

Rank-R1: Enhancing Reasoning in LLM-based Document Rerankers via Reinforcement Learning

Shengyao Zhuang*
The University of Queensland
Brisbane, QLD, Australia

Xueguang Ma*
University of Waterloo
Waterloo, Canada

Zheng Yao
The University of Queensland
Brisbane, QLD, Australia

Shuai Wang
The University of Queensland
Brisbane, QLD, Australia

Bevan Koopman
CSIRO and
The University of Queensland
Brisbane, QLD, Australia

Jimmy Lin
University of Waterloo
Waterloo, Canada

Guido Zuccon
The University of Queensland
Brisbane, QLD, Australia

Abstract

We introduce Rank-R1, an LLM-based reranker that reasons over queries and candidate documents before ranking. Existing rerankers based on LLMs rely on prompting or fine-tuning to order documents by relevance without explicit reasoning. Rank-R1 uses reinforcement learning with only relevance labels (no reasoning supervision) to learn an intermediate reasoning step that improves ranking. Experiments show that Rank-R1 is competitive with supervised fine-tuning on in-domain TREC DL queries. On the out-of-domain BRIGHT benchmark, which requires complex reasoning, Rank-R1 outperforms both zero-shot and supervised baselines; the 14B model surpasses RankGPT4. With a scaled configuration combining a 32B backbone, improved first-stage retrieval, and harder training data, Rank-R1 achieves state-of-the-art performance on BRIGHT.

CCS Concepts

• **Information systems** → **Language models; Information retrieval; Retrieval models and ranking.**

Keywords

LLMs as Rankers, Reasoning Rankers, Reinforcement Learning

ACM Reference Format:

Shengyao Zhuang, Xueguang Ma, Zheng Yao, Shuai Wang, Bevan Koopman, Jimmy Lin, and Guido Zuccon. 2026. Rank-R1: Enhancing Reasoning in LLM-based Document Rerankers via Reinforcement Learning. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '26)*, July 20–24, 2026, Melbourne, VIC, Australia. ACM, New York, NY, USA, 7 pages. <https://doi.org/10.1145/3805712.3809961>

*Equal Contribution



This work is licensed under a Creative Commons Attribution 4.0 International License. *SIGIR '26, Melbourne, VIC, Australia*

© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2599-9/2026/07
<https://doi.org/10.1145/3805712.3809961>

1 Introduction

Large language models (LLMs) have shown strong performance in document ranking tasks [1, 2, 18, 33, 37, 41–44]. Zero-shot prompting methods such as RankGPT rerank documents by directly prompting LLMs to generate reordered document lists [33]. However, these methods rely on the assumption that the LLM follows instructions well. Moreover, being zero-shot, they do not leverage available human-annotated relevance data for further improvement.

In contrast, methods like RankLlama fine-tune LLMs using human relevance judgments, assigning scores to each query-document pair [17]. While effective, these approaches do not explicitly model reasoning processes, mainly due to the lack of high-quality reasoning data for supervised fine-tuning. In practice, user relevance judgments often come in the form of selecting the most relevant document from a set, but achieving high-quality rankings requires strong reasoning to interpret complex relevance relationships.

Recent advancements in reinforcement learning (RL) for LLMs, such as DeepSeek-R1 [8] and Simple RL [39], have demonstrated that reward-based training can enhance reasoning abilities, particularly in mathematical question-answering [29]. These results suggest that verifiable reward scoring alone can improve an LLM's ability to reason.

Building on this insight, we ask whether reinforcement learning in the style of DeepSeek-R1 can improve reasoning in document reranking. Specifically, we apply Group Relative Policy Optimization (GRPO) [29] to train an LLM-based reranker. Given a user query and a list of retrieved candidate documents, the reranker generates reasoning steps before selecting the most relevant document. The reward signal is whether the model correctly identifies the most relevant document among the candidates.

