DISTILL WHAT YOU RETRIEVE: CANDIDATE-SET ON-POLICY DISTILLATION WITH RELIABLE PREFERENCE CORRECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Dense retrieval is essential for LLM agents to identify relevant tools and reusable skills from large candidate collections. Knowledge distillation (KD) has become a widely adopted paradigm for transferring the capabilities of powerful teachers to compact student retrievers. However, conventional offline distillation relies on static candidate pools that cannot adapt to the student's evolving retrieval behavior. On-policy distillation (OPD) addresses this mismatch by collecting teacher supervision on student-induced candidates, yet its application to dense retrieval remains limited by two intrinsic issues: query-candidate-level supervision fails to capture the setwise competition that governs retrieval, and soft distribution matching does not explicitly guarantee that annotated positives outrank negatives. To address these issues, we propose a two-stage framework that combines candidate-set on-policy distillation with targeted preference optimization. In Stage 1, Candidate-Set On-Policy Distillation (CS-OPD) constructs query-specific Top-K candidate sets from the training student and transfers the teacher's listwise preferences over the same candidates via KL divergence. In Stage 2, Reliable-Disagreement Direct Preference Optimization (RD-DPO) identifies reliable residual ranking errors through teacher-validated margin disagreement and corrects them using a score-level DPO objective tailored to dense retrieval. Extensive experiments on ToolRet and SkillRet demonstrate consistent improvements over strong retrieval and distillation baselines. Further analyses confirm that CS-OPD improves candidate-set distribution alignment, whereas RD-DPO resolves residual ordering inconsistencies, highlighting their complementary contributions.

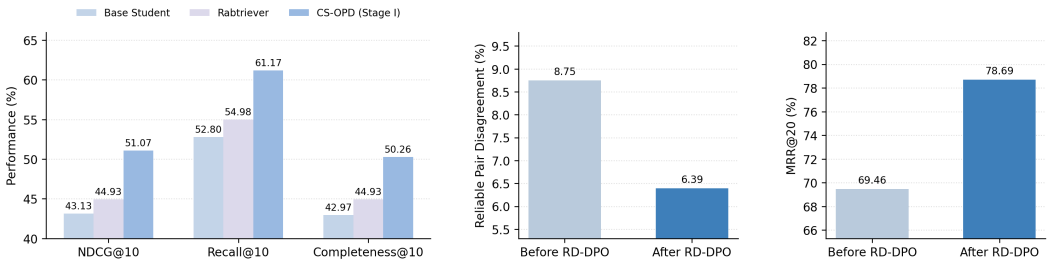


Figure 1: **Retrieval performance and residual ranking errors. Left:** Comparison of the base student, Rabtriever, and CS-OPD after Stage I on NDCG@10, Recall@10, and Completeness@10. **Middle–right:** Applying RD-DPO to the CS-OPD checkpoint reduces reliable pair disagreement from 8.75% to 6.39% and increases MRR@20 from 69.46% to 78.69%.

1 INTRODUCTION

As large language model (LLM) agents move beyond text generation toward complex real-world tasks, external tools and reusable skills have become essential for extending their capabilities (Schick

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.

et al., 2023; Shi et al., 2025; Kang et al., 2026). Embedding retrievers can pre-encode tool descriptions and perform efficient vector search, making them well suited for first-stage retrieval over large tool collections (Reimers & Gurevych, 2019; Karpukhin et al., 2020). Existing embedding retrievers are commonly trained with contrastive objectives such as InfoNCE using positive and negative examples (Karpukhin et al., 2020). Such supervision does not directly specify graded preferences among similar candidate tools. Knowledge distillation (KD) offers a way to transfer these relative relevance signals from a stronger embedding retriever to a smaller student while retaining efficient vector retrieval at inference time (Lin et al., 2021).

A common retrieval-distillation approach relies on a fixed candidate pool whose teacher scores or rankings are computed in advance (Hofstätter et al., 2020; Wang et al., 2022a). As the student evolves during training, however, its retrieved candidates and dominant errors also change. Consequently, supervision derived from a fixed candidate pool may gradually become misaligned with the student's current confusions. On-policy distillation (OPD) addresses an analogous mismatch by applying teacher supervision to outputs induced by the current student policy, rather than restricting training to a fixed offline distribution (Agarwal et al., 2024). This motivates conditioning retrieval distillation on the candidates selected by the current student.

Nevertheless, adapting OPD to dense retrieval requires reconciling the granularity of supervision with that of the retrieval decision. Existing adaptations, such as Rabtriever, optimize teacher supervision independently for each query–candidate pair (Chen et al., 2026). Because the resulting objective decomposes over candidates, it does not explicitly couple or normalize the relevance signals of candidates that compete within the same retrieved list. Although accurate pair-level matching may still recover the teacher's ranking, it provides no direct constraint on the teacher's relative relevance distribution over a student-induced candidate set. This distinction is particularly relevant to tool and skill retrieval, where closely related candidates often compete for the top positions. As shown in Figure 1 (left), Rabtriever improves over the base student on all three metrics, while CS-OPD achieves larger gains after Stage I. This result is consistent with the benefit of candidate-set-level alignment in our setting.

Candidate-set distribution alignment does not guarantee relevance-consistent ordering for every positive–negative pair. Although the KL objective matches the teacher's probability allocation over retrieved candidates, approximate agreement may still leave inversions among closely scored candidates. As shown in Figure 1 (middle), such residual disagreements remain after CS-OPD. Correcting them based on the student's ordering alone would be insufficient: a teacher preference that contradicts the annotated relevance relation cannot provide a label-consistent correction signal. RD-DPO therefore focuses on teacher-consistent residual inversions, where the teacher scores the annotated positive above the negative while the student reverses this ordering.

To address these two challenges, we propose a two-stage framework that combines candidate-set-level distribution distillation with pairwise ranking correction. The first stage, **Candidate-Set On-Policy Distillation (CS-OPD)**, aligns teacher and student relevance distributions over the top-K candidates retrieved by the student, transferring relative ranking information within the candidate sets induced by its retrieval behavior. The second stage, **Reliable Disagreement DPO (RD-DPO)**, targets residual positive–negative inversions for which the teacher's ordering agrees with the relevance annotations. The two stages are complementary: CS-OPD models competition within the retrieved set, whereas RD-DPO corrects local ordering errors that remain after distillation. As shown in Figure 1 (middle and right), RD-DPO reduces reliable-pair disagreement from 8.75% to 6.39% while increasing MRR@20 from 69.46% to 78.69%.

We evaluate our framework on ToolRet and SkillRet using 0.6B and 4B students. As shown in Table 1, our method achieves the best NDCG@10, Recall@10, and C@10 within both student-size groups. For the 0.6B student, it improves average NDCG@10 and Recall@10 by 25.00% and 20.82%, respectively, relative to the Rabtriever-based pair-level distillation baseline. Compared with the strongest offline distillation result for each metric, it achieves relative gains of 4.77% in average NDCG@10, 4.56% in Recall@10, and 6.43% in C@10, with the largest gain on C@10. On ToolRet, RD-DPO further reduces Query-RDR from approximately 46% to 38% and raises MRR@20 from 69.46% to 78.69%, while positive coverage@20 changes little. This pattern suggests that RD-DPO mainly corrects the ordering of relevant tools already retrieved. At inference time, only the student embedding retriever is retained, requiring no teacher inference.

In summary, our main contributions are as follows:

- We propose **CS-OPD**, which aligns teacher and student distributions over the Top-K candidates retrieved by the current student. This design adapts the distillation signal to the student's evolving retrieval behavior while modeling listwise competition among candidate tools.
- We propose **RD-DPO**, which constructs preference pairs from label-consistent teacher–student ranking disagreements and directly corrects the residual ranking reversals left by candidate-set-level distillation.
- We conduct extensive experiments across multiple tool-retrieval settings. The results and ablation studies demonstrate the complementary effects of candidate-set distribution distillation and pairwise preference optimization on retrieval coverage and ranking quality.

2 Related Work

2.1 Tool and Skill Retrieval for LLM Agents

Tool and skill retrieval aims to identify a small set of external capabilities relevant to a user request. General tool-use benchmarks, including ToolBench, API-Bank, ToolACE, MetaTool, and TaskBench, evaluate different aspects of tool selection and execution (Qin et al., 2024; Li et al., 2023; Liu et al., 2025; Huang et al., 2024; Shen et al., 2024). More recent benchmarks isolate retrieval as a standalone problem: ToolRet consolidates heterogeneous tool-use datasets for large-scale tool retrieval (Shi et al., 2025), while SkillRet evaluates retrieval from realistic libraries of reusable agent skills (Kang et al., 2026). Embedding retrievers encode queries and documents independently and are commonly trained with InfoNCE-style contrastive objectives that distinguish positive documents from sampled negatives (van den Oord et al., 2018). DPR uses labeled positives and in-batch negatives (Karpukhin et al., 2020), while ANCE mines hard negatives using a full-corpus index periodically updated with the current retriever (Xiong et al., 2021). Contriever applies contrastive learning without relevance labels (Izacard et al., 2022), and E5 scales contrastive training with weakly supervised text pairs (Wang et al., 2022b).

