RLRank: Distilling Offline Oracles into Online Policies for Document Reranking

Anonymous Authors[1]

Abstract

We investigate whether small language models (SLMs) can learn effective document reranking through reinforcement learning, using a state-of-the-art cross-encoder as an offline oracle reward signal, without any supervised labels. We compare two action space formulations under Group Relative Policy Optimization (GRPO): listwise reranking, where the model outputs a full permutation of document indices, and pairwise reranking, where the model makes binary document preference judgments. We identify a zero-gradient deadlock that occurs when all rollouts within a GRPO group receive identical rewards, and propose a partial reward shaping strategy that resolves it. After this fix, a 1B-parameter listwise model reaches NDCG@10 = 0.854 after only 515 training steps (≈3% of a full run), while a 3B pairwise model achieves 76.1% pairwise accuracy after 396 steps. Listwise training is substantially more sample-efficient, requiring one LLM call per query vs. $O(K^2)$ for pairwise.

1. Introduction

Document reranking is a critical component of modern retrieval-augmented generation (RAG) pipelines. State-of-the-art rerankers are cross-encoders that jointly encode a query and document to output a relevance score, but require a separate forward pass per query-document pair, making them expensive to operate and inflexible to update.

This work fits squarely in the **Offline RL for Foundation Models** theme of the workshop. The cross-encoder oracle is invoked once, offline, to score a static dataset of query-document pairs. A small language model is then trained via GRPO entirely from these offline rewards, with no further oracle queries at training time. At inference, the trained SLM serves ranking decisions with a single forward pass, independent of the oracle entirely. This offline-to-online separation is the core design principle: expensive oracle evaluation happens once; cheap SLM inference happens at scale.

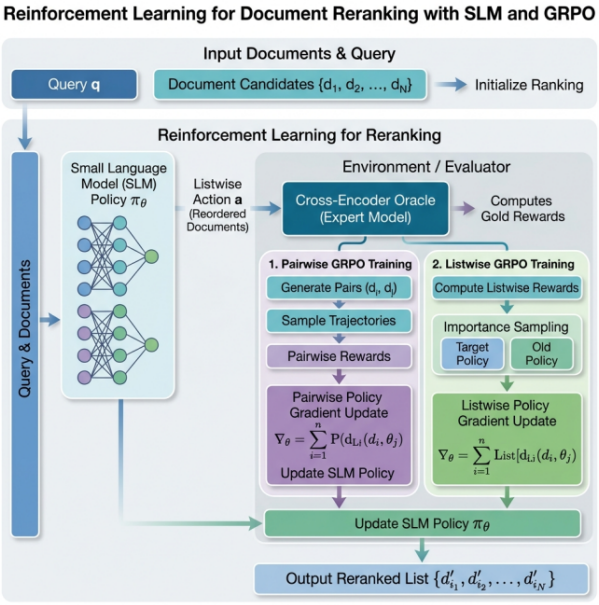

Figure 1. RLRank architecture: an offline cross-encoder oracle constructs a reward dataset once; the SLM policy is then trained via GRPO and deployed online without any oracle dependency.

We make three contributions:

1. We identify the **zero-gradient deadlock**, a failure mode where GRPO stalls because all rollouts in a group receive zero reward, yielding zero advantage and zero gradient.

2. We propose **partial reward shaping** to break the deadlock by creating within-group reward variance before the model has learned the target output format.

3. We show that a 1B listwise model trained entirely offline achieves NDCG@10 = 0.854 in just 515 steps,

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

demonstrating that offline RL can produce deployment-ready online policies with minimal compute.

2. Background

2.1. Document Reranking

Given a query q and K candidates $d_1, \ldots, d_K$, the task is to produce a permutation σ ordered by relevance. Quality is measured by NDCG@K:

$$\text{NDCG@K} = \frac{\text{DCG@K}}{\text{IDCG@K}}, \quad \text{DCG@K} = \sum_{i=1}^{K} \frac{\text{rel}(d_{\sigma(i)})}{\log_2(i+1)}. \tag{1}$$

2.2. GRPO

Group Relative Policy Optimization (Shao et al., 2024) samples G completions per input and normalizes rewards within the group to compute advantages:

$$\hat{A}_i = \frac{r_i - \mu_G}{\sigma_G}, \qquad \mu_G = \frac{1}{G}\sum_j r_j, \quad \sigma_G = \sqrt{\frac{1}{G}\sum_j (r_j - \mu_G)^2}. \tag{2}$$

If all rewards in a group are identical, $\sigma_G = 0$, all advantages vanish, and the gradient is exactly zero, regardless of the absolute reward value.

3. Method

3.1. Offline Data Construction

We build a static reward dataset from source PDFs without human labels: (1) chunks of 1000 characters are embedded and indexed in FAISS; (2) `GPT-4o-mini` generates 3 queries per chunk; (3) top-10 chunks per query are retrieved and scored by `BAAI/bge-reranker-v2-m3` to produce reference rankings. This yields 1,529 items. The oracle is invoked *only during dataset construction*, never at training time, embodying the offline phase.

