yin[3], Hainan Zhang[1,2]

[1]School of Artificial Intelligence, Beihang University, Beijing, China
[2]Beijing Advanced Innovation Center for Future Blockchain and Privacy Computing, Beihang University
[3]Baidu Inc., Beijing, China

Correspondence: yush93@qq.com, zhanghainan@buaa.edu.cn

Abstract

The whole-page reranking integrates retrieval results from multiple modalities and is critical for user experience of search engines, yet it requires costly large-scale expert annotations due to the complexity of assessing cross-modal relevances. In this paper, we propose SMAR, a novel whole-page reranking framework that converts single-modal rankers into page-level guidance by constructing budget-aware candidates for cross modal annotations and distilling intra-modality preferences to align relevance scales across modalities. Specifically, we use pre-trained single-modal rankers to construct candidate pages for limited cross-modal annotation at the page level. The whole-page reranker is then trained on these samples, enforcing consistency with single-modal preferences to preserve intra-modal ranking quality. Experiments on the Qilin and CrossRank datasets demonstrate that SMAR reduces annotation costs by 70-90% while outperforming the fully-annotated reranking baselines. Further offline and online A/B tests confirm significant gains in both ranking metrics and user experience, validating the effectiveness and practical value of our approach in real-world search scenarios.

1 Introduction

Modern search engines integrate information from multiple modalities, including text, images, videos, and large language model (LLM) outputs (Wang et al., 2016; Zhang et al., 2018). A critical component of this process is whole-page reranking (Mao et al., 2024), which determines the final ordering of heterogeneous results on the search engine results page (SERP). Unlike single-modal ranking, whole-page reranking must jointly balance relevance, consistency, and user satisfaction across modalities, which makes it challenging. Existing whole-page reranking methods (Li et al., 2023, 2025; Mao et al., 2024) rely heavily on large-scale human annotations. However, obtaining high-quality page-level labels is costly, as annotators must compare heterogeneous results and resolve cross-modal trade-offs. Meanwhile, single-modal rankers have achieved strong performance in text (Devlin et al., 2019; Zhang et al., 2025b) and vision (Radford et al., 2021; Wang et al., 2024). However, directly fusing them often fails due to mismatched feature distributions and score scales (Zhang et al., 2025a). This raises a key question: **How can we exploit strong single-modal rankers to enhance whole-page reranking while minimizing reliance on costly page-level annotations?**

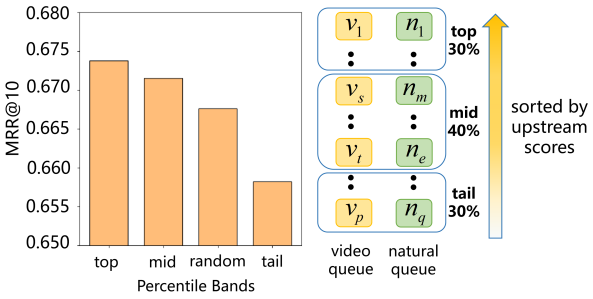

Figure 1: Performance of Percentile bands on Qilin. The video queue and the natural result queue are sorted by upstream model scores. "Random" denotes randomly selecting 30% of the data from each of the two queues.

To answer this question, we evaluate the performance of a multimodal reranking model on the Qilin dataset using four groups: a randomly selected 30%, the top 30%, middle 40%, and tail 30% from single-modal rankers, as shown in Figure 1. The results demonstrate that the higher the ranking relevance of the single-modal ranker, the better the final multimodal reranking model performs. This insight motivates a selective annotation strategy that prioritizes informative candidates.

In this paper, we propose SMAR, an efficient whole-page reranking framework that combines budget-aware page-level annotations (Top-P and Iso-label anchors) with low-cost single-modal preference signals. Top-P annotates only high-

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track), pages 966–978
July 2-7, 2026

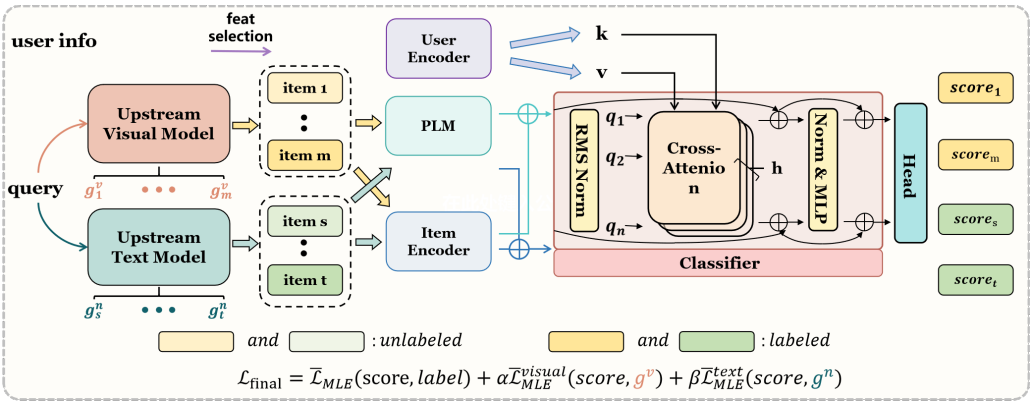

Figure 2: Overall architecture of the proposed user–item interaction model. ①Pretrained language model and an item feature extractor produce query semantics and item representations. ②User feature extractor produces user embeddings, which are injected into the classifier as keys and values. ③Cross-attention-based classifier models user–item interactions and produces relevance scores via MLP layers.

confidence items within each modality, and Iso-label anchors annotate cross-modal candidates at comparable upstream scores to align preference scales. They use limited human cross-modal labels for alignment while simultaneously exploiting abundant, low-cost upstream signals to enforce intra-modality consistency.

Experiments on Qilin and CrossRank show that SMAR cuts annotation cost by 70–90% while outperforming fully-annotated baselines. Moreover, online A/B tests further confirm consistent gains, yielding 0.86% and 0.25% gains in NDCG and CTR, as well as 1.58% and 0.33% gains in ΔGSB and next-day retained users. These results highlight the promise of aligning single-modal expertise with limited whole-page supervision to improve SERP quality in real-world search scenarios. Our contributions can be summarized as follows: [1]

- We identify the annotation bottleneck in whole-page reranking and show that strong single-modal rankers can substantially reduce page-level annotation costs.
- We propose two novel modal-wise relevance alignment strategies that distill single-modal signals into whole-page reranking by preserving intra-modality preference orderings.
- We validate SMAR through extensive offline and online evaluations and release a large-scale industrial dataset for future whole-page reranking research.