2.2 Retriever Distillation

KD transfers the ranking capability of a strong teacher to a more efficient dense retriever. Existing approaches operate at different levels of supervision. MarginMSE matches the teacher's positive–negative score margins (Hofstätter et al., 2020), whereas TCT-ColBERT and EmbedDistill transfer representation-level knowledge from the teacher to the student (Lin et al., 2021; Kim et al., 2023). Other methods provide listwise supervision over groups of candidates (Qin et al., 2023; Tamber et al., 2025). Score-Only Distillation minimizes the mean squared error between row-centered teacher and student score vectors over each candidate set (Dubovikov et al., 2026) On-policy distillation instead collects supervision from the states or outputs induced by the current student (Agarwal et al., 2024). Rabtriever adapts this paradigm to dense retrieval by aligning teacher representations and relevance signals for individual query–document interactions (Chen et al., 2026). However, it does not directly align the teacher and student distributions over the full-corpus Top-K candidates retrieved together by the current student. Our method differs by allowing the student to retrieve a query-specific Top-K set from the complete catalog, using the teacher to rescore the same candidates, and aligning their normalized score distributions over this shared candidate support.

3 Method

In this section, we introduce our two-stage framework. We begin with the problem formulation (Section 3.1). We then present CS-OPD, which transfers teacher supervision over candidate sets retrieved by the student (Section 3.2). Finally, we describe RD-DPO, which selects teacher-supported residual ranking errors using student-side difficulty filtering and optimizes the resulting preference pairs (Section 3.3).

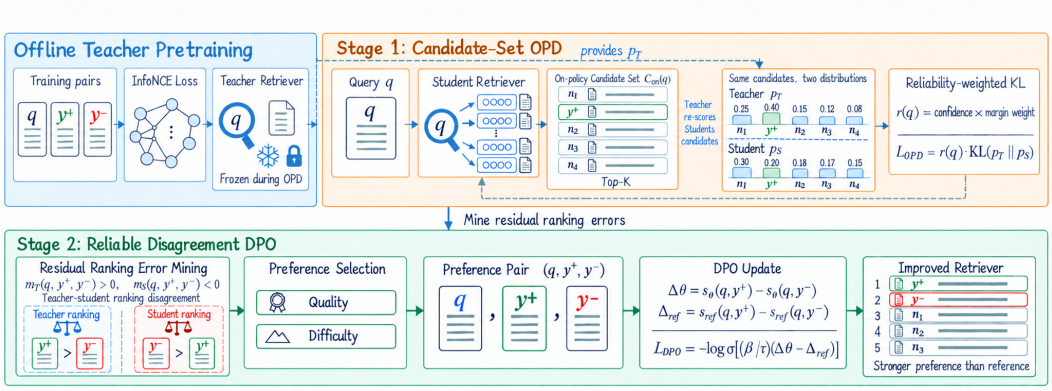


Figure 2: Overview of the proposed two-stage CS-OPD and RD-DPO framework.

3.1 Problem Formulation

Let $\mathcal{C} = \{c_j\}_{j=1}^{M}$ denote a catalog of M candidates, where each candidate represents either a tool or a reusable agent skill. Let $\mathcal{D} = \{(q_i, Y_i^+)\}_{i=1}^{N}$ denote the training set, where q_i is a user query and $Y_i^+ \subseteq \mathcal{C}$ is its set of relevant candidates. A dense retriever independently encodes queries and candidates into a shared embedding space. For a student retriever parameterized by θ, we define the cosine retrieval score as

$$s_\theta(q, c) = \left\langle \frac{f_\theta^q(q)}{\|f_\theta^q(q)\|_2}, \frac{f_\theta^c(c)}{\|f_\theta^c(c)\|_2} \right\rangle, \tag{1}$$

where f_θ^q and f_θ^c denote the query and candidate encoders, respectively. The frozen teacher, parameterized by ϕ, produces the corresponding score $s_\phi(q, c)$ analogously.

Given a query q and a candidate subset $C \subseteq \mathcal{C}$, the student and teacher scores induce two retrieval distributions over the same candidate support:

$$p_\theta(c \mid q, C) = \frac{\exp(s_\theta(q,c)/\tau)}{\sum_{c' \in C} \exp(s_\theta(q,c')/\tau)}, \qquad p_\phi(c \mid q, C) = \frac{\exp(s_\phi(q,c)/\tau)}{\sum_{c' \in C} \exp(s_\phi(q,c')/\tau)}, \tag{2}$$

where τ is the distillation temperature. Our goal is to transfer the teacher's relative relevance structure over query-specific candidate sets while preserving the retrieval efficiency of the compact student.

3.2 Stage 1: Candidate-Set On-policy Distillation

Student-induced candidate set. Distillation over a fixed, externally constructed set of negatives may overlook the candidates that the student actually ranks highly. We therefore allow the student to determine the candidate support on which distillation is performed. Specifically, for each query q, we construct the candidate set using the student's Top-K retrieval results:

$$C_{\mathrm{on}}(q; \theta) = \mathrm{TopK}_{c \in \mathcal{C}}\, s_\theta(q, c). \tag{3}$$

This candidate set represents the student's retrieval neighborhood and contains the alternatives that currently compete most strongly under its scoring function. Performing distillation on this student-induced support therefore concentrates teacher supervision on the ranking decisions exposed by the student's own retrieval behavior, rather than on a predetermined offline negative set. For the student-retrieved set C_q, the student and the frozen teacher score the same candidates and form the distributions in Eq. equation 2.

Reliability-weighted distribution alignment. A strong teacher is not necessarily reliable for every query. We weight each query's distillation loss using the teacher's distributional concentration

and its ordering of annotated positives relative to retrieved non-positives. Define

$$h_\phi(q) = 1 - \frac{H(p_\phi(\cdot \mid q, C_q))}{\log |C_q|}, \tag{4}$$

$$m_\phi(q) = \max_{y \in Y^+(q)} s_\phi(q, y) - \max_{c \in C_q \setminus Y^+(q)} s_\phi(q, c), \tag{5}$$

$$r(q) = h_\phi(q)\, \sigma\left(\frac{m_\phi(q) - m_0}{\tau_m}\right). \tag{6}$$

Here $H(\cdot)$ is entropy, while m_0 and τ_m control the margin-based weight. Annotated positives outside C_q are scored only to compute $r(q)$; they are not added to the distribution-matching support. The query-level CS-OPD objective is

$$\mathcal{L}_{\text{CS-OPD}}(q) = r(q)\, D_{\text{KL}}(p_\phi(\cdot \mid q, C_q) \,\|\, p_\theta(\cdot \mid q, C_q)), \tag{7}$$

with both teacher probabilities and $r(q)$ detached. Stage 1 minimizes the average loss over training queries:

$$\mathcal{L}_{\text{stage1}} = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}_{\text{CS-OPD}}(q_i). \tag{8}$$

3.3 Stage 2: Reliable Disagreement DPO

Residual ranking-error mining. Candidate-level distribution alignment improves retrieval but may leave local ranking inversions around relevant tools. Let θ_1 denote the Stage 1 checkpoint. For each query, the student retrieves a Top-K_{mine} list from the full catalog. For each annotated positive y^+ in this list, we select up to h nearest non-positive candidates ranked above it, excluding all other annotated positives for the query. If c_j is the j-th nearest such candidate, the mined pool is

$$\mathcal{N}(q, y^+) = \{c_j\}_{j=1}^{h}, \tag{9}$$

where unavailable candidates are omitted. These candidates are ranked ahead of y^+ by the Stage 1 student and thus constitute local positive–negative inversions.

For a labeled positive $y^+ \in Y^+(q)$, we compute the student and teacher margins

$$m_S(q, y^+, y^-) = s_{\theta_1}(q, y^+) - s_{\theta_1}(q, y^-), \qquad m_T(q, y^+, y^-) = s_\phi(q, y^+) - s_\phi(q, y^-). \tag{10}$$

We define a residual disagreement as $m_S < 0$ and $m_T > 0$: the Stage 1 student still ranks the negative above the positive, whereas the frozen teacher supports the labeled ordering. Each retained disagreement forms a preference triple (q, y^+, y^-).

Data selection. Not every mined disagreement provides equally useful supervision. We select preference pairs along two dimensions. First, *quality* requires $m_T \geq \delta_T$, which removes pairs for which the teacher does not confidently support the annotation. Second, *difficulty* retains student margins within a data-dependent interval

$$Q_\ell(m_S) \leq m_S \leq \min\{Q_u(m_S), 0\}, \tag{11}$$

where Q_ℓ and Q_u are lower and upper quantiles. This retains residual inversions with intermediate-to-large negative student margins, while excluding the most extreme conflicts and the weak inversions closest to zero..

Score-Level DPO. Standard Direct Preference Optimization (DPO) increases the reference-relative likelihood of a preferred response y^+ over a dispreferred response y^- (Rafailov et al., 2023). Given a preference dataset $\mathcal{P}$, its objective is

$$\mathcal{L}_{\text{DPO}}^{\text{seq}}(\theta) = -\mathbb{E}_{(q, y^+, y^-) \sim \mathcal{P}} \log \sigma\left(\beta \left[\log \frac{\pi_\theta(y^+ \mid q)}{\pi_{\text{ref}}(y^+ \mid q)} - \log \frac{\pi_\theta(y^- \mid q)}{\pi_{\text{ref}}(y^- \mid q)}\right]\right), \tag{12}$$

where π_θ is the trainable policy, π_{ref} is a frozen reference policy, σ is the sigmoid function, and β controls the preference-update strength.