3.2. RL Action Spaces

Listwise. The model receives all K documents and outputs a comma-separated permutation of indices. Reward = NDCG@10 vs. the oracle ranking; one forward pass per query at inference.

Pairwise. Each query-document set yields $\binom{K}{2} = 45$ binary comparisons. The model outputs `A` or `B`; reward = 1.0 (correct) or 0.0. Preferences are aggregated via win-count sort; 45 forward passes required per query.

3.3. Partial Reward Shaping

Both models suffered a zero-gradient deadlock early in training: the listwise model produced off-topic text (not digit sequences) and the pairwise model produced verbose explanations (not a single letter). With all group rewards equal to zero, GRPO received zero gradient. We break this with partial reward shaping:

Table 1. Partial reward shaping schedules.

Output type	Reward
Listwise	
Invalid, no digits	0.00
Invalid, contains digits	0.05
Valid permutation	NDCG@10 $\in [0, 1]$
Pairwise	
No A/B in output	0.00
Verbose + correct	0.50
Verbose + wrong	0.00
Clean A/B + correct	1.00
Clean A/B + wrong	0.00

Any nonzero within-group reward variance gives GRPO a gradient to work with. The 2× gap between verbose-correct (0.5) and clean-correct (1.0) also provides a curriculum incentive toward output conciseness.

3.4. Training Setup

Table 2. Hyperparameters for both formulations.

Hyperparameter	Listwise	Pairwise
Base model	Llama-3.2-1B	Llama-3.2-3B
Algorithm	GRPO	GRPO
Group size	8	8
Learning rate	2e-5	2e-5
Max new tokens	64	16
KL coeff.	0.01	0.0
Epochs	3	3
Data aug.	3×	none
Train / eval	1,479 / 50	1,479 / 50

Both models use LoRA fine-tuning on remote GPUs.

4. Results

4.1. Zero-Gradient Deadlock and Recovery

Both models stalled completely before reward shaping. The listwise (1B) model showed no learning signal after 250 steps: entropy *increased* from 3.16 to 3.45, confirming random drift. The pairwise (3B) model converged on verbose

responses, collapsing entropy from 3.5 to 1.0 while achieving only 48–56% accuracy (near random).

Partial reward shaping broke the deadlock immediately. Within 14 steps, `frac_all_bad` dropped from 1.0 to 0.0, `frac_mixed` hit 1.0, and entropy reversed direction (3.45 → 2.77).

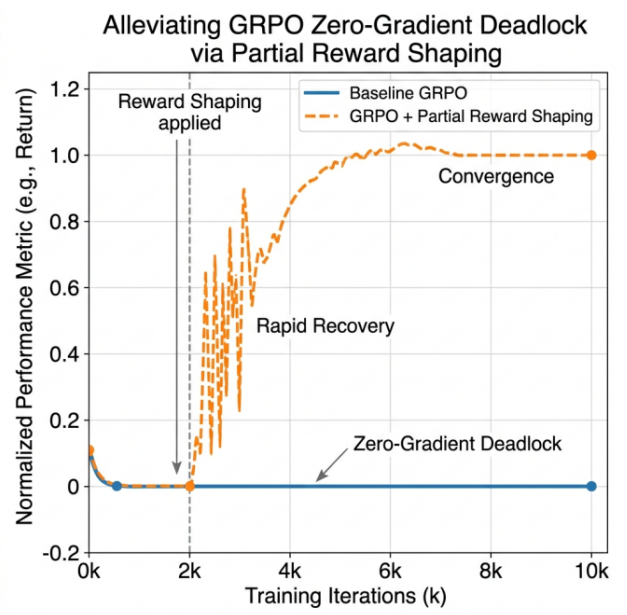

Figure 2. Partial reward shaping breaks the zero-gradient deadlock and restores policy plasticity within 14 steps.

4.2. Listwise Training Efficiency

After shaping, the listwise model's learning proceeds in three phases:

Table 3. Listwise training phases.

Phase	Steps	NDCG@10	Valid%
Deadlocked	0–260	0.0	0
Format acq.	260–380	0.0–0.2	var.
Rapid impr.	380–515	**0.8–0.9**	**100**

NDCG@10 = 0.854 at step 515, representing only ≈2.9% of a full 3-epoch run. The phase-transition shape (zero progress then rapid emergence) indicates strong *latent* relevance understanding from pretraining: the offline RL signal teaches the model how to express a ranking rather than what relevance means.

4.3. Listwise vs. Pairwise

The pairwise model requires $O(K^2)$ calls per query vs. $O(1)$ for listwise. This gap is structural, not an artifact of training length.

Table 4. Comparison at step ≈400.