[1]Our code and data are available at https://github.com/zzs97str/SMAR. Arxiv version: https://arxiv.org/abs/2510.16803

2 Model

2.1 Preliminaries

In modern search engine systems, candidate items are often retrieved from multiple heterogeneous sources, such as text corpora, multimedia collections, or structured databases. Crucially, to address diverse user intents and preferences, these systems increasingly incorporate personalized retrieval mechanisms that adapt the selection process based on specific user contexts. Consequently, each source provides a distinct set of candidate items with different feature spaces and relevance distributions tailored to the inidividual user. Let $\mathcal{U}$ denote the user space, $\mathcal{Q}$ the query space, and $\{\mathcal{I}^1, \mathcal{I}^2, \ldots, \mathcal{I}^M\}$ the candidate item sets retrieved from M heterogeneous sources. Given a user $u \in \mathcal{U}$ and a query $q \in \mathcal{Q}$, the system first performs multimodal retrieval to construct a personalized unified candidate sets $\mathcal{I} = \bigcup_{m=1}^{M} \mathcal{I}^m$.

The goal of personalized heterogeneous ranking is to learn a multimodal scoring function

$$f : \mathcal{U} \times \mathcal{Q} \times \mathcal{I} \to \mathbb{R},$$

that assigns each item $i \in \mathcal{I}$ a relevance score $s_i = f(u, q, i)$, such that items from different modalities can be compared in a shared semantic space and ranked in descending order of relevance:

$$\pi(u, q) = \underset{i \in \mathcal{I}}{\text{argsort}} \ s_i,$$

where $\pi(u, q)$ denotes the final ranked list. Although this formulation is conceptually straightforward, learning an accurate scoring function $f(u, q, i)$ is highly non-trivial. The joint space

Method	Labeled Size	Bert-base-chinese			Qwen3-Reranker-4B		
		MRR@1	MRR@2	NDCG	MRR@1	MRR@2	NDCG
SFT	1% data	0.6097	0.7338	0.5762	0.6854	0.7858	0.6332
SFT	5% data	0.7078	0.7989	0.6538	0.7213	0.8071	0.6684
SFT	10% data	0.7388	0.8196	0.6840	0.7269	0.8099	0.6710
SFT	20% data	0.7366	0.8160	0.6817	0.7291	0.8142	0.6774
SFT	30% data	0.7418	0.8235	0.6921	0.7377	0.8207	0.6841
SFT	full data	0.7414	0.8218	0.6870	0.7537	0.8308	0.7008
Only Upstream	-	0.7418	0.8248	0.6864	0.7276	0.8170	0.6751
SMAR	1% data	0.7485	0.8285	0.6908	0.7347	0.8183	0.6778
SMAR	5% data	0.7534	0.8317	0.6942	0.7340	0.8160	0.6801
SMAR	10% data	0.7526	0.8326	0.6947	0.7459	0.8224	0.6863
SMAR	20% data	0.7522	0.8338	**0.6972**	0.7410	0.8224	0.6852
SMAR	30% data	**0.7586**	**0.8340**	0.6952	0.7425	0.8256	0.6913
SMAR	full data	0.7448	0.8285	0.6944	**0.7559**	**0.8341**	**0.7014**

Table 1: CrossRank: compares retrieval performance across data sizes. The SFT baseline fine-tunes the reranker on a random x% of queries without single-modal ranker supervision. x% of the dataset is treated at the query level, rather than item level.

$\mathcal{U} \times \mathcal{Q} \times \mathcal{I}$ is extremely large and high-dimensional. Training a reliable model over this space requires massive amounts of human-labeled data, which is expensive and leads to low efficiency in practice.

2.2 Single-modal Ranker Supervision

Let $\pi^m(q) = \{(i, g_i^m(q)) | i \in \mathcal{I}^m\}$ denote the items and their scores assigned by the upstream single-modal ranker g^m, where $g_i^m(q)$ is the score of item $i \in \mathcal{I}^{(m)}$ under query q. These scores serve as supervision signals for training the multimodal reranker $f(u, q, i)$.

Pairwise supervision. Given a query, for any two items $i, j \in \mathcal{I}^{(m)}$ from the same source, we define the pairwise preference (Wan et al., 2022; Burges et al., 2006) as:

$$y_{ij} = \mathbb{I}\big[g_i^m > g_j^m\big],$$

where $\mathbb{I}[\cdot]$ is the indicator function. The reranker is trained to preserve this preference by minimizing the pairwise loss:

$$\mathcal{L}_m = y_{ij} \cdot \max\big(0,\ \gamma - \big(f(u, q, i) - f(u, q, j)\big)\big),$$

where $\gamma > 0$ is the margin hyperparameter. The total multimodal reranker loss is

$$\mathcal{L}_{pair} = \sum_{m \in M} \beta_m \cdot \mathcal{L}_m,$$

where β_m is a coefficient of modality m.

Listwise supervision. Due to feature processing in upstream systems, exact relevance scores are often unavailable, while the relative ordering of items can still be reliably obtained. Accordingly, we adopt ListMLE (Xia et al., 2008), which directly models the permutation likelihood of a ranked list. Given a permutation π sorted by the upstream scores, the ListMLE loss is defined as:

$$\mathcal{L}_{\text{ListMLE}} = -\sum_{k=1}^{|\mathcal{I}^{(m)}|} \log \frac{\exp(f(u, q, \pi_k))}{\sum_{j=k}^{|\mathcal{I}^{(m)}|} \exp(f(u, q, \pi_j))}.$$

Importantly, the proposed supervision relies solely on relative preferences—that is, whether item A should be ranked above item B—rather than on their absolute scores. Even if an upstream ranker provides only coarse-grained signals, as long as a partial order exists, our strategies remain valid. For example, the upstream model may only return binned features rather than explicit model prediction scores. In such cases, the multimodal ranking model can still be supervised using pairwise comparisons constructed between items from different bins within a single modality. This design aligns with real-world industrial settings, where information silos across teams or organizations often limit access to raw data.

2.3 Annotation Strategy

Training multimodal rerankers using only single-modal data is inadequate, as single-modal rankers

capture modality-specific relevance but ignore cross-modal preferences and modality suitability. Thus, heterogeneous ranking requires high-quality cross-modal annotations. Given the high cost of large-scale multimodal labeling, we focus on annotating a small set of informative samples and propose two strategies: Top-P sampling and binary search for iso-label anchors.