Dense retrievers produce relevance scores rather than sequence log-probabilities. For a candidate set C containing both y^+ and y^-, we define a score-induced distribution $p_\theta(c \mid q, C) \propto \exp(s_\theta(q,c)/\tau_s)$ for $c \in C$, where τ_s is the score temperature. The shared normalization term cancels in the pairwise log-odds:

$$\log \frac{p_\theta(y^+ \mid q, C)}{p_\theta(y^- \mid q, C)} = \frac{s_\theta(q, y^+) - s_\theta(q, y^-)}{\tau_s}. \tag{13}$$

Applying this identity to both the trainable student and the frozen reference model expresses the DPO objective in terms of their positive–negative score margins.

The trainable student is initialized from the Stage 1 checkpoint θ_1, and a frozen copy of the same checkpoint serves as the reference model, i.e., $\theta_{\text{ref}} = \theta_1$. For each preference pair selected by the reliable-disagreement criterion, we define

$$\Delta_\theta = \frac{s_\theta(q, y^+) - s_\theta(q, y^-)}{\tau_s}, \qquad \Delta_{\text{ref}} = \frac{s_{\theta_{\text{ref}}}(q, y^+) - s_{\theta_{\text{ref}}}(q, y^-)}{\tau_s}. \tag{14}$$

Let $\mathcal{P}_{\text{RD}}$ denote the preference pairs retained by the reliable-disagreement selection procedure. Our score-level DPO objective is

$$\mathcal{L}_{\text{RD-DPO}}(\theta) = -\mathbb{E}_{(q,y^+,y^-)\sim\mathcal{P}_{\text{RD}}} \log \sigma \left[\beta \left(\Delta_\theta - \Delta_{\text{ref}}\right)\right]. \tag{15}$$

Minimizing Eq. equation 15 encourages the updated student to express a stronger positive-over-negative score margin than the Stage 1 reference, rather than merely requiring $s_\theta(q, y^+) > s_\theta(q, y^-)$.

4 Experiments

4.1 Experimental Setup

Benchmarks and training data. We evaluate our method on ToolRet (Shi et al., 2025) and SkillRet (Cho et al., 2026). ToolRet contains 7.6K retrieval tasks and approximately 44K tools collected from 35 tool-calling datasets across Web APIs, code functions, and customized applications. We use the same training query–tool pairs as Tool-DE (Lu et al., 2026), while removing the `tool-prefix` field from the tool input. SkillRet contains approximately 16K public agent skills, 63K training instances, and 4.4K evaluation queries. We use its official training split. Its training and evaluation skill pools are disjoint, allowing us to measure generalization to unseen skills. We report NDCG@K, Recall@K, and Completeness@K($1[\text{R@}K = 1]$), denoted as N@K, R@K and C@K in this paper.

Baselines. We compare our method against three categories of baselines across the two benchmarks. General-purpose retrievers include BM25 (Robertson & Zaragoza, 2009), e5-small-v2 and e5-large-v2 (Wang et al., 2022b), NV-Embed-v1 (Lee et al., 2025), and the Qwen3-Embedding series (Zhang et al., 2025). Supervised task-specific baselines include Tool-Embed trained with Tool-DE (Lu et al., 2026) and the official SkillRet-Embedding checkpoints (Cho et al., 2026) at matched parameter scales. We also consider several distillation methods, including MarginMSE (Hofstätter et al., 2020), Score-Only KD (Dubovikov et al., 2026), and Rabtriever (Chen et al., 2026). For controlled comparisons, all training variants use the same student initialization, teacher, training data and perparameters whenever applicable. More details are deferred to Appendix A.

Implementation. We distill Qwen3-Embedding students of sizes 0.6B and 4B from the corresponding InfoNCE-trained 8B teachers, which remain frozen throughout both stages. For CS-OPD, we use student-retrieved top-10 candidate sets on ToolRet and top-30 candidate sets on SkillRet for teacher scoring; RD-DPO subsequently optimizes the selected residual ranking disagreements. Both stages are run on $8\times$ NVIDIA A800 GPUs. Full hyperparameters are deferred to Appendix B.1.

4.2 Main Results

Table 1 shows that our method achieves the best N@10, R@10, and C@10 within both student-size groups on ToolRet and SkillRet. Rabtriever is the most direct on-policy counterpart: it transfers

Table 1: Overall retrieval performance on ToolRet and SkillRet at $K = 10$. The best result within each student-size group is highlighted in **bold**. Methods marked with * are reproduced by us.

	ToolRet			SkillRet			Avg.		
Method	N@10	R@10	C@10	N@10	R@10	C@10	N@10	R@10	C@10
General-purpose retriever									
BM25	36.41	46.40	39.02	51.69	59.41	44.56	44.05	52.91	41.79
e5-small-v2	31.57	40.58	32.77	44.66	51.59	37.45	38.12	46.09	35.11
e5-large-v2	31.97	40.43	32.45	53.41	60.55	45.45	42.69	50.49	38.95
NV-Embed-v1	42.71	53.43	43.41	57.03	64.64	48.00	49.87	59.04	45.71
Qwen3-Embedding-8B	46.31	57.61	47.57	63.64	70.29	54.14	54.98	63.95	50.86
Teacher	53.60	64.05	52.42	86.44	93.25	88.11	70.02	78.65	70.27
0.6B student									
Zero-shot	43.13	52.80	42.97	60.64	66.48	50.16	51.89	59.64	46.57
InfoNCE	47.64	59.54	48.52	79.68	86.31	76.66	63.66	72.93	62.59
MarginMSE	48.18	58.49	48.36	78.82	85.62	75.70	63.50	72.06	62.03
Score-Only KD	48.84	59.51	48.84	78.57	84.89	74.13	63.71	72.20	61.49
Rabtriever*	44.93	54.98	44.93	61.87	69.98	53.60	53.40	62.48	49.27
CS-OPD + RD-DPO	**51.42**	**61.93**	**51.00**	**82.08**	**89.05**	**81.03**	**66.75**	**75.49**	**66.02**
4B student									
Zero-shot	45.54	57.36	47.27	58.90	66.29	49.89	52.22	61.83	48.58
InfoNCE	51.43	62.99	51.35	83.90	90.65	83.36	67.67	76.82	67.36
MarginMSE	50.82	61.62	50.51	83.09	89.95	82.14	66.96	75.79	66.33
Score-Only KD	52.56	63.28	51.75	83.03	89.92	82.08	67.80	76.60	66.92
Rabtriever*	47.23	57.93	47.42	60.29	70.10	54.14	53.74	63.93	50.67
CS-OPD + RD-DPO	**55.25**	**65.86**	**53.96**	**85.26**	**92.11**	**86.11**	**70.26**	**78.99**	**70.04**

teacher supervision independently for each query–candidate interaction. Its substantially lower performance on both benchmarks is consistent with the observation that the limitation does not directly constrain the relative relevance of candidates competing within the same retrieved list. CS-OPD instead aligns teacher and student distributions over that shared candidate set.

Relative to InfoNCE, our 0.6B student improves the unweighted cross-benchmark averages of N@10, R@10, and C@10 by 3.09, 2.56, and 3.43 percentage points, respectively; the corresponding gains for the 4B student are 2.59, 2.17, and 2.68 points. InfoNCE distinguishes annotated positives from negatives but does not provide the teacher's graded preferences among closely related candidates. On ToolRet, the 4B student's gains at the first rank reach 5.26 points in N@1 and 3.41 points in R@1, exceeding its corresponding @10 gains of 3.82 and 2.87 points. This pattern is consistent with improved ordering near the head of the list, where RD-DPO is designed to correct the residual positive–negative inversions left after CS-OPD.

Our method also outperforms the offline distillation baselines on all three average metrics at both model scales. MarginMSE matches teacher-specified positive–negative score margins, but does not directly transfer relations among the other candidates in a retrieved set. Score-Only KD does match score differences among all candidates in a row; its distinction from CS-OPD is that these rows are constructed before training rather than retrieved by the current student. By distilling over student-induced candidate sets and subsequently correcting reliable residual inversions, our framework addresses both the changing candidate support and the remaining ordering errors. Its largest advantage over the stronger offline result for each average metric occurs in C@10, with gains of 3.99 and 3.12 points for the 0.6B and 4B students, respectively.

4.3 Ablation Studies

We examine the contribution of each stage on ToolRet using a 0.6B student at $K = 10$. The ablations include *w/o RD-DPO*, which evaluates the CS-OPD checkpoint; *w/o CS-OPD*, which retains

Table 2: Stage-wise analysis on ToolRet at $K = 10$ using a 0.6B student. All results are reported as percentages.

Method	NDCG@10	Recall@10	Comp.@10
Stage I: Distillation paradigm			
Base Student	43.13	52.80	42.97
InfoNCE	47.64	59.54	48.52
Offline KD	48.83	59.52	48.49
On-policy pair-level KD	49.59	59.62	48.25
CS-OPD	51.07	61.17	50.26
Stage II: Preference correction			
CS-OPD (Stage-I checkpoint)	51.07	61.17	50.26
Continued CS-OPD	50.85	61.13	50.11
CS-OPD + standard DPO	50.73	60.94	49.85
CS-OPD + RD-DPO	51.42	61.93	51.00

preference optimization but omits candidate-set distillation; and *w/o Training*, the original student without task-specific training.