We train our model, Rank-R1, on the MS MARCO passage ranking dataset and evaluate it on TREC DL19 and DL20 (in-domain datasets). Our results show that RL-based training is competitive on in-domain web search queries. Additionally, we evaluate Rank-R1 on the BRIGHT dataset [31], which requires complex query reasoning and relevance understanding, and is out-of-domain with respect to the training data. Rank-R1 outperforms both zero-shot prompting and supervised fine-tuning on this out-of-domain dataset. Notably,

our 14B model surpasses zero-shot GPT4 in reranking performance on BRIGHT.

We further examine whether Rank-R1 can scale to state-of-the-art performance when equipped with stronger backbone models and more challenging training data. In Section 5.3, we present an enhanced configuration using larger reasoning-capable LLMs, harder synthetic training data, and improved first-stage retrieval. This work, which was initially conducted in early 2025, is, to our knowledge, the first LLM-based reasoning reranker trained with RL using verifiable rewards.

2 Related Work

Four broad families of zero-shot LLM rankers have emerged in the literature [44]: pointwise [9, 14, 27, 42], pairwise [25], listwise [3, 4, 18, 23, 33], and setwise [13, 22, 43]. These families differ in the ranking mechanisms encoded within the prompt instructions [32]. In the pointwise formulation, the LLM evaluates the relevance of an individual document; in the pairwise formulation, it judges which of two documents is more relevant; listwise methods consider an entire ranked list of documents at once; and setwise methods assess groups of documents in relation to one another.

Earlier supervised neural rerankers, such as monoT5 [20], fine-tune pretrained T5 models to score query–document relevance, but do not produce explicit reasoning. Recent work has introduced reasoning into reranking. JudgeRank [19] uses a multi-step prompting strategy to analyze query and document content before reranking, but as a zero-shot method, it relies on the LLM’s inherent instruction-following ability. Rank1 [36], ReasonRank [16], and Rank-K [38] take a different approach, applying knowledge distillation from DeepSeek-R1 to train reasoning-based rerankers. Outside of reranking, Search-R1 [12] and R1-Searcher [30] use reinforcement learning to improve retrieval in RAG pipelines.

Our work differs from these approaches. We extend the setwise paradigm [43] by training reasoning ability directly through reinforcement learning with verifiable rewards, rather than distillation or prompting. This allows the model to learn to reason over candidate sets without requiring reasoning annotations.

3 Method

3.1 LLM Reranking

To efficiently rank a large set of candidate documents, Setwise adopts a heap-based sorting strategy. Given a list of retrieved documents, the method organizes them into a binary heap structure. Each parent-child comparison in the heap is implemented by prompting the LLM with a small set of candidate documents (e.g., two or more) and asking it to select the most relevant one to the query. The selected document is promoted within the heap. By repeatedly applying these “heapify” operations, the algorithm constructs a max-heap in which the root node corresponds to the most relevant document. The top-ranked documents are then obtained by iteratively extracting the root and re-heapifying the remaining elements.

However, the original Setwise ranking approach does not encourage the LLM to reason about the relevance between the query and the documents. Instead, it directly asks for the most relevant one. To

SYSTEM:

A conversation between User and Assistant. The user asks a question, and the Assistant solves it. The assistant first thinks about the reasoning process in the mind and then provides the user with the answer. The reasoning process and answer are enclosed within <think> </think> and <answer> </answer> tags, respectively, i.e., <think> reasoning process here </think> <answer> answer here </answer>.

USER:

Given the query: "{query}", which of the following documents is most relevant?

[1] {document1}

[2] {document2}

....

[20] {document20}

After completing the reasoning process, please provide only the label of the most relevant document to the query, enclosed in square brackets, within the answer tags. For example, if the third document is the most relevant, the answer should be: <think> reasoning process here </think> <answer>[3]</answer>.

Figure 1: Prompt used for Rank-R1.

unlock the reranker’s reasoning ability, we modify the original Setwise prompt by adding a reasoning instruction, as shown in Figure 1. Specifically, we adapt the system prompt from the DeepSeek-R1-Zero method to the Setwise prompt. This modification encourages LLMs to reason first before providing an answer—in our case, predicting the label of the most relevant candidate document to the query. We refer to the Setwise method using this modified prompt as *Rank-R1*.