Top-P Strategy Users expect to find a satisfactory item among the top entries with minimal effort (Chuklin et al., 2022). Guided by this insight, we adopt a budget-aware annotation policy that prioritizes the highest-exposure region of each modality list. For query q and modal m, an upstream ranker g_m produces a scored list $\mathcal{C}_m = \{(x_j, g_m(x_j))\}_{j=1}^{n_m}$. Given a budget ratio $p \in (0, 1]$, we form the labeled pool by taking the top $p \cdot n_m$ items from each $\mathcal{C}_m$. The remaining items are left unlabeled, supervised by their upstream ranker. This approach allows us to make the most of our limited resources and effectively enhance the performance of the multimodal reranker models in heterogeneous ranking scenarios.

Binary Search for Iso-Label Anchors Although feature distributions across different modalities vary for the same query, upstream single-modality models often accurately capture intra-modality preferences. To leverage this, we propose establishing "bridges" between modalities by identifying items that share identical prior satisfaction scores. We term this strategy *Binary Search for Iso-Label Anchors*. Its core objective is to efficiently locate candidate items in two distinct sorted queues that can serve as cross-modal anchors. More details are given in Appendix A.

2.4 Whole-Page Reranker

Prior context-aware ranking methods (Pobrotyn et al., 2020) primarily model inter-document relationships while ignoring user-specific signals, which are crucial in heterogeneous ranking scenarios. Thus, we incorporate user features via a cross-attention mechanism between user and item embeddings.

As shown in Figure 2, the encoder first derives semantic representations $\mathbf{e}_i^{\text{sem}}$ using a pretrained multimodal language model. Following Zhang et al. (2025a), we adopt a hybrid early–late fusion strategy to integrate visual and textual information. Early fusion projects visual tokens into the text embedding space, while late fusion applies a learnable gating mechanism to adaptively combine visual and textual representations:

$$z = \sigma(W[e^{visual}, e^{text}] + b), \quad (1)$$

$$e_i^{\text{sem}} = z \odot e^{visual} + (1 - z) \odot e^{text}. \quad (2)$$

Here, σ is the sigmoid function, $[\cdot, \cdot]$ denotes vector concatenation, and $\odot$ is the element-wise product.

In parallel, structured item and user features are encoded via feed-forward networks:

$$\mathbf{e}_i^{\text{feat}} = \text{FFN}(x_i), \mathbf{e}^{\text{user}} = \text{FFN}(x_u)$$

where both x_i and x_u are bucketized and binarized feature vectors (Pan et al., 2024). The final item representation is formed by concatenating semantic and feature embeddings.

To model personalized relevance, we apply a multi-head cross-attention module where each item embedding attends to the user embedding:

$$\mathbf{h}_i = \text{Multi-head CrossAttn}(\mathbf{e}_i, \mathbf{e}^{user}, \mathbf{e}^{user}).$$

The output representation $\mathbf{h}_i$ with a residual connection (He et al., 2016) is then passed through an MLP head to produce the final ranking score. We optimize a length-normalized ListMLE objective. For unlabeled queries, we further incorporate listwise distillation losses from upstream visual and text rankers. The final training objective is:

$$\mathcal{L}_{\text{final}} = \overline{\mathcal{L}}_{\text{MLE}} + \alpha \overline{\mathcal{L}}_{\text{MLE}}^{v} + \beta \overline{\mathcal{L}}_{\text{MLE}}^{n},$$

where α and β balance label supervision and upstream supervision.

3 Experiments

3.1 Experimental Settings

Dataset To evaluate our approach, we adopt the public Qilin dataset (Chen et al., 2025) and the CrossRank dataset. Qilin is sourced from a major Chinese e-commerce platform, containing 1.9 million user-generated notes and 5 million images, together with app-level session contexts from over 15,000 users. With its diverse multimodal content and rich user interactions, Qilin provides a challenging yet realistic benchmark for studying ranking over multimodal heterogeneous data. The CrossRank dataset is a real-world expert-labeled dataset (May 2024–March 2025) with >70K queries and 300K results. These datasets act as benchmarks for heterogeneous ranking in web search scenarios. The details of CrossRank are in Appendix C.

Method	Labeled Size	Qwen2-VL-2B		
		MRR@10	MAP@10	NDCG
SFT	0%-10% data	0.6269	0.4699	0.4172
SFT	0%-20% data	0.6338	0.4913	0.4428
SFT	0%-30% data	0.6606	0.5016	0.4533
SFT	full data	0.6664	0.5180	0.4693
Only Upstream	-	0.6172	0.4541	0.4038
SMAR	random 30%	0.6676	0.5125	0.4659
SMAR	0%-10%	0.6702	**0.5218**	**0.4756**
SMAR	0%-20%	0.6701	0.5184	0.4691
SMAR	0%-30%	**0.6738**	0.5203	0.4722
SMAR	30%-70%	0.6715	0.5135	0.4670
SMAR	70%-100%	0.6582	0.4647	0.4507
SMAR	full data	0.6687	0.5061	0.4600
SMAR	0%-20%	**0.6701**	**0.5184**	**0.4691**
SMAR	20%-40%	0.6677	0.5023	0.4564
SMAR	40%-60%	0.6676	0.5062	0.4556
SMAR	60%-80%	0.6684	0.4912	0.4433
SMAR	80%-100%	0.6419	0.4805	0.4307

Table 2: Qilin: we test top-p sampling across budgets, where x%–y% is the slice of labeled instances by the upstream model's ranking scores. We fine-tune SMAR on this selected data with human labels plus upstream supervision; the SFT baseline fine-tunes the Reranker only on the top-x% data without single-modal ranker supervision.

Metrics We evaluate our reranking methods using MRR, MAP, and NDCG metrics for the main experiments. For the offline and online A/B testing, we additionally compare F1, PNR, and the user experience metrics, such as IRQ, Next-day Retained User, and ΔGSB. Positive ΔGSB indicates a net human preference for our model. The details of the metrics are in the Appendix E.

Settings The server is equipped with a Gigabit Ethernet card and utilizes multiple GPUs, including eight NVIDIA A100 and eight NVIDIA V100. FP32 is used to maximize performance for Bert-base-chinese and the online reranker, while FP16 and lora (Hu et al., 2022) are used to accelerate the training for Qwen2-VL-2B and Qwen3-Reranker-4B. Other parameter settings are provided in the Appendix D.