As shown in Figure 3, CS-OPD alone increases N@10 from 42.64 to 51.07 and R@10 from 52.77 to 61.17. These scores approach the teacher's 53.60 N@10 and 64.05 R@10 reported in Table 1, indicating that candidate-set distillation transfers much of the teacher's retrieval capability to the smaller student. Once the student has reached this level of performance, RD-DPO targets the remaining positive–negative ranking inversions among retrieved candidates. It further raises N@10 to 51.42 and R@10 to 61.93, relative gains of 0.69% and 1.24% over the CS-OPD checkpoint. In contrast, omitting CS-OPD reduces N@10 and R@10 by 8.63% and 7.17%, respectively, relative to the full method. The results distinguish the roles of the two stages: CS-OPD provides the principal transfer of retrieval capability, while RD-DPO makes a smaller, targeted improvement to the already trained student's ranking.

4.4 Stage-wise Analysis

Table 2 examines the two training stages on ToolRet using a 0.6B student. Stage I compares contrastive training and three distillation strategies; Stage II compares preference correction with continued distillation and standard DPO. Implementation details are provided in Appendix B.2.

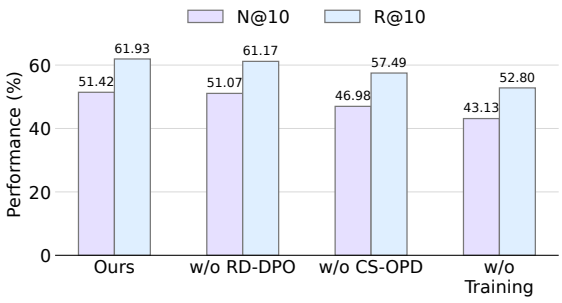


Figure 3: Ablation results on ToolRet at $K = 10$.

In Stage I, offline KD improves NDCG@10 over InfoNCE but leaves Recall@10 and Comprehensiveness@10 nearly unchanged. On-policy pair-level KD yields a further NDCG gain, yet its effects on the other two metrics remain limited. CS-OPD instead improves all three metrics, reaching 51.07, 61.17, and 50.26, respectively. It exceeds on-policy pair-level KD by 1.48, 1.55, and 2.01 percentage points. Both methods use student-retrieved candidates, but CS-OPD aligns their relative relevance within each query-specific set. The comparison therefore suggests that adapting the candidate pool is not sufficient on its own; the granularity of the distillation objective also matters.

In Stage II, RD-DPO raises NDCG@10, Recall@10, and Comprehensiveness@10 by 0.35, 0.76, and 0.74 points over the CS-OPD checkpoint. Continued CS-OPD and standard DPO do not improve any of the three metrics. The former comparison suggests that the gain is not simply a consequence of additional training, while the latter highlights the importance of selecting which ranking errors receive preference supervision. Together, these results support the complementary roles of candidate-set alignment and targeted correction of residual inversions.

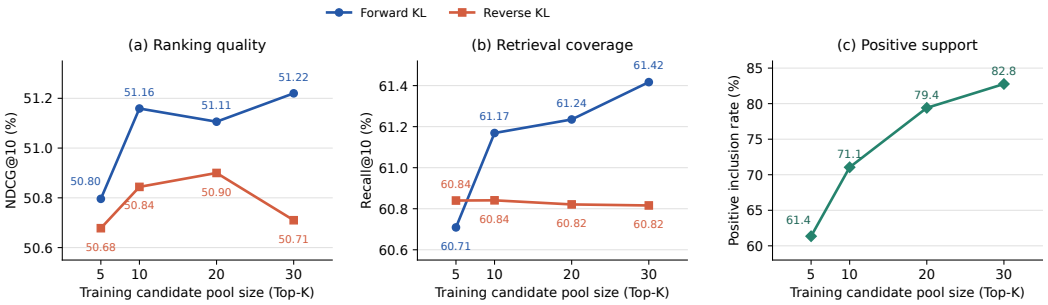


Figure 4: Sensitivity to candidate-pool size K and KL direction on ToolRet with a 0.6B student: (a) NDCG@10, (b) Recall@10, and (c) positive inclusion rate averaged over KL directions.

4.5 Sensitivity Analysis of CS-OPD

Figure 4 investigates how the size of the student-generated candidate pool and the direction of KL divergence affect CS-OPD. We vary the candidate pool size $K \in \{5, 10, 20, 30\}$ while keeping all other training configurations unchanged. We compare forward KL, $D_{\mathrm{KL}}(p_T \| p_S)$, with reverse KL, $D_{\mathrm{KL}}(p_S \| p_T)$, and report N@10 and R@10 after one training epoch.

As shown in Figures 4(a) and 4(b), forward KL benefits more consistently from enlarging the candidate pool. Its N@10 improves from 50.80 at $K = 5$ to 51.22 at $K = 30$, while R@10 increases from 60.71 to 61.42. In comparison, reverse KL exhibits a non-monotonic trend. Its N@10 reaches the highest value of 50.90 at $K = 20$, but decreases to 50.71 at $K = 30$. R@10 remains almost unchanged across different candidate-pool sizes, varying only between 60.82 and 60.84. Consequently, the performance gap between forward and reverse KL becomes more pronounced as K increases. At $K = 30$, forward KL outperforms reverse KL by 0.51% in N@10 and 0.60% in R@10.

Figure 4(c) provides a mechanism-level explanation for the effect of candidate-pool size. Increasing K substantially raises the probability that a labeled positive is included in the student-generated distillation support. The positive inclusion rate increases from 61.35% at $K = 5$ to 71.05%, 79.41%, and 82.77% at $K = 10$, 20, and 30, respectively. A larger candidate pool therefore exposes the teacher to more positive-containing and hard-negative contexts, providing richer supervision for transferring the teacher's local ranking behavior. However, the gain in positive coverage gradually saturates: the increase is 9.70% from $K = 5$ to $K = 10$, 8.36 points from $K = 10$ to $K = 20$, and only 3.36% from $K = 20$ to $K = 30$.

The different responses of forward and reverse KL can be explained by their distribution-matching properties. Forward KL is mode covering: candidates assigned high probability by the teacher directly contribute to the optimization even when the student currently assigns them a low probability. It can therefore make effective use of the additional teacher-supported candidates introduced by a larger K. Reverse KL is mode-seeking and weights candidates according to the student's current distribution. Newly introduced candidates with low student probability contribute relatively little to its gradient, even if the teacher considers them important. Enlarging the candidate pool may also introduce more weakly relevant or ambiguous candidates, which explains why reverse KL stops improving and eventually degrades at $K = 30$.

5 Conclusion

In this paper, we propose CS-OPD and RD-DPO, which together form a two-stage framework for transferring ranking knowledge from a strong teacher to a compact dense retriever. CS-OPD distills listwise supervision over student-induced candidate sets, while RD-DPO corrects residual ranking errors using reliable preference pairs. Extensive experiments on ToolRet and SkillRet demonstrate consistent improvements over strong dense-retrieval and distillation baselines across ranking and coverage metrics. By combining candidate-set distribution alignment with targeted preference correction, our framework enables effective and efficient tool and skill retrieval for LLM-based agents.

AI USE STATEMENT

We used generative AI tools to brainstorm and refine research ideas, obtain feedback on the methodology and experimental design, and assist in interpreting experimental results. We also used these tools to discover and summarize related literature, suggest manuscript structure, draft and polish portions of the paper, and assist with visualization code. We did not use generative AI to generate synthetic datasets or prove mathematical claims. The authors reviewed and revised AI-assisted outputs and made the final decisions regarding the methods, experiments, citations, and reported results. We take responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The main text specifies the CS-OPD and RD-DPO objectives and the preference-pair construction procedure. Appendix B provides the model configurations, training hyperparameters, and evaluation protocol for our experiments. We will release the training code, configuration files, and preference-pair construction scripts upon acceptance.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/5be69a584901a26c521c2b51e40a4c20-Abstract-Conference.html.

Teng Chen, Sheng Xu, Feixiang Guo, Xiaoyu Wang, Qingqing Gu, Hongyan Li, and Luo Ji. Efficient rationale-based retrieval: On-policy distillation from generative rerankers based on JEPA, 2026. URL https://arxiv.org/abs/2604.23336.

Hongcheol Cho, Ryangkyung Kang, and Youngeun Kim. Skillret: A large-scale benchmark for skill retrieval in llm agents. *arXiv preprint arXiv:2605.05726*, 2026.

Kirill Dubovikov, Martin Takáč, and Salem Lahlou. Score-only distillation for compact dense retrieval, 2026. URL https://arxiv.org/abs/2607.11465.

Sebastian Hofstätter, Sophia Althammer, Michael Schröder, Mete Sertkan, and Allan Hanbury. Improving efficient neural ranking models with cross-architecture knowledge distillation, 2020. URL https://arxiv.org/abs/2010.02666.