3.2 Reinforcement Learning for Reranking

Although any modern instruction-tuned LLM with the Rank-R1 prompt can exhibit strong zero-shot reasoning, its reasoning process may still be suboptimal for the Setwise ranking approach because the model is not fine-tuned on ranking-oriented data. Even if one wanted to address this through supervised fine-tuning, collecting large-scale human-annotated reasoning traces would be costly and impractical.

To address this, we adopt the GRPO algorithm [29]. We choose the Setwise ranking approach as the backbone for two main reasons. First, Setwise prompting enables a simple and verifiable reward design, since the task reduces to predicting the single most relevant document label, which can be directly matched against the ground truth; other LLM-based ranking paradigms do not offer this benefit. Second, it allows a fair comparison between the RL-trained Rank-R1 model and a supervised fine-tuned Setwise reranker trained on the same data but without explicit reasoning. In contrast, other LLM-based ranking approaches such as Listwise prompting are

harder to evaluate fairly, because MSMARCO typically provides only one judged relevant document per query and does not supply a complete ground-truth ranking.

Training Data. The Setwise prompts used for GRPO training are constructed from MSMARCO training data¹ provided by the Tevatron IR toolkit [7]. This dataset contains training queries, human-labeled relevant documents, and BM25-retrieved top-100 documents. For each training query, we sample 19 documents from the BM25-retrieved set and combine them with one labeled relevant document to form a Rank-R1 prompt.

Reward Function. We design a simple and verifiable reward function. A reward of one is assigned if and only if the model’s output satisfies two conditions: (1) it follows the required format (i.e., correctly generates the `<think>` `</think>` and `<answer>` `</answer>` spans), and (2) the predicted document label matches the ground-truth relevant document. Otherwise, the reward is zero. This reward design does not constrain the internal reasoning process itself but encourages the model to produce correctly formatted reasoning that leads to the correct answer. Importantly, this RL procedure does not require any human-annotated reasoning data, which is difficult and costly to collect.

GRPO Training. GRPO encourages the model to generate outputs that achieve higher rewards while preventing the updated model from drifting too far from a reference policy (in our case, the original instruction-tuned LLM). During training, for each Setwise ranking prompt, we sample multiple reasoning outputs from the current model. These sampled outputs are then used to update the model based on their relative quality. A clipping mechanism is applied to stabilize training, and a KL regularization term penalizes excessive deviation from the reference model. We compute rewards based on the correctness of the predicted label and normalize them within each sampled group so that the model is encouraged to prefer better-than-average outputs while discouraging worse ones. Each batch of sampled outputs is used for a single policy update.

4 Experimental Settings

Datasets. We evaluate Rank-R1 in both in-domain and out-of-domain settings. For in-domain evaluation, we use TREC DL19 and DL20 [5, 6], which are based on MSMARCO, which is the same dataset used for training but contain deeper relevance assessments. For out-of-domain evaluation, we use the BRIGHT benchmark [31], which covers diverse domains such as biology, code, and math and requires complex reasoning.

Initial Retrieval. For all methods, initial retrieval is performed using BM25 implemented in *pyserini* [15]. Rerankers are applied to the top 100 retrieved documents.

Rank-R1 Settings. We consider two variants: (i) **Zero-shot**, which applies the Setwise reasoning prompt without additional training, and (ii) **GRPO**, which further optimizes the same prompt using GRPO reinforcement learning on MSMARCO training data. Due to computational constraints, RL training is conducted on approximately 18% of the MSMARCO training set; at this step the model was found to achieve a fair convergence on training validation data.

¹<https://huggingface.co/datasets/Tevatron/msmarco-passage>

Table 1: TREC DL19 and DL20 nDCG@10 results. SFT-supervised fine-tuned. GRPO trained on only 18% of 400k data used to train SFT. * denotes statistically significant improvement over the corresponding zeroshot version ($p < 0.05$).