3.2 Main Results

Qilin Dataset We simulate industrial candidate generation by merging two modality-specific retrievers, a BERT-base-Chinese (Devlin et al., 2019) text retriever and a Qwen-VL-2B (Zhang et al., 2025b) vision retriever, into a unified multimodal candidate pool. Under a limited cross-modal annotation budget, we distill their modality-specific ranking preferences into a unified reranker to approximate the performance of full supervision.

Experimental results on the Qilin Dataset are reported in Table 2. Incorporating supervision from upstream single-modality scores consistently improves model performance on annotated cross-modal data. When the annotation budget is limited, training on top-ranked quantile segments yields superior performance compared to random sampling or annotating lower-ranked candidates. Notably, with only the top 10% of annotated data, SMAR outperforms the vanilla model trained on the full annotation set without single-modality supervision across all evaluation metrics.

The performance degradation caused by including the bottom 70%-100% of candidates indicates that a smaller, high-quality subset is more effective than the full dataset. This can be attributed to two factors. First, tail candidates suffer from strong position bias and limited exposure, leading to sparse and biased interaction signals. Second, their low upstream scores suggest an intrinsic misalignment with user intents. As a result, incorporating these noisy samples introduces interference, ultimately harming ranking performance.

Method	MRR@10	MAP@10	NDCG
MLP	0.6021	0.4652	0.4089
self-attention (Pobrotyn et al., 2020)	0.6168	0.4785	0.4251
cross-attention(w/o hybrid fusion)	0.6633	0.4996	0.4579
cross-attention	**0.6664**	**0.5180**	**0.4693**

Table 3: Whole-page reranker with different attention modules on Qilin Dataset. Training excludes the upstream-ranker supervision.

Method	T_{round}	Labeled Size	F1	NDCG@4	pnr
Baseline	-	-	0.7258	0.8766	1.885
Vanilla	-	100%	0.7211	0.8788	**2.1527**
binary search	1	42.09%	0.7134	0.8803	2.0619
binary search	2	58.62%	**0.7269**	0.8791	2.1492
SMAR	-	100%	0.7266	**0.8842**	2.1342

Table 4: Offline Metrics of Binary Search for Iso-Label Anchors.

CrossRank dataset In the CrossRank dataset, upstream signals are provided by independently developed in-house ranking systems. Each query contains an average of 4.4 candidates, making top-p sampling inapplicable. Consequently, we apply x% sampling at the query level as a Top-P variant, requiring at least two natural and two video candidates per query, balancing top-ranked items across modalities, and filtering out queries with maximum relevance scores below 0.8. BERT-Base-Chinese is trained for 30 epochs, while Qwen3-Reranker is trained for 15 epochs.

Strategy	Test Split	MRR@4	MAP@4	NDCG
Top-P	Head	0.740	0.661	0.617
	Middle	0.625	0.548	0.499
	Tail	0.661	0.593	0.545
	All	0.659	0.485	0.473
Tail-P	Head	0.677	0.608	0.565
	Middle	0.626	0.546	0.504
	Tail	0.669	0.597	0.551
	All	0.643	0.463	0.450

Table 5: Performance comparison of Top-P vs. Tail-P strategies across different test splits.

Results on CrossRank are reported in Table 1. SMAR consistently improves performance across different model architectures, including both the encoder-based BERT-Base-Chinese and the decoder-only Qwen3-Reranker, demonstrating the robustness of our approach. Notably, BERT-Base-Chinese trained with only 30% of labeled data outperforms its fully supervised counterpart, whereas Qwen3-Reranker benefits from single-modality supervision but does not yet exceed full-data training.

Binary Search for Iso-label Anchors Results of the binary search for iso-label anchors are reported in Table 4. With two rounds of binary search, SMAR achieves strong performance in terms of F1 and NDCG while using only 58.62% of the original dataset. Moreover, its PNR score closely matches that of the fully supervised model, with a difference of less than 0.01.

Cross-Attention Mechanism We evaluate the proposed cross-attention mechanism on the Qilin dataset. Results are reported in Table 3. The cross-attention architecture incorporating user features consistently outperforms all baselines, and jointly adopting early and late fusion for item representation further improves performance.

3.3 Analysis

Robustness Analysis We evaluate the generalization of Top-P beyond high-scoring candidates across head (top 30%), middle (30%–70%), and tail (bottom 30%) subsets defined by upstream scores. As shown in Table 5, Top-P achieves performance on the middle and tail subsets comparable to Tail-P, with differences below 1% across all metrics, indicating no degradation in generalization to lower-scoring candidates. Moreover, Top-P significantly outperforms Tail-P on the head subset, supporting its effectiveness as a practical trade-off between annotation cost and ranking quality.

Method	MRR@10	MAP@10	NDCG@10	α-$NDCG$@10
KD	0.664	0.510	0.629	0.659
PL	0.667	0.510	0.630	0.662
SMAR	0.674	0.521	0.641	0.674

Table 6: Main Results comparing SMAR with semi-supervised baselines.

Performance and Diversity Analysis To evaluate SMAR under limited-annotation settings, we compare it with two strong semi-supervised baselines using the same 30% labeled data and student architecture, trained for 3 epochs with a learning rate of 1×10^{-5}. For knowledge distillation (KD), a larger teacher model is trained on the 30% labeled subset and used to supervise the student. The teacher has 12 layers and a hidden size $1.5\times$ that of the student. For pseudo-labeling (PL), the same teacher generates labels for the remaining 70% unlabeled data, and the student is trained on the combined human- and pseudo-labeled set.

Results in Table 6 show that SMAR consistently outperforms both KD and PL. Leveraging modality-specific experts provides more faithful preference signals than distillation from a single teacher trained with limited supervision, reducing teacher-induced bias. We further evaluate ranking diversity using α-NDCG@10 with $\alpha = 0.5$ under standard subtopic-aware evaluation. SMAR achieves the best α-NDCG@10, demonstrating superior performance on both standard ranking metrics and diversity-aware evaluation.

3.4 Online A/B Testing

Settings To fully evaluate the performance of SMAR, we adopt pairwise supervision to distill the upstream single-modal ability into the multimodal reranker, where candidates with higher upstream scores are treated as positives and those with lower scores as negatives. The loss function is defined as

$$\begin{aligned} loss &= loss_{pointwise} + \alpha \cdot loss_{label\ pairwise} \\ &\quad + \beta \cdot loss_{upstream\ pairwise} \\ &= loss(pred, label) \\ &\quad + \alpha \cdot max\left(margin_1 - (s_{pos} - s_{neg}), 0\right) \\ &\quad + \beta \cdot max\left(margin_2 - (s_{pos} - s_{neg}), 0\right), \end{aligned} \tag{3}$$

where $\alpha = 0.5$, and $\beta = 0.2$. For online evaluation, we conduct a seven-day A/B test (from 2025-09-16 to 2025-10-01) comparing SMAR with the online relevance reranker in a real-world search system.