Yue Huang, Jiawen Shi, Yuan Li, Chenrui Fan, Siyuan Wu, Qihui Zhang, Yixin Liu, Pan Zhou, Yao Wan, Neil Zhenqiang Gong, and Lichao Sun. MetaTool benchmark for large language models: Deciding whether to use tools and which to use. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=R0c2qtalgG.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*, 2022. URL https://openreview.net/forum?id=jKN1pXi7b0.

Ryangkyung Kang, Hongcheol Cho, and Youngeun Kim. SkillRet: A large-scale benchmark for skill retrieval in LLM agents, 2026. URL https://arxiv.org/abs/2605.05726.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781, 2020. doi: 10.18653/v1/2020.emnlp-main.550. URL https://aclanthology.org/2020.emnlp-main.550/.

Seungyeon Kim, Ankit Singh Rawat, Manzil Zaheer, Sadeep Jayasumana, Veeranjaneyulu Sadhanala, Wittawat Jitkrittum, Aditya Krishna Menon, Rob Fergus, and Sanjiv Kumar. EmbedDistill: A geometric knowledge distillation for information retrieval, 2023. URL https://arxiv.org/abs/2301.12005.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. NV-Embed: Improved techniques for training LLMs as generalist embedding models. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/c4bf73386022473a652a18941e9ea6f8-Abstract-Conference.html.

Minghao Li, Yingxiu Zhao, Bowen Yu, Feifan Song, Hangyu Li, Haiyang Yu, Zhoujun Li, Fei Huang, and Yongbin Li. Api-bank: A comprehensive benchmark for tool-augmented llms. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 3102–3116, 2023.

Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. In-batch negatives for knowledge distillation with tightly-coupled teachers for dense retrieval. In *Proceedings of the 6th Workshop on Representation Learning for NLP*, pp. 163–173. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.repl4nlp-1.17. URL https://aclanthology.org/2021.repl4nlp-1.17/.

Weiwen Liu, Xu Huang, Xingshan Zeng, Xinlong Hao, Shuai Yu, Dexun Li, Shuai Wang, Weinan Gan, Zhengying Liu, Yuanqing Yu, Zezhong Wang, Yuxian Wang, Wu Ning, Yutai Hou, Bin Wang, Chuhan Wu, Xinzhi Wang, Yong Liu, Yasheng Wang, Duyu Tang, Dandan Tu, Lifeng Shang, Xin Jiang, Ruiming Tang, Defu Lian, Qun Liu, and Enhong Chen. ToolACE: Winning the points of LLM function calling. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=8EB8k6DdCU.

Xuan Lu, Haohang Huang, Rui Meng, Yaohui Jin, Wenjun Zeng, and Xiaoyu Shen. Tools are underdocumented: Simple document expansion boosts tool retrieval. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=g9D9MgG7iW.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, Sihan Zhao, Lauren Hong, Runchu Tian, Ruobing Xie, Jie Zhou, Mark Gerstein, Dahai Li, Zhiyuan Liu, and Maosong Sun. ToolLLM: Facilitating large language models to master 16000+ real-world APIs. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=dHng2O0Jjr.

Zhen Qin, Rolf Jagerman, Rama Kumar Pasumarthi, Honglei Zhuang, He Zhang, Aijun Bai, Kai Hui, Le Yan, and Xuanhui Wang. RD-Suite: A benchmark for ranking distillation. In *Advances in Neural Information Processing Systems*, volume 36. Curran Associates, Inc., 2023. doi: 10.52202/075280-1551. URL https://papers.nips.cc/paper_files/paper/2023/hash/701eba0f98c6f28ffee0de5969d8d034-Abstract-Datasets_and_Benchmarks.html.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992, 2019. doi: 10.18653/v1/D19-1410. URL https://aclanthology.org/D19-1410/.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009. doi: 10.1561/1500000019.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language

models can teach themselves to use tools. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://papers.nips.cc/paper/2023/hash/d842425e4bf79ba039352da0f658a906-Abstract-Conference.html.

Yongliang Shen, Kaitao Song, Xu Tan, Wenqi Zhang, Kan Ren, Siyu Yuan, Weiming Lu, Dongsheng Li, and Yueting Zhuang. TaskBench: Benchmarking large language models for task automation. In *Advances in Neural Information Processing Systems*, volume 37, pp. 4540–4574, 2024. doi: 10.52202/079017-0148. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/085185ea97db31ae6dcac7497616fd3e-Abstract-Datasets_and_Benchmarks_Track.html.

Zhengliang Shi, Yuhan Wang, Lingyong Yan, Pengjie Ren, Shuaiqiang Wang, Dawei Yin, and Zhaochun Ren. Retrieval models aren't tool-savvy: Benchmarking tool retrieval for large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 24497–24524, Vienna, Austria, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.findings-acl.1258. URL https://aclanthology.org/2025.findings-acl.1258/.

Manveer Singh Tamber, Suleman Kazi, Vivek Sourabh, and Jimmy Lin. Teaching dense retrieval models to specialize with listwise distillation and LLM data augmentation, 2025. URL https://arxiv.org/abs/2502.19712.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Kexin Wang, Nandan Thakur, Nils Reimers, and Iryna Gurevych. GPL: Generative pseudo labeling for unsupervised domain adaptation of dense retrieval. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2345–2360, 2022a. doi: 10.18653/v1/2022.naacl-main.168. URL https://aclanthology.org/2022.naacl-main.168/.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*, 2022b. URL https://arxiv.org/abs/2212.03533.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=zeFrfgyZln.

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. Qwen3 Embedding: Advancing text embedding and reranking through foundation models, 2025. URL https://arxiv.org/abs/2506.05176.

A Baselines

A.1 General-Purpose Retrievers

BM25. BM25 provides a lexical retrieval baseline, ranking tool or skill documents by their term-level match with the query without neural encoding or task-specific training.

Pretrained embedding models. We evaluate e5-small-v2 and e5-large-v2 (Wang et al., 2022b), NV-Embed-v1 (Lee et al., 2025), and the Qwen3-Embedding-8B (Zhang et al., 2025). These models encode queries and candidate documents independently and rank the resulting embeddings by cosine similarity. They are evaluated without fine-tuning on ToolRet or SkillRet.

A.2 Student and Teacher References

Zero-shot students. The zero-shot rows evaluate the Qwen3-Embedding-0.6B and Qwen3-Embedding-4B checkpoints without task-specific training.

InfoNCE. We fine-tune each student scale with an InfoNCE-style contrastive objective using annotated positive query–document pairs and negative examples. This baseline measures the effect of supervised retrieval training without teacher distillation.

Teacher. The teacher row reports the InfoNCE-trained Qwen3-Embedding-8B model used to supervise the distillation methods. It is distinct from the off-the-shelf Qwen3-Embedding-8B row.

A.3 Offline Distillation Baselines

MarginMSE. MarginMSE (Hofstätter et al., 2020) trains the student to match the teacher's score difference between a positive and a negative candidate. It transfers relative pairwise scores rather than a distribution over the retrieved candidate set.

Score-Only KD. Score-Only KD (Dubovikov et al., 2026) matches row-centered teacher and student score vectors over candidate rows. Centering removes a row-wide score offset while retaining the score differences among candidates.

A.4 On-Policy Distillation Baseline

Rabtriever. Rabtriever (Chen et al., 2026) transfers information from a generative reranker to a retriever through on-policy distillation. Its original formulation uses a JEPA-style predictor to approximate contextualized teacher representations, together with an auxiliary logit-distillation objective. In our comparison, Rabtriever* uses the same frozen InfoNCE-trained Qwen3-Embedding-8B teacher as the other distillation methods. It is therefore an adaptation to a shared embedding-teacher setting rather than an exact reproduction of the original generative-reranker configuration.

B Implementation Details

B.1 Training Hyperparameters

Table 3 summarizes the hyperparameters used for the experiments on ToolRet and SkillRet. The two benchmarks use the same configuration except for the size of the candidate set retrieved in CS-OPD. The teacher remains frozen throughout training. Query and tool embeddings are truncated to the specified dimension, ℓ_2-normalized, and scored by cosine similarity. For CS-OPD, the checkpoint obtained at the end of training is used for evaluation. During RD-DPO, checkpoints are saved every 15 steps, and the checkpoint with the lowest validation loss is selected for evaluation. All models use the same query formatting, tool catalog, and retrieval protocol. Queries and tools are encoded independently, and their normalized embeddings are ranked by cosine similarity.

B.2 Stage-wise Comparison Protocol

Table 2 compares training paradigms on ToolRet using a Qwen3-Embedding-0.6B student. All rows use the same training and evaluation splits, tool catalog, query and tool formatting, embedding dimension, and full-catalog retrieval protocol. Distillation methods use the same frozen teacher. The base student is evaluated without task-specific training.

Stage I: Distillation paradigm. We compare CS-OPD with three training alternatives under the same student initialization and ToolRet evaluation protocol. InfoNCE trains the student using annotated query–tool relevance pairs. *Offline Candidate KD* retrieves a Top-K candidate set for each query once using the initial student and keeps this support fixed throughout training. It distills the frozen teacher's score distribution over that fixed set. *On-policy Pair KD* instead pairs each annotated positive with the current student's highest-scoring non-positive tool. It forms teacher and student binary distributions over the resulting pair and aligns them using forward KL, $D_{\mathrm{KL}}(p_T^{\mathrm{pair}} \| p_S^{\mathrm{pair}})$.