Model	Training	DL19	DL20
BM25	Zeroshot	.506	.480
RankZephyr-7B	GPT4-distil.	.739	.706
RankGPT	Zeroshot	.756	.706
Setwise-3B	Zeroshot	.453	.342
Setwise-3B	SFT	.738*	.684*
Rank-R1-3B	Zeroshot	.570	.546
Rank-R1-3B	GRPO	.693*	.647*
Setwise-7B	Zeroshot	.683	.630
Setwise-7B	SFT	.733*	.682*
Rank-R1-7B	Zeroshot	.694	.654
Rank-R1-7B	GRPO	.739*	.691*
Setwise-14B	Zeroshot	.698	.657
Setwise-14B	SFT	.735*	.704*
Rank-R1-14B	Zeroshot	.678	.658
Rank-R1-14B	GRPO	.719*	.692*

Baselines. To isolate the impact of reasoning and RL training, we compare against the original Setwise method in both zero-shot and supervised fine-tuning (SFT) settings. We also report results from two strong listwise rerankers: RankZephyr [24] and RankGPT [31, 33]. For RankGPT on BRIGHT, note that results are based on a different BM25 implementation.

Backbones and Training. Rank-R1 and Setwise use instruction-tuned Qwen2.5 models ranging from 3B to 14B parameters [26]. RankZephyr uses Zephyr-7B [34], and RankGPT uses GPT4 [21]. For training, we use the TRL library [35] with LoRA adapters [10]. For SFT, the loss is computed only on the answer span.

5 Results

5.1 In-domain effectiveness

In Table 1, we present the effectiveness of Rank-R1 variants compared to the Setwise reranker on the TREC-DL19 and DL20 passage ranking datasets.

We start by comparing Setwise and Rank-R1 under the zero-shot setting. Rank-R1 achieves higher nDCG@10 with the 3B and 7B backbone models and demonstrates comparable effectiveness with the 14B model. These results suggest that incorporating a reasoning process into the Setwise method improves zero-shot ranking effectiveness. The improvements are particularly substantial for the 3B model.

Next, we examine the effects of training with GRPO. With GRPO training, the effectiveness of Rank-R1 increases, indicating that reasoning and answer generation (i.e., ranking) are enhanced through RL training. However, Rank-R1 trained with GRPO appears to be less effective than Setwise trained with SFT (except for the 7B backbone), which is directly optimized to predict document relevance.

Table 2: BRIGHT nDCG@10 results. For the first block (up to Rank-R1-14B), all methods rerank our BM25 baseline (first line). For the second block, the last three rows (Rank1-32B, Rank-K-32B, Rank-R1-32B) rerank the BM25+GPT4 CoT first-stage retrieval. *: Results directly copied from the paper, which uses a different BM25 ranking system (with a higher average nDCG@10 than our BM25).

Model	Training	Bio.	Earth.	Econ.	Psy.	Rob.	Stack.	Sus.	Pony	Leet.	AoPS	TheoT.	TheoQ.	Avg.
BM25	Zeroshot	.182	.279	.164	.134	.109	.163	.161	.043	.247	.065	.021	.073	.137
RankZephyr-7b	GPT4-distill	.219	.237	.144	.103	.076	.137	.166	.065	.247	.068	.020	.073	.130
RankGPT4*	Zeroshot	.338	.342	.167	.270	.223	.277	.111	.156	.034	.012	.086	.002	.170
Setwise-3B	Zeroshot	.143	.175	.120	.102	.077	.079	.154	.053	.154	.017	.042	.021	.095
Setwise-3B	SFT	.220	.188	.104	.115	.091	.058	.167	.057	.099	.040	.034	.038	.101
Rank-R1-3B	Zeroshot	.137	.173	.119	.152	.100	.066	.178	.037	.077	.040	.060	.025	.097
Rank-R1-3B	GRPO	.184	.171	.137	.169	.090	.100	.165	.047	.111	.035	.059	.032	.108
Setwise-7B	Zeroshot	.236	.223	.161	.171	.149	.092	.183	.063	.149	.041	.104	.056	.136
Setwise-7B	SFT	.287	.301	.141	.239	.189	.137	.196	.071	.207	.070	.082	.082	.167
Rank-R1-7B	Zeroshot	.268	.248	.179	.221	.174	.103	.211	.044	.156	.033	.104	.059	.150
Rank-R1-7B	GRPO	.260	.285	.172	.242	.191	.104	.242	.043	.198	.043	.109	.083	.164
Setwise-14B	Zeroshot	.295	.322	.205	.248	.189	.147	.236	.087	.187	.080	.093	.076	.180
Setwise-14B	SFT	.220	.293	.154	.230	.201	.157	.203	.062	.194	.095	.099	.097	.167
Rank-R1-14B	Zeroshot	.301	.366	.221	.246	.217	.154	.250	.090	.170	.091	.116	.092	.193
Rank-R1-14B	GRPO	.312	.385	.212	.264	.226	.189	.275	.092	.202	.097	.119	.092	.205
BM25+GPT4 CoT	Zeroshot	.536	.541	.243	.387	.189	.277	.263	.176	.193	.039	.208	.192	.270
Rank1-32B	Distill	.497	.358	.220	.375	.225	.217	.350	.325	.188	.108	.437	.229	.294
Rank-K-32B	Distill	.504	.462	.306	.467	.324	.330	.412	.322	.240	.076	.266	.283	.333
Rank-R1-32B	GRPO	.623	.593	.341	.507	.324	.389	.463	.181	.266	.106	.412	.311	.376