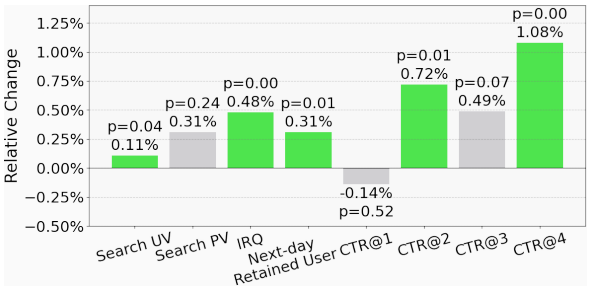

Figure 3: **Online A/B test.** Bars report the relative change (%) of each metric for the treatment vs. baseline. Color encodes statistical significance: *green* denotes a significant improvement ($p \leq 0.05$ and $\Delta > 0$) and *gray* denotes a non-significant effect ($p > 0.05$).

Results Figure 3 reports relative lifts over the baseline. The treatment improves key scale metrics, with Search UV increasing by 0.11% (p=0.04), Search PV by 0.31% (p=0.24), and IRQ, the total quantity of results issued, by 0.48% ($p \approx 0.00$), alongside a 0.31% gain in next-day retention (p=0.01), indicating improved user engagement and perceived relevance. Although CTR@1 slightly decreases, this is attributable to LLM-generated answer cards ranked at the top position, which often satisfy user intent without click-through. In contrast, the treatment increases mid- and tail-rank click propensity, with aggregated CTR over top-4 and top-8 improving by 0.16% and 0.25%, respectively. Expert evaluations further show consistent gains in ΔGSB on both randomly sampled (1.58%) and long-tail queries (0.50%), demonstrating that SMAR outperforms the latest online model.

4 Related Work

4.1 Learning to Rank

Learning to Rank (LTR) has been extensively studied in search and recommendation systems, and existing methods generally fall into three paradigms: pointwise, pairwise, and listwise. Pointwise approaches (Guo et al., 2017) treat ranking as a regression or classification task, predicting each item's relevance independently. Pairwise methods (Wan et al., 2022; Burges et al., 2006) optimize relative preferences by training on positive–negative item pairs, improving discriminative capability. Listwise approaches further consider the entire ranking list, optimizing list-level metrics and modeling item dependencies more effectively. Attention mechanism, as a listwise approach, is designed for explicitly capturing inter-item relationships. Pobrotyn et al. (Pobrotyn et al., 2020) introduced a context-aware listwise model using self-attention to model pairwise interactions within a candidate list. However, such models often overlook personalization, assuming uniform user preferences. To address this, we propose Cross-Attention-Rank, which incorporates the user features via cross-attention, aligning user embeddings with item features for personalized ranking.

4.2 Multimodal Ranking

Recent work in multimodal ranking explores integrating visual and textual signals through Multimodal Large Language Models (MLLMs). While these models aim for unified multimodal representations, fine-tuning often favors textual modalities, weakening visual contributions. To mitigate this bias, NoteLLM-2 (Zhang et al., 2025a) introduces multimodal in-context learning (mICL) with late fusion, separating visual and textual prompts and applying a gating mechanism to effectively preserve visual information, leading to superior performance in multimodal recommendation. Similarly, the Joint Relevance Estimation (JRE) framework (Zhang et al., 2018) fuses screenshots, HTML, and text of search result pages using inter- and intra-modality attention to balance heterogeneous signals. However, JRE primarily addresses intra-page reranking and does not resolve the scoring misalignment that arises when jointly ranking heterogeneous modality items lacking shared objectives.

5 Conclusion

We introduced SMAR, an annotation-efficient whole-page reranking framework that uses Top-P and Iso-label anchors to focus limited annotations, preserving intra-modality order while improving cross-modal ranking ability. Experiments on Qilin and CrossRank show that SMAR cuts annotation costs by 70–90% and delivers consistent gains in NDCG, CTR, and user engagement in offline evaluations and online A/B tests. These results highlight that coupling single-modal expertise with budget-aware alignment is a practical and scalable path to higher-quality SERP. Future work includes adaptive distillation that responds to intent and distribution shifts, broader modality coverage (e.g., code, agents), and deeper integration with LLM-generated outputs to enhance the user experience.

Acknowledgments

This work was funded by the National Natural Science Foundation of China (NSFC) under Grants No. U25B2070 and No. 62406013, the Beijing Advanced Innovation Center Funds for Future Blockchain and Privacy Computing (GJJ-24-034), and the Fundamental Research Funds for the Central Universities. This work was conducted during an internship at Baidu, Inc. The author thanks Baidu, Inc. for the research platform and academic guidance.

References

Gavin Brown, Adam Pocock, Ming-Jie Zhao, and Mikel Luján. 2012. Conditional likelihood maximisation: a unifying framework for information theoretic feature selection. *The journal of machine learning research*, 13(1):27–66.

Christopher Burges, Robert Ragno, and Quoc Le. 2006. Learning to rank with nonsmooth cost functions. *Advances in neural information processing systems*, 19.

Jia Chen, Qian Dong, Haitao Li, Xiaohui He, Yan Gao, Shaosheng Cao, Yi Wu, Ping Yang, Chen Xu, Yao Hu, and 1 others. 2025. Qilin: A multimodal information retrieval dataset with app-level user sessions. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3670–3680.

Aleksandr Chuklin, Ilya Markov, and Maarten De Rijke. 2022. *Click models for web search*. Springer Nature.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.

Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. Deepfm: a factorization-machine based neural network for ctr prediction. *arXiv preprint arXiv:1703.04247*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Canjia Li, Xiaoyang Wang, Dongdong Li, Yiding Liu, Yu Lu, Shuaiqiang Wang, Zhicong Cheng, Simiu Gu, and Dawei Yin. 2023. Pretrained language model based web search ranking: From relevance to satisfaction. *arXiv preprint arXiv:2306.01599*.

Yuchen Li, Hao Zhang, Haojie Zhang, Hengyi Cai, Xinyu Ma, Shuaiqiang Wang, Haoyi Xiong, Zhaochun Ren, Maarten de Rijke, and Dawei Yin. 2025. Fultr: A large-scale fusion learning to rank dataset and its application for satisfaction-oriented ranking. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 5583–5594.