Table 3: Hyperparameters for the ToolRet and SkillRet experiments with a 0.6B student. Except for the CS-OPD candidate-set size, the listed settings are shared across the two benchmarks.

Hyperparameter	Value
Shared configuration	
Embedding dimension	1,024
Maximum sequence length	2,048
Precision / GPUs	bfloat16 / 8
Optimizer	AdamW
Weight decay	0.01
Training epochs	1 per stage
Gradient clipping	1.0
Gradient checkpointing	Enabled
CS-OPD	
Retrieved candidate set	Top-10 (ToolRet); Top-30 (SkillRet)
KL direction	Forward, $\mathrm{KL}(p_T \| p_S)$
Distillation temperature τ	0.05
KL loss weight	1.0
Learning rate	3×10^{-6}
LR schedule / warmup	Cosine / 0
Batch size (device / accumulation / global)	4 / 8 / 256
Distributed optimization	DeepSpeed ZeRO Stage 2
RD-DPO	
Candidate pool for pair mining	Top-20
Negatives above	Up to 5
Teacher–student margin criteria	$m_T > 0,\ m_S < 0$
Student-margin percentiles	10th–50th
Reference model	Frozen Stage-1 checkpoint
Score temperature τ_s	0.05
DPO coefficient β	0.01
Negatives per training record	1
Learning rate	1×10^{-6}
LR schedule / warmup	Cosine / 0
Batch size (device / accumulation / global)	1 / 4 / 32

Stage II: Preference correction. All Stage-II methods start from the same Stage-I CS-OPD checkpoint. Continued CS-OPD resumes candidate-set distillation. For the two DPO variants, the Stage-I student retrieves the Top-20 tools from the full catalog and mines up to five non-positive tools ranked above each annotated positive; other annotated positives for the same query are excluded from the negative pool. Standard DPO trains on the resulting mined pairs without preference filtering. RD-DPO retains pairs for which the teacher prefers the positive ($m_T > 0$) but the Stage-I student ranks it below the negative ($m_S < 0$), followed by the 10th–50th percentile filter on student margins. Both DPO variants use the same score-level objective and frozen Stage-I reference model.

C Theoretical Analysis of Reliable Disagreement DPO

This appendix provides additional theoretical motivation for the second stage of our framework. We first show why candidate-set distribution alignment does not necessarily eliminate every local positive–negative inversion. We then formulate reliable disagreement selection and derive our score-level DPO objective from standard DPO by defining a local retrieval policy over each selected candidate pair. Finally, we analyze its optimization behavior, compare it with alternative pairwise objectives, and clarify the scope of the theoretical claims.

C.1 Why Distribution Alignment Can Leave Local Inversions

Stage 1 aligns the teacher and student distributions over a shared candidate set. For a query q and candidate set C, let

$$p(c) = p_\phi(c \mid q, C), \qquad r(c) = p_{\theta_1}(c \mid q, C) \tag{16}$$

denote the teacher distribution and the Stage 1 student distribution, respectively. Here, θ_1 denotes the student parameters after Candidate-Set On-Policy Distillation.

Consider an annotated positive $y^+ \in C$ and a negative $y^- \in C$. Suppose that the teacher supports the annotated ordering, with probability gap

$$\gamma = p(y^+) - p(y^-) > 0. \tag{17}$$

Although minimizing $D_{\mathrm{KL}}(p\|r)$ encourages the complete student distribution to approach the teacher distribution, it does not directly impose a separate hard constraint on every candidate pair. The following result characterizes the amount of distributional deviation required for a pairwise inversion.

Proposition 1: KL alignment and pairwise inversions. If the student reverses the teacher ordering,

$$r(y^+) \le r(y^-), \tag{18}$$

then

$$\mathrm{TV}(p, r) \ge \frac{\gamma}{2}, \qquad D_{\mathrm{KL}}(p\|r) \ge \frac{\gamma^2}{2}, \tag{19}$$

where TV denotes total variation distance.

Proof. Define

$$d^+ = p(y^+) - r(y^+), \qquad d^- = p(y^-) - r(y^-). \tag{20}$$

Using Eqs. equation 17 and equation 18, we obtain

$$d^+ - d^- = \big[p(y^+) - p(y^-)\big] - \big[r(y^+) - r(y^-)\big] \tag{21}$$

$$\ge \gamma. \tag{22}$$

By the triangle inequality,

$$|d^+| + |d^-| \ge |d^+ - d^-| \ge \gamma. \tag{23}$$

Therefore,

$$\mathrm{TV}(p, r) = \frac{1}{2} \sum_{c \in C} |p(c) - r(c)| \tag{24}$$

$$\ge \frac{1}{2} \left(|d^+| + |d^-|\right) \tag{25}$$

$$\ge \frac{\gamma}{2}. \tag{26}$$

Applying Pinsker's inequality,

$$\mathrm{TV}(p, r) \le \sqrt{\frac{1}{2} D_{\mathrm{KL}}(p\|r)}, \tag{27}$$

gives

$$D_{\mathrm{KL}}(p\|r) \ge 2\,\mathrm{TV}(p, r)^2 \ge \frac{\gamma^2}{2}. \tag{28}$$

This completes the proof.

Interpretation. Proposition 1 shows that sufficiently accurate distribution alignment can rule out an inversion when the teacher probability gap is large. In particular,

$$D_{\mathrm{KL}}(p\|r) < \frac{\gamma^2}{2} \quad \Longrightarrow \quad r(y^+) > r(y^-). \tag{29}$$

However, this condition becomes increasingly demanding when γ is small. For semantically similar tools or skills, the teacher may assign similar probabilities to multiple candidates, making $\gamma^2/2$ very small. Consequently, a student can achieve close global distributional agreement while still retaining a small number of local inversions among nearby candidates.

This observation does not imply that candidate-set distribution alignment is ineffective. Rather, it identifies a distinction between global distribution matching and explicit local ordering correction. Stage 1 transfers the teacher's listwise structure over the complete candidate set, whereas Stage 2 targets the residual inversions that remain consequential for ranking.

C.2 Reliable Disagreement Selection

Not every teacher–student disagreement should be treated as a valid preference target. The teacher may be uncertain, or its ordering may contradict the annotated relevance relation. We therefore use both the ground-truth annotation and the teacher ordering to identify reliable residual errors.

For an annotated positive y^+ and a negative y^- retrieved for query q, define the teacher and Stage 1 student score margins as

$$m_\phi(q, y^+, y^-) = s_\phi(q, y^+) - s_\phi(q, y^-), \tag{30}$$

$$m_1(q, y^+, y^-) = s_{\theta_1}(q, y^+) - s_{\theta_1}(q, y^-). \tag{31}$$

The reliable-disagreement set is

$$\begin{aligned} \mathcal{P}_{\mathrm{RD}} = \big\{(q, y^+, y^-) \;\big|\; & y^+ \in Y^+(q), \\ & y^- \notin Y^+(q), \\ & m_\phi(q, y^+, y^-) > \delta_{\mathrm{T}}, \\ & m_1(q, y^+, y^-) < -\delta_{\mathrm{S}}\big\}. \end{aligned} \tag{32}$$

where $\delta_{\mathrm{T}} \geq 0$ controls the required teacher reliability and $\delta_{\mathrm{S}} \geq 0$ controls the required student disagreement. When only the signs of the two margins are used, both thresholds are set to zero.

The three conditions in Eq. equation 32 serve different purposes. First, the annotation determines the desired preference direction. Second, the teacher margin removes pairs for which the teacher does not provide supporting evidence. Third, the negative student margin restricts training to actual residual inversions rather than repeatedly optimizing already-correct pairs. Reliable disagreement selection therefore uses the teacher as a preference validator, not as the reference model in the subsequent DPO objective.

C.3 Score-Level DPO as a Local Retrieval-Policy Objective

Standard DPO is formulated using sequence log-probabilities. A dense retriever does not produce autoregressive sequence probabilities, but its scores naturally define a categorical retrieval policy over a candidate set. For a selected pair $\{y^+, y^-\}$, define the local two-candidate retrieval policy

$$\pi_\theta^{(2)}\left(c \mid q, \{y^+, y^-\}\right) = \frac{\exp\left(s_\theta(q, c)/\tau_s\right)}{\sum_{c' \in \{y^+, y^-\}} \exp\left(s_\theta(q, c')/\tau_s\right)}, \tag{33}$$

where $\tau_s > 0$ is the score temperature.