We hypothesize that this is because TREC DL queries are simple, straightforward web search queries that do not require extensive reasoning to produce correct predictions. In the next section, we explore performance in more challenging query settings.

5.2 Reasoning-intensive out-of-domain effectiveness

Next we consider results from our out-of-domain and reasoning-intensive setting, reported in Table 2.

We observe that the SOTA RankZephyr reranker, which does not incorporate reasoning, does not provide better rankings than BM25 in most datasets from the BRIGHT benchmark – effectively failing at the reranking task. This suggests that the BRIGHT benchmark poses a challenge for current SOTA LLM rerankers.

On the other hand, Rank-R1 trained with GRPO outperforms or is on par with both zero-shot and Setwise SFT models in most cases. Notably, when using the 14B model, Setwise SFT effectiveness plateaued and even performed worse than its zero-shot counterparts, suggesting that the large model trained on the MSMARCO could not generalize to BRIGHT using the standard Setwise approach. However, Rank-R1 based on the 14B model achieves the largest performance gain over Setwise SFT and even surpasses the GPT4-based Listwise reranker baseline (RankGPT4). These results highlight that the reasoning process can help the model generalize to different domains and that strong reasoning abilities, along with larger model sizes, are crucial for LLM-based rerankers to be effective in reasoning-intensive ranking tasks.

5.3 Improving Rank-R1 with Advanced Training Strategies

In the previous experiments, we evaluated Rank-R1 under controlled settings to better understand the role of reasoning and reinforcement learning. In this section, we examine whether Rank-R1 can achieve state-of-the-art performance when scaled with stronger backbone models and enhanced training data.

Compared to the earlier setup, we make two key upgrades. First, we replace the Qwen2.5 backbone with Qwen3-32B, which provides stronger reasoning capacity. Second, we switch the training data to the ReasonIR dataset [28], which contains synthetic hard queries, positive documents, and synthetic hard negatives designed to promote reasoning-intensive ranking. This training setup is substantially more challenging than the MSMARCO-based configuration used in earlier controlled experiments.

Table 2 (last block) reports the results. We compare Rank-R1-32B against strong baselines, including Rank1-32B and Rank-K-32B. All the methods rerank candidates produced by BM25+GPT4 CoT, ensuring a fair comparison under the same first-stage retrieval.

Rank-R1-32B achieves the best overall performance among all compared methods, outperforming the distilled reasoning-based reranker baselines (Rank1-32B and Rank-K-32B). These results demonstrate that integrating a stronger reasoning-capable LLM backbone with more challenging training data can significantly improve the performance of Rank-R1.