Haitao Mao, Lixin Zou, Yujia Zheng, Jiliang Tang, Xiaokai Chu, Jiashu Zhao, Qian Wang, and Dawei Yin. 2024. Whole page unbiased learning to rank. In *Proceedings of the ACM Web Conference 2024*, pages 1431–1440.

Junwei Pan, Wei Xue, Ximei Wang, Haibin Yu, Xun Liu, Shijie Quan, Xueming Qiu, Dapeng Liu, Lei Xiao, and Jie Jiang. 2024. Ads recommendation in a collapsed and entangled world. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5566–5577.

Przemysław Pobrotyn, Tomasz Bartczak, Mikołaj Synowiec, Radosław Białobrzeski, and Jarosław Bojar. 2020. Context-aware learning to rank with self-attention. *arXiv preprint arXiv:2005.10084*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR.

Qi Wan, Xiangnan He, Xiang Wang, Jiancan Wu, Wei Guo, and Ruiming Tang. 2022. Cross pairwise ranking for unbiased item recommendation. In *Proceedings of the ACM web conference 2022*, pages 2370–2378.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Yue Wang, Dawei Yin, Luo Jie, Pengyuan Wang, Makoto Yamada, Yi Chang, and Qiaozhu Mei. 2016. Beyond ranking: Optimizing whole-page presentation. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, pages 103–112.

Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. 2008. Listwise approach to learning to rank: theory and algorithm. In *Proceedings of the 25th international conference on Machine learning*, pages 1192–1199.

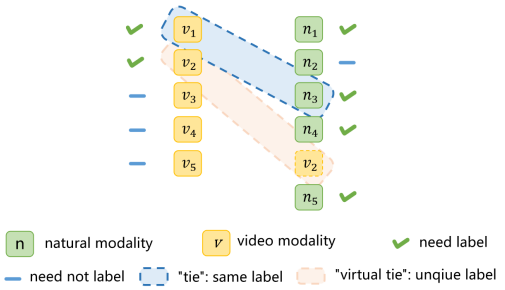

Figure 4: An example of binary search for iso-label anchors. Blue dashed boxes represent tie segments where items share the same label, and orange dashed boxes indicate virtual ties where no item in the natural queue has the same label as v_2, but natural items with both higher and lower labels exist, leading to the insertion of v_2 as a virtual tie.

Chao Zhang, Haoxin Zhang, Shiwei Wu, Di Wu, Tong Xu, Xiangyu Zhao, Yan Gao, Yao Hu, and Enhong Chen. 2025a. Notellm-2: Multimodal large representation models for recommendation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, pages 2815–2826.

Junqi Zhang, Yiqun Liu, Shaoping Ma, and Qi Tian. 2018. Relevance estimation with multiple information sources on search engine result pages. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 627–636.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025b. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*.

A Binary Search for Iso-Label Anchors

When an exact label match is unavailable, we introduce a "virtual tie" to maintain comparable ranking orders. The procedure is outlined in Algorithm 1.

Formally, let Q_1 and Q_2 denote two queues (e.g., video and natural results) sorted by their upstream scores. We iterate through items $i \in Q_1$ and seek a corresponding anchor in Q_2. Since Q_2 is sorted, we employ a binary search to find an index k such that $label(Q_1[i]) = label(Q_2[k])$. During this process, three scenarios arise, determining the construction of our alignment supervision.

- "Tie": If an item k is found in Q_2 such that their labels are identical, we establish a "tie" anchor (e.g., $\{v_1, n_3\}$ in Figure 4). These items are manually labeled and share identical annotations, allowing for direct pointwise supervision. More importantly, they serve as comparable points that align the scales of the upstream scores from different modalities.

- Monotonic Difference: If no exact match exists, and for all remaining items j in the search space of Q_2, the labels are strictly consistently larger or smaller than $label(Q_1[i])$ (i.e., $\forall j, label(Q_2[j]) > label(Q_1[i])$ or vise versa). In this case, we can directly apply pairwise supervision without creating a specific anchor.

- "Virtual Tie": If $label(Q_1[i])$ does not exist in Q_2 and Q_2 contains items with labels both higher and lower than $Q_1[i]$, we insert $Q_1[i]$ into Q_2 as a virtual tie. Importantly, the "Virtual Tie" acts as a necessary interpolation point. It represents a virtual anchor in-

Algorithm 1 Binary search for iso-label anchors

Input: Two modality queues Q_1 (video) and Q_2 (natural), sorted by upstream scores; maximum rounds T_{rounds}

Output: Cross-modality sequence $\mathcal{S}$ containing `tie`, `virtual_tie`, and `single` segments.

Initialize $\mathcal{S} \leftarrow [\,]$, and search index $i \leftarrow 0$ in Q_1
for $i = 1$ to $|Q_1|$ **do**
  $a \leftarrow Q_1[i]$
  $k \leftarrow$ **BinarySearch** in Q_2 for item with $Q_2[k]$.label $= a$.label
  if k found **then**
    Append $\texttt{tie}(a, Q_2[k])$ to $\mathcal{S}$
    $T_{\text{rounds}} += 1$
  else if there exist higher and lower labels around a.label in Q_2 **then**
    Append $\texttt{virtual\_tie}(a, a^{(\text{virtual})})$ to $\mathcal{S}$
    $T_{\text{rounds}} += 1$
  else if all labels in $Q_2[j:] > a.label$ **then**
    Append $concatenate(Q_2[j\ :], a)$ as `single` to $\mathcal{S}$
  else if all labels in $Q_2[j:] < a.label$ **then**
    Append $concatenate(a, Q_2[j\ :])$ as `single` to $\mathcal{S}$
  end if
  if rounds exceed T_{rounds} **then**
    Append remaining items as `single` segments; **break**
  end if
end for
return $\mathcal{S}$

dicating where item i would be ranked in Q_2 based on ground truth labels. This effectively segments the ranking space, allowing us to distinguish between results with known relative intra-modality orders and those requiring alignment.

As illustrated in Figure 4, these anchors divide the ranked lists into distinct segments: $S_1 = \{n_1, n_2\}$, $Tie_1 = \{v_1, n_3\}$, and $S_2 = \{n_4\}$. Within a single state (e.g., S_1), items belong to the same modality. Here, we distill preferences directly from the upstream model scores, as the relative ranking within a modality is assumed to be reliable. Since the labels in Ties and Virtual Ties are explicitly known by binary search, they can form pointwise and pairwise loss between items in adjacent states (e.g., pairing S_1 with Tie_1, or S_2 with Tie_2). By utilizing T search rounds with a larger T incorporating more data, this strategy effectively transforms limited labeled data into a robust structure of relative preferences. It aligns the multimodal label spaces using anchors while preserving the fine-grained ranking signals inherent in the upstream models.