The log-odds assigned to the positive over the negative are

$$\log \frac{\pi_\theta^{(2)}(y^+ \mid q)}{\pi_\theta^{(2)}(y^- \mid q)} = \log \frac{\exp(s_\theta(q, y^+)/\tau_s)}{\exp(s_\theta(q, y^-)/\tau_s)} \tag{34}$$

$$= \frac{s_\theta(q, y^+) - s_\theta(q, y^-)}{\tau_s} \tag{35}$$

$$\Delta_\theta. \tag{36}$$

Likewise, for a frozen reference model θ_{ref},

$$\log \frac{\pi^{(2)}_{\mathrm{ref}}(y^+ \mid q)}{\pi^{(2)}_{\mathrm{ref}}(y^- \mid q)} = \Delta_{\mathrm{ref}}, \tag{37}$$

where

$$\Delta_{\mathrm{ref}} = \frac{s_{\theta_{\mathrm{ref}}}(q, y^+) - s_{\theta_{\mathrm{ref}}}(q, y^-)}{\tau_s}. \tag{38}$$

The standard DPO loss for a preferred output y^+ and a rejected output y^- is

$$\mathcal{L}_{\mathrm{DPO}} = -\log \sigma \left(\beta \left[\log \frac{\pi^{(2)}_{\theta}(y^+ \mid q)}{\pi^{(2)}_{\mathrm{ref}}(y^+ \mid q)} - \log \frac{\pi^{(2)}_{\theta}(y^- \mid q)}{\pi^{(2)}_{\mathrm{ref}}(y^- \mid q)} \right] \right). \tag{39}$$

Rearranging the log-ratios and applying Eq. equation 36 yields

$$\mathcal{L}_{\mathrm{DPO}} = -\log \sigma \left[\beta \left(\Delta_\theta - \Delta_{\mathrm{ref}} \right) \right]. \tag{40}$$

Therefore, our score-level DPO objective is not merely a heuristic replacement of sequence probabilities with retrieval scores. Equation equation 40 is exactly the standard DPO objective applied to the local two-candidate softmax policy induced by the retriever scores.

In our implementation, the trainable model is initialized from the Stage 1 checkpoint,

$$\theta \leftarrow \theta_1, \tag{41}$$

and a frozen copy of the same checkpoint is used as the reference:

$$\theta_{\mathrm{ref}} = \mathrm{stopgrad}(\theta_1). \tag{42}$$

The teacher ϕ and reference model θ_{ref} thus play different roles. The teacher determines whether a disagreement is reliable, whereas the reference model measures how the student's pairwise preference changes relative to its Stage 1 initialization.

C.4 Optimization Behavior

To examine how score-level DPO corrects a selected inversion, define

$$z = \beta \left(\Delta_\theta - \Delta_{\mathrm{ref}} \right), \qquad \ell(z) = -\log \sigma(z). \tag{43}$$

The derivative with respect to the trainable score gap is

$$\frac{\partial \mathcal{L}_{\mathrm{DPO}}}{\partial \Delta_\theta} = -\beta \sigma(-z) < 0. \tag{44}$$

Therefore, gradient descent always increases Δ_θ for a selected positive–negative pair.

More specifically, the score derivatives are

$$\frac{\partial \mathcal{L}_{\mathrm{DPO}}}{\partial s_\theta(q, y^+)} = -\frac{\beta}{\tau_s} \sigma(-z), \tag{45}$$

$$\frac{\partial \mathcal{L}_{\mathrm{DPO}}}{\partial s_\theta(q, y^-)} = \frac{\beta}{\tau_s} \sigma(-z). \tag{46}$$

Thus, treating the two scores locally, gradient descent increases the positive score and decreases the negative score.

At initialization, $\theta = \theta_{\mathrm{ref}}$ and hence $z = 0$. The initial gradients are

$$\left. \frac{\partial \mathcal{L}_{\mathrm{DPO}}}{\partial s_\theta(q, y^+)} \right|_{\theta = \theta_{\mathrm{ref}}} = -\frac{\beta}{2\tau_s}, \tag{47}$$

$$\left. \frac{\partial \mathcal{L}_{\mathrm{DPO}}}{\partial s_\theta(q, y^-)} \right|_{\theta = \theta_{\mathrm{ref}}} = \frac{\beta}{2\tau_s}. \tag{48}$$

Every selected residual inversion therefore receives a nonzero corrective signal at the beginning of Stage 2.

The curvature with respect to the score gap is

$$\frac{\partial^2 \mathcal{L}_{\mathrm{DPO}}}{\partial \Delta_\theta^2} = \beta^2 \sigma(z)\sigma(-z) > 0. \tag{49}$$

The loss is consequently convex as a function of the scalar score gap. Moreover, the gradient magnitude

$$\left| \frac{\partial \mathcal{L}_{\mathrm{DPO}}}{\partial \Delta_\theta} \right| = \beta \sigma(-z) \tag{50}$$

decreases as the student establishes a stronger preference improvement over the reference. Score-level DPO therefore provides a smooth corrective signal while reducing the update magnitude once the reference-relative preference has been sufficiently strengthened.

C.5 Why Use DPO Instead of Alternative Pairwise Objectives?

Several pairwise objectives could be applied to the reliable disagreement set. We next clarify why DPO is particularly compatible with our two-stage design, while not claiming that it universally dominates every alternative.

Comparison with RankNet or BPR. A standard pairwise logistic objective can be written as

$$\mathcal{L}_{\mathrm{RankNet}} = -\log \sigma\left(\beta \Delta_\theta\right). \tag{51}$$

This objective optimizes the student's absolute positive–negative ordering. In contrast, DPO applies the logistic loss to $\Delta_\theta - \Delta_{\mathrm{ref}}$:

$$\mathcal{L}_{\mathrm{DPO}} = -\log \sigma\left[\beta\left(\Delta_\theta - \Delta_{\mathrm{ref}}\right)\right]. \tag{52}$$

DPO therefore expresses Stage 2 as a reference-relative update: the quantity being optimized is the change in pairwise log-odds relative to the Stage 1 model. This is consistent with the role of Stage 2 as residual correction rather than an independent retraining stage.

Comparison with MarginMSE. A teacher-margin regression objective takes the form

$$\mathcal{L}_{\mathrm{MarginMSE}} = \left(\Delta_\theta - \Delta_\phi\right)^2, \tag{53}$$

where Δ_ϕ is the teacher score gap. MarginMSE transfers both the preference direction and the numerical magnitude of the teacher margin. This can be useful when teacher and student score scales are well calibrated, but it also makes the student sensitive to the teacher's score magnitude.

RD-DPO instead uses the teacher only to validate the preference direction and does not require the student to reproduce the teacher's margin. This separation is useful in our setting because the teacher and student can differ substantially in model capacity and score calibration.

Comparison with a hinge-ranking loss. A pairwise hinge objective is

$$\mathcal{L}_{\mathrm{hinge}} = \max\left(0, \mu - \Delta_\theta\right), \tag{54}$$

where μ is a fixed target margin. Its behavior depends on the choice of μ, and it provides no gradient after the target margin is reached. DPO instead uses a smooth logistic objective and defines the update relative to the Stage 1 preference rather than through a globally fixed absolute margin.

Comparison with continued listwise distillation. Continuing the Stage 1 KL objective would further reduce global teacher–student distributional discrepancy. However, it distributes supervision across all candidates in the set and does not isolate the small number of annotation-consistent residual inversions. RD-DPO serves a complementary role: candidate-set KL transfers global listwise structure, while reliable disagreement selection and DPO concentrate updates on remaining local ordering errors.

Summary. DPO is suitable for Stage 2 because it simultaneously provides:

1. a pairwise objective matching the form of the residual error;
2. a frozen Stage 1 reference for measuring preference changes;
3. a smooth, adaptive correction signal; and
4. no requirement to regress the numerical teacher margin.

These properties motivate our choice of DPO. They do not, by themselves, establish that DPO must outperform every pairwise alternative; that question is evaluated empirically using controlled objective comparisons.

C.6 Scope and Limitations of the Analysis

The preceding analysis has several important boundaries.

First, Eq. equation 33 defines a local policy over a selected pair. The resulting derivation establishes equivalence to DPO for this two-candidate retrieval policy, not for a normalized policy over the entire catalog.

Second, the frozen reference model does not guarantee that all unselected candidate scores or the complete Stage 1 candidate distribution remain unchanged. The reference term measures pairwise preference improvement, but it is not a global constraint on parameter distance or full-catalog distributional drift.

Third, reliable disagreement selection depends on the quality of the annotations and the teacher. The teacher-margin threshold reduces ambiguous supervision but cannot guarantee that every retained teacher preference is correct.

Fourth, minimizing the logistic DPO loss encourages an increasingly large reference-relative preference gap. It does not impose a finite target margin or guarantee that every inversion is corrected after a finite number of optimization steps. In practice, the score range, temperature, update strength, and number of training steps control the magnitude of the correction.

Finally, the theoretical analysis motivates pairwise correction and establishes the connection between score-level DPO and standard DPO. It does not prove the empirical superiority of DPO over RankNet, MarginMSE, or hinge ranking. A controlled comparison using the same reliable-disagreement pairs is therefore necessary to isolate the effect of the optimization objective.

D Additional Experimental Results

ToolRet results across cutoffs. Table 4 reports the ToolRet results at multiple retrieval depths. Compared with Tool-Embed$_{\text{Tool-DE}}$-4B, our 4B model gains $+5.26$ in NDCG@1 and $+3.41$ in Recall@1. The corresponding gains at $K = 10$ are $+3.74$ and $+3.06$, respectively. The larger NDCG gain at the first rank is consistent with improved placement of relevant tools near the top of the list.