Dimension	Listwise (1B)	Pairwise (3B)
Metric	NDCG = **0.854**	Acc = **76.1%**
Steps	≈380	Ongoing at 396
Calls/query	**1**	45
Format learned	Yes	No (verbose)

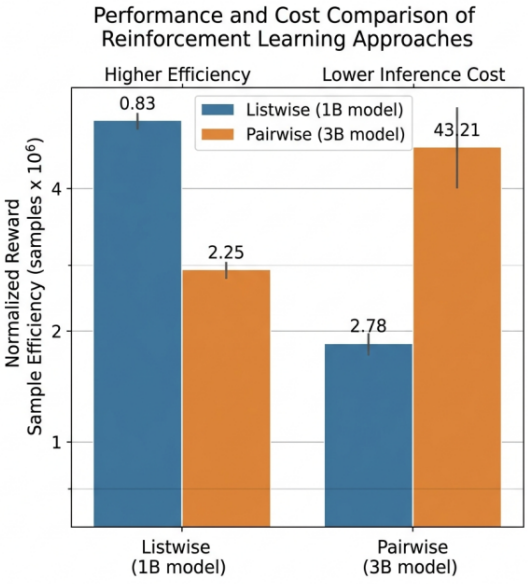

Figure 3. Efficiency and cost comparison between listwise and pairwise models.

5. Discussion

Offline-to-online transfer. Our methodology provides a clean offline-to-online separation. The offline phase: the cross-encoder oracle scores 1,529 query-document sets once, at full quality. The oracle is too slow for real-time use (BGE-Reranker-v2-m3 requires $K = 10$ forward passes per query), but that cost is amortized across all future training steps. The online phase: the trained 1B SLM serves ranking decisions with a single forward pass, operating independently of the oracle at interactive latencies. This separation is the key deployment property that offline RL enables: oracle-quality reward signals need not imply oracle-level inference cost.

The cold-start problem in GRPO. The zero-gradient deadlock is a general failure mode for GRPO on structured output tasks: when the base model has no format prior, initial within-group reward variance is zero and training cannot bootstrap. Partial reward shaping resolves this by making

the reward landscape navigable before format is acquired. The key insight: any nonzero reward gradient is infinitely more useful than no gradient.

Latent capability and offline RL. The phase-transition learning curve suggests the SLM already possesses substantial relevance understanding from pretraining. The offline reward signal unlocks this latent capability by teaching the model to express rankings in a structured format, not by teaching it what relevance means. This has an important implication for offline RL more broadly: the offline dataset need not contain all the knowledge required for the task; it only needs to provide sufficient reward signal to surface what the model already knows.

6. Related Work

Cross-encoder rerankers. Chen et al. (2024) and ms-marco-MiniLM (Reimers & Gurevych, 2019) are strong baselines but require per-document forward passes and serve as the offline oracle in our framework.

LLM-based reranking. RankGPT (Sun et al., 2023) and PairRank (Qin et al., 2023) use large models zero-shot. We instead train small models via offline RL, enabling deployment-grade inference cost.

Offline RL for foundation models. GRPO (Shao et al., 2024) was originally applied to mathematical reasoning. DeepSeek-R1 (DeepSeek-AI, 2025) demonstrated GRPO's effectiveness at bootstrapping reasoning from offline reward signals. Our work characterizes the zero-gradient deadlock as a general failure mode and provides a fix applicable across structured-output offline RL tasks.

Reward shaping. Potential-based shaping (Ng et al., 1999) provides policy-invariance guarantees for reward augmentation. Our partial shaping smooths the early offline reward landscape to enable gradient flow before format is acquired.

7. Conclusion

We presented RLRank, showing that small language models can be trained offline via GRPO to produce deployment-ready online reranking policies. The offline cross-encoder oracle is queried once to build a reward dataset; the resulting 1B SLM policy serves queries with a single forward pass at inference. By identifying the zero-gradient deadlock and resolving it with partial reward shaping, we achieved NDCG@10 = 0.854 in 515 steps (3% of a full run). Future work will evaluate on standard benchmarks (BEIR, MS-MARCO) and explore targeted negative rewards for deliberate document forgetting.

References

Chen, J., Xiao, S., Zhang, P., Luo, K., Lian, D., and Liu, Z. BGE M3-Embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *arXiv preprint arXiv:2402.03216*, 2024.

DeepSeek-AI. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. Technical report, DeepSeek, 2025.

Ng, A. Y., Harada, D., and Russell, S. Policy invariance under reward transformations: Theory and application to reward shaping. In *Proceedings of the 16th International Conference on Machine Learning (ICML 1999)*, pp. 278–287. Morgan Kaufmann, 1999.

Qin, Z., Jagerman, R., Hui, K., Zhuang, H., Wu, J., Shen, L., Liu, T., Liu, J., Metzler, D., Wang, X., and Bendersky, M. Large language models are effective text rankers with pairwise ranking prompting. *arXiv preprint arXiv:2306.17563*, 2023.

Reimers, N. and Gurevych, I. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP 2019)*, pp. 3982–3992. Association for Computational Linguistics, 2019.

Shao, Z., Wang, P., Zhu, Q., Guo, R., Yang, J., Wu, Y., Liu, C., Du, K., and Gao, M. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Sun, W., Yan, L., Ma, X., Wang, S., Ren, P., Chen, Z., Yin, J., and Ren, C. Is ChatGPT good at search? investigating large language models as re-ranking agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP 2023)*, pp. 14918–14937. Association for Computational Linguistics, 2023.