B Data Processing

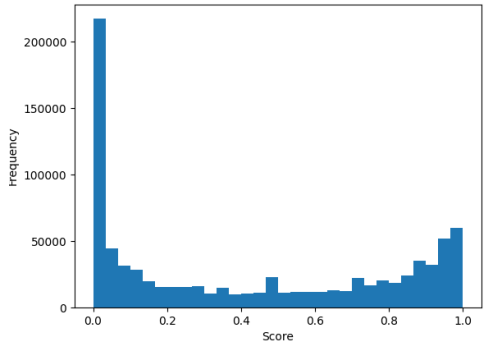

Figure 5: Distribution of Qwen3 Reranker scores for candidate items in the Qilin dataset. Scores are used to distinguish text-driven and image-driven clicks.

To facilitate both single-modality and mixed-modality ranking experiments, we reorganize the original Qilin dataset into dedicated training and test sets, ensuring that evaluation metrics reflect realistic ranking scenarios. In particular, we enforce that, for each query in the test set, the number of non-clicked items exceeds the number of clicked items, ideally by a factor of three. This design prevents artificially inflated MRR scores that could arise if all candidate items under a query are clicked.

For single-modality data, we split the training set according to text and image modalities. For textual data, each candidate document for a given query is first scored by the Qwen3 Reranker using the concatenated query-document input. Among the clicked items, those with a Qwen3 reranker similarity score below 0.1 are treated as image-driven clicks and are excluded from the textual training set. The threshold of 0.1 is set manually after careful inspection. Conversely, clicked items with a score above 0.1 are retained as positive samples. Non-clicked items, regardless of modality, are included as negative samples, forming a balanced training set for text ranking. For image-based ranking, since user clicks are naturally driven by both images and titles, we retain the original dataset without further filtering. These two datasets are independent for training models of different modalities.

The test set is processed in a similar fashion. For text ranking, clicked items with a low Qwen score are considered image-driven and their click label is set to 0, while clicked items with a high score retain a click label of 1. Non-clicked items are consistently labeled as 0. Image-based test data remain unchanged, as user interactions already reflect the joint effect of images and titles. For listwise objection, all candidate items are further sorted according to Wilson-smoothed click-through rates.

For mixed-modality ranking, we distinguish the modality of candidate items for each query. Clicked items with low textual relevance are treated as image-driven, while high textual scores are assumed to be text-driven, though potential image contributions are ignored. The ranking objective in this setting is to ensure that clicked items are placed above non-clicked ones, allowing the model to learn to integrate signals across modalities effectively.

Figure 5 presents the distribution of QwenReranker scores across candidate items, illustrating the separation between text- and image-driven clicks and providing guidance for subsequent model training and evaluation.

For the text modality, we concatenate each item's title and content as its textual representation. For the visual modality, the image title is incorporated into the visual-language model (VLM) prompt, while the image itself is used as input to the ViT component of Qwen-VLM.

Algorithm 2 Pair construction based on iso-label anchors

Input: Aligned sequence $\mathcal{S}$ with optional labels or upstream scores
Output: Pair set $\mathcal{P}$ for pairwise supervision, and pointwise set $\mathcal{P}^0$

$\mathcal{P} \leftarrow \emptyset$, $\mathcal{P}^0 \leftarrow \emptyset$
for all pairs (x, y) within or across queues in $\mathcal{S}$ **do**
  if x and y have labels **then**
    if x.label $>$ y.label **then**
      $\mathcal{P} \leftarrow \mathcal{P} \cup \{(x, y)\}$
    else
      $\mathcal{P} \leftarrow \mathcal{P} \cup \{(y, x)\}$
    end if
  else if labels unavailable and x, y from same queue **then**
    if x.upstream scores $>$ y.upstream scores **then**
      $\mathcal{P} \leftarrow \mathcal{P} \cup \{(x, y)\}$
    else
      $\mathcal{P} \leftarrow \mathcal{P} \cup \{(y, x)\}$
    end if
  end if
end for
for all item x with known label **do**
  $\mathcal{P}^0 \leftarrow \mathcal{P}^0 \cup \{(x, x)\}$ ▷ pointwise supervision
end for
return $(\mathcal{P}, \mathcal{P}^0)$

B.1 Feature Selection via Entropy-Based Sorting

To mitigate the high cost of manual annotation while preserving discriminative features, we employ entropy-based sorting for numerical feature selection (Brown et al., 2012). The entropy H(X) of each feature X is calculated to quantify its information content, with features ranked in descending order of entropy. This ensures that features with higher information gain (e.g., upstream scores) are prioritized over redundant ones. The final selected numerical features include 5 candidate features, 5 query features, and an upstream score, while categorical features like is_wenda are encoded as binary indicators.

For the cross-modality ranking model in CrossRank, we align upstream scores with human-annotated 5-point labels (0-4) to minimize the discrepancy between predicted and true rankings. The dataset is partitioned into training (70%), validation (15%), and test (15%) sets, ensuring sufficient candidate diversity to compute robust metrics like MRR and NDCG.

C CrossRank Dataset

The CrossRank dataset was constructed from real user interactions. It contains query data and high-quality annotations obtained through expert labeling. The dataset primarily consists of search results retrieved from online platform natural ranking channel and video ranking channel, with more than 200,000 video results and 40,000 natural results included. Each sample contains a query, the corresponding retrieved results with titles, summaries, categorizations, upstream model scores, and human-annotated relevance, etc. The CrossRank dataset comprises single-modality and multi-modality corpora. The single-modality corpus provides only a training set, where all results for a given query come from the same source, either natural search results or video results, and each query is paired with at least two results. The multi-modality corpus includes both training and test sets, and each query is associated with at least two results drawn from different modalities. Expert-annotated relevance is provided on a five-point scale ranging from 0 to 4, indicating the degree to which an item satisfies the user query. A score of 0.0 denotes that the item fails to meet the query at all, while a score of 4.0 denotes that the item fully satisfies the query. Owing to its high-quality human annotations, the CrossRank dataset enables search engines to better capture the relevance between user queries and retrieved results, thereby providing a valuable resource for multimodal ranking research.