ToolRet results by tool category. Table 5 separates Web, Code, and Customized tools. Relative to Tool-Embed$_{\text{Tool-DE}}$-4B, our 4B model improves NDCG@10 by $+3.76$, $+3.80$, and $+3.67$ in these categories, respectively. Recall@10 and Comprehensiveness@10 also improve in all three categories, showing that the gain is not confined to one type of tool.

SkillRet results across cutoffs. Table 6 shows gains at every reported cutoff for both student sizes. At $K = 10$, the 0.6B student distilled from the 8B teacher exceeds SkillRet-Embedding-0.6B by $+0.96$ in NDCG, $+1.31$ in Recall, and $+2.09$ in Completeness. Relative to SkillRet-Embedding-4B, our 4B model gains $+1.36$, $+1.46$, and $+2.75$ on the same metrics. The larger Completeness gain indicates that more queries have their full set of relevant skills retrieved. For the 0.6B student, the 8B-teacher run also exceeds the 4B-teacher run by $+0.40$, $+0.41$, and $+0.77$ at $K = 10$.

Table 4: Detailed retrieval results on ToolRet at $K \in \{1, 3, 5, 10\}$. NDCG, Recall, and Comprehensiveness are reported as percentages. Bold and underlined values indicate the best and second-best results in each column, respectively.

	NDCG				Recall				Comprehensiveness			
Model	@1	@3	@5	@10	@1	@3	@5	@10	@1	@3	@5	@10
Qwen3-Embedding-0.6B	36.07	38.05	40.09	43.13	27.57	40.14	45.69	52.80	22.31	32.80	37.61	42.97
Qwen3-Embedding-4B	39.16	40.22	42.45	45.54	29.10	41.97	48.02	57.36	23.04	34.09	38.84	47.27
Qwen3-Embedding-8B	38.95	40.78	43.11	46.21	29.48	42.66	48.72	57.52	23.55	34.62	39.60	47.52
Tool-Embed$_{\text{Tool-DE}}$-0.6B	39.82	41.72	44.28	47.35	29.15	43.45	50.51	59.10	22.89	34.49	40.09	48.14
Tool-Embed$_{\text{Tool-DE}}$-4B	43.53	45.92	48.30	<u>51.51</u>	31.96	<u>48.15</u>	54.74	<u>62.80</u>	25.10	<u>38.73</u>	43.92	<u>51.02</u>
CS-OPD + RD-DPO (0.6B)	<u>44.74</u>	<u>46.33</u>	<u>48.76</u>	51.42	<u>33.03</u>	48.14	<u>54.79</u>	61.93	<u>26.12</u>	38.64	<u>44.25</u>	51.00
CS-OPD + RD-DPO (4B)	**48.79**	**50.03**	**52.44**	**55.25**	**35.37**	**51.13**	**58.05**	**65.86**	**27.48**	**40.27**	**46.31**	**53.96**

Table 5: Detailed ToolRet results by tool category at $K = 10$. N, R, and C denote NDCG, Recall, and Comprehensiveness; all values are percentages. The Teacher column identifies the Qwen3-Embedding model used for distillation. Bold and underlined values indicate the best and second-best results in each column.

		Web			Code			Customized			Avg.		
Model	**Teacher**	N@10	R@10	C@10	N@10	R@10	C@10	N@10	R@10	C@10	N@10	R@10	C@10
BM25s	–	26.18	34.13	22.80	41.90	56.47	55.36	41.16	48.61	38.90	36.41	46.40	39.02
GritLM-7B	–	36.58	46.01	27.65	41.26	53.81	52.07	45.55	54.01	41.40	41.13	51.28	40.37
NV-Embed-v1	–	31.51	40.52	26.74	47.92	62.07	59.60	48.70	57.69	43.88	42.71	53.43	43.41
gte-Qwen2-1.5B-instruct	–	36.73	46.39	28.24	40.56	53.08	51.37	46.52	55.38	42.10	41.27	51.62	40.57
e5-mistral-7b-instruct	–	32.59	42.17	27.51	44.05	57.43	55.09	43.42	50.69	39.16	40.02	50.10	40.59
Qwen3-Embedding-0.6B	–	38.22	46.17	28.47	48.60	62.29	60.06	42.58	49.93	40.38	43.13	52.80	42.97
Qwen3-Embedding-4B	–	40.90	50.14	31.97	53.56	71.12	69.61	42.15	50.80	40.22	45.54	57.36	47.27
Qwen3-Embedding-8B	–	41.61	50.23	32.56	53.43	69.21	67.29	43.59	53.13	42.73	46.21	57.52	47.52
Tool-Embed$_{\text{original}}$-0.6B	–	38.57	48.05	32.39	50.11	62.22	60.52	51.77	60.17	46.28	46.82	56.81	46.40
Tool-Embed$_{\text{original}}$-4B	–	40.87	50.61	34.51	54.63	68.62	67.58	52.12	60.77	46.31	49.21	60.00	49.47
Tool-Embed$_{\text{Tool-DE}}$-0.6B	–	42.35	52.79	34.95	49.37	64.41	62.50	50.34	60.11	46.98	47.35	59.10	48.14
Tool-Embed$_{\text{Tool-DE}}$-4B	–	<u>45.74</u>	<u>55.54</u>	<u>37.20</u>	54.40	69.09	67.43	<u>54.38</u>	<u>63.76</u>	<u>48.42</u>	<u>51.51</u>	<u>62.80</u>	<u>51.02</u>
CS-OPD + RD-DPO (0.6B)	Qwen3-4B	44.04	54.35	36.12	<u>57.43</u>	<u>71.81</u>	<u>69.86</u>	50.04	56.56	44.61	50.50	60.90	50.20
CS-OPD + RD-DPO (0.6B)	Qwen3-8B	45.50	55.21	36.85	57.17	71.40	69.57	51.59	59.19	46.59	51.42	61.93	51.00
CS-OPD + RD-DPO (4B)	Qwen3-8B	**49.50**	**58.44**	**40.02**	**58.20**	**73.06**	**71.41**	**58.05**	**66.06**	**50.45**	**55.25**	**65.86**	**53.96**

Table 6: Detailed retrieval results on SkillRet at $K \in \{5, 10, 15\}$. NDCG, Recall, and Completeness are reported as percentages. The Teacher column identifies the Qwen3-Embedding model used for distillation; "–" indicates no teacher. Bold and underlined values indicate the best and second-best results in each column.

		NDCG			Recall			Completeness		
Model	**Teacher**	@5	@10	@15	@5	@10	@15	@5	@10	@15
BM25	–	49.31	51.69	52.75	53.03	59.41	62.92	38.21	44.56	47.93
e5-small-v2	–	42.63	44.66	45.66	46.03	51.59	54.89	32.61	37.45	40.35
e5-large-v2	–	51.30	53.41	54.44	54.88	60.55	63.89	40.16	45.45	48.68
bge-small-en-v1.5	–	52.57	54.51	55.45	54.73	60.01	63.07	38.96	43.97	47.11
snowflake-arctic-embed-s	–	54.48	56.39	57.35	56.95	61.96	65.02	41.37	46.22	49.32
bge-large-en-v1.5	–	57.04	59.00	59.80	59.19	64.37	66.95	42.96	48.34	50.80
F2LLM-v2-80M	–	46.26	48.22	49.27	49.51	54.86	58.32	34.77	39.82	43.06
pplx-embed-v1-0.6b	–	51.71	54.26	55.52	56.72	63.60	67.70	41.51	48.27	52.32
NV-Embed-v1	–	54.62	57.03	57.96	58.16	64.64	67.64	42.12	48.00	50.87
KaLM-Gemma3-12B	–	56.31	58.95	60.11	60.48	67.56	71.35	44.49	52.37	56.42
Qwen3-Embedding-0.6B	–	59.92	61.94	62.98	62.59	67.87	71.25	45.88	51.09	54.67
jina-embeddings-v5-text-small	–	60.85	62.96	63.90	63.46	68.99	72.02	46.93	53.26	56.74
Qwen3-Embedding-8B	–	61.35	63.64	64.73	64.24	70.29	73.76	47.43	54.14	58.15
harrier-oss-v1-270m	–	62.42	64.56	65.33	65.12	70.80	73.27	49.00	55.37	58.06
Octen-Embedding-8B	–	64.00	65.93	66.85	65.98	71.07	73.99	49.32	55.35	58.99
harrier-oss-v1-0.6b	–	68.16	70.26	71.09	70.64	76.16	78.76	54.80	61.61	64.82
SkillRouter-Embedding-0.6B	–	71.80	73.54	74.33	73.73	78.18	80.68	57.22	62.70	65.96
SkillRet-Embedding-0.6B	–	79.02	81.12	81.87	82.38	87.74	90.16	70.83	78.94	82.65
SkillRet-Embedding-4B	–	81.93	83.90	84.56	85.62	90.65	92.73	75.57	83.36	86.95
CS-OPD + RD-DPO (0.6B)	Qwen3-4B	79.49	81.68	82.40	83.04	88.64	90.90	71.86	80.26	83.99
CS-OPD + RD-DPO (0.6B)	Qwen3-8B	80.00	82.08	82.85	83.71	89.05	91.52	73.09	81.03	84.97
CS-OPD + RD-DPO (4B)	Qwen3-8B	**83.38**	**85.26**	**85.83**	**87.34**	**92.11**	**93.93**	**78.21**	**86.11**	**89.14**