D Parameter settings

In the Qilin experiments, we train for 30 epochs with a batch size of 1 at the query level, where each query contains multiple candidate items whose total number corresponds to the actual model input batch size. The maximum sequence length is set to 512, which is sufficient for nearly all items, and the learning rate is 1×10^{-5}.

In the CrossRank experiments, for the Qwen3-Reranker model, we train for 15 epochs with a learning rate of 2×10^{-6} and a batch size of 2, applying LoRA fine-tuning to both attention and MLP layers within the attention blocks. For the

BERT-Base-Chinese model, we train for 30 epochs with a batch size of 32, a maximum sequence length of 512, and a learning rate of 1×10^{-5}.

E Metrics

E.1 Offline Metrics

- **MRR@k** (Mean Reciprocal Rank)
  MRR@k measures how well the model ranks the *first relevant result* at a high position. Higher MRR indicates that users can find relevant items earlier. For a set of queries $\mathcal{Q}$, let rank_i denote the rank position of the first relevant item for query q_i. Then

$$\text{MRR@}k = \frac{1}{|\mathcal{Q}|} \sum_{i=1}^{|\mathcal{Q}|} \frac{\mathbb{I}(\text{rank}_i \le k)}{\text{rank}_i},$$

  where $\mathbb{I}(\cdot)$ is the indicator function.

- **MAP@k** (Mean Average Precision)
  MAP@k measures the model's ability to rank *all relevant items* higher, not just the first one. Higher MAP indicates better ranking consistency across relevant results. For query q_i,

$$P_i@j = \frac{1}{j} \sum_{t=1}^{j} \text{rel}_i(t).$$

  where,

$$\text{rel}_i(t) = \begin{cases} 1, & \text{if item at rank } t \text{ is relevant,} \\ 0, & \text{otherwise.} \end{cases}$$

  The average precision at cutoff k is

$$AP_i@k = \frac{1}{R_i} \sum_{j=1}^{k} P_i(j) \cdot \text{rel}_i(j),$$

  where $R_i = \sum_{j=1}^{k} \text{rel}_i(j)$. The mean average precision is

$$\text{MAP@}k = \frac{1}{|\mathcal{Q}|} \sum_{i=1}^{|\mathcal{Q}|} AP_i@k.$$

- **NDCG** (Normalized Discounted Cumulative Gain)
  NDCG measures the ability of the model to rank highly relevant items near the top, with emphasis on both position and graded relevance. For query q_i, the discounted cumulative gain is:

$$DCG_i = \sum_{j=1}^{N} \frac{2^{\text{rel}_i(j)} - 1}{\log_2(j+1)}$$

  where N is the number of retrieved items for query q_i and $\text{rel}_i(j)$ is the graded relevance of the item at rank j. The ideal DCG (IDCG) is computed by sorting items by true relevance. Then

$$\text{NDCG} = \frac{1}{|\mathcal{Q}|} \sum_{i=1}^{|\mathcal{Q}|} \frac{DCG_i}{IDCG_i}$$

- **F1** (Harmonic Mean of Precision and Recall)
  F1 evaluates the balance between precision and recall without relying on a cutoff. For query q_i, let $\mathcal{S}_i$ be the set of items returned by the system (e.g., the full ranked list or those above a fixed decision threshold), and let $\mathcal{R}_i$ be the set of ground-truth relevant items with $r_i = |\mathcal{R}_i|$. Define

$$P_i = \begin{cases} \frac{|\mathcal{S}_i \cap \mathcal{R}_i|}{|\mathcal{S}_i|}, & |\mathcal{S}_i| > 0, \\ 0, & \text{otherwise,} \end{cases}$$

  where

$$R_i = \frac{|\mathcal{S}_i \cap \mathcal{R}_i|}{r_i}.$$

  The per-query F1 score is

$$F1_i = \begin{cases} \dfrac{2\,P_i R_i}{P_i + R_i}, & P_i + R_i > 0, \\ 0, & \text{otherwise.} \end{cases}$$

  We report the macro-average over queries with at least one relevant item:

$$\text{F1} = \frac{1}{|\mathcal{Q}'|} \sum_{i \in \mathcal{Q}'} F1_i, \quad \mathcal{Q}' = \{\, i \mid r_i > 0 \,\}.$$

- **PNR** (Positive–Negative Ratio)
  PNR quantifies pairwise order consistency between model scores and ground-truth relevance. For query q_i with candidates indexed by $a \in \{1, \ldots, n_i\}$, let $y_{i,a}$ denote the ground-truth relevance (larger is more relevant) and $s_{i,a}$ the model score. Define the counts of concordant and discordant labeled pairs as

$$C_i = \sum_{a=1}^{n_i} \sum_{b=1}^{n_i} \mathbb{I}\big(y_{i,a} > y_{i,b}\big)\, \mathbb{I}\big(s_{i,a} > s_{i,b}\big), \tag{4}$$

$$D_i = \sum_{a=1}^{n_i} \sum_{b=1}^{n_i} \mathbb{I}\big(y_{i,a} > y_{i,b}\big)\, \mathbb{I}\big(s_{i,a} < s_{i,b}\big). \tag{5}$$

Label ties are excluded by construction; score ties are ignored. Then

$$\mathrm{PNR}_i = \frac{C_i}{D_i},$$

and the dataset-level metric is the macro-average

$$\mathrm{PNR} = \frac{1}{|\mathcal{Q}|} \sum_{i=1}^{|\mathcal{Q}|} \mathrm{PNR}_i.$$

Higher PNR indicates fewer pairwise inversions and stronger alignment with ground truth.

E.2 Online Metrics

Search UV counts the number of unique users who interacted with the search surface. **Search PV** counts the number of search page views. **Issued Result Quantity (IRQ)** is the number of results delivered to users and subsequently viewed. It is a key indicator of search scale and, to some extent, quantifies the amount of information contributed by the search engine. **Next-day Retained User** refers to the fraction of users who return the following day. For click metrics, **CTR@k** denotes the position-specific click-through rate at rank position k (clicks on position k divided by impressions of position k), for $k \in \{1, 2, 3, 4\}$. Relative to the online baseline, we obtain N expert pairwise judgments. Letting #good and #bad denote counts favoring our model vs. baseline, we define

$$\Delta\mathrm{GSB} = \frac{\#\mathrm{good} - \#\mathrm{bad}}{2 \times (\#\mathrm{good} + \#\mathrm{bad} + \#\mathrm{same})}.$$

Positive ΔGSB indicates a net human preference for our model.