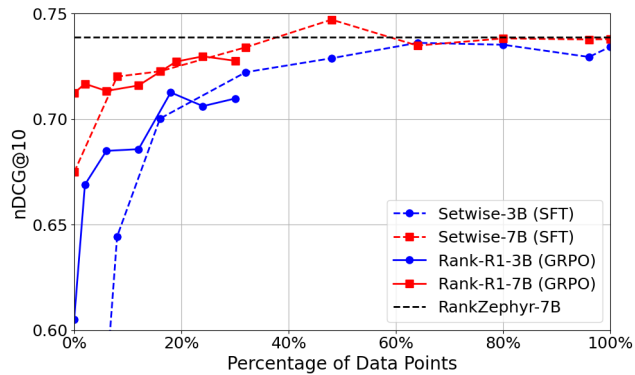


Figure 2: Data efficiency comparison between Setwise SFT and Rank-R1.

6 Analysis

6.1 Effect of quantity of training data

The Rank-R1 GRPO results in Table 1 are obtained using only 18% of the MSMARCO training data, while SFT used the full set. To examine whether longer training would further improve effectiveness, we continued training the 3B and 7B Rank-R1 models for an additional two days and evaluated checkpoints saved during training. Figure 2 reports the results, together with Setwise SFT trained on incremental fractions of the data. Rank-R1 requires substantially less data than Setwise SFT to reach a similar level of effectiveness in the early training stage; however, this data-efficiency advantage diminishes once both methods are trained on more than roughly 5–7% of the data, after which the two curves track each other closely. SFT remains less computationally expensive, while GRPO additionally equips the reranker with an explicit reasoning capability.

6.2 Case study

In Figure 3, we provide an example of Rank-R1’s generation. We compare the outputs of the Zeroshot model and the model after GRPO training. Both models successfully follow the instruction by providing a reasoning process within the `<think>` span and predicting a relevant document label in the correct format. However, the Zeroshot model tends to merely describe what each document mentions and ultimately makes an incorrect prediction. In contrast, the GRPO-trained model focuses on the most relevant documents, compares them, and correctly selects the best one. In addition, we argue that Rank-R1’s transparent reasoning process makes its predictions more explainable, which could be particularly important in domains such as medical document ranking.

7 Conclusion

In this paper, we introduce Rank-R1, which uses training based on the GRPO reinforcement learning strategy to enhance the reasoning process of an LLM-based Setwise reranker. Our training approach does not require human-annotated reasoning data; instead, it relies only on document relevance assessments. We find that RL training performs competitively with supervised fine-tuning on in-domain tasks. However, in reasoning-intensive ranking tasks,

Query: common minerals list

Zeroshot Rank-R1 Completion:

```
<think>The query is "common minerals list". Document [3] talks about mica being a common rock-forming mineral. Document [19] mentions that chemical sedimentary rocks are formed through precipitation of minerals. However, it does not provide a comprehensive list of common minerals. Documents [1], [2], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], and [20] do not mention anything related to common minerals.</think>
<answer>[3]</answer>
```

GRPO-Trained Rank-R1 Completion:

```
<think>The query is "common minerals list". Document [3] talks about mica, which is a common rock-forming mineral. Document [4] also mentions common rock-forming minerals. Both documents seem relevant to the query. However, document [4] provides a list of common rock-forming minerals, which directly answers the query.</think>
<answer>[4]</answer>
```

Figure 3: Example of completion. The Rank-R1 used in this example is based on Qwen2.5-3B-Instruct. The GRPO-Trained Rank-R1 gives the correct answer.

Rank-R1 based on a 14B LLM achieves considerably higher effectiveness than the SOTA non-reasoning Listwise reranker, highlighting the importance of incorporating a reasoning process in document reranking. We have made our code open-source at <https://github.com/ielab/llm-rankers/tree/main/Rank-R1>.

A further direction is to extend Rank-R1 from single-turn reranking over a fixed candidate set to multi-turn, session-based search for “deep research” applications [11, 40]. Concretely, we envision a hierarchical agent in which a planner decomposes an information need into sub-questions and iterative query reformulations, a retriever proposes evidence at each step, and Rank-R1 serves as the reasoning reranker that triages and prioritizes sources across turns, with a learned stopping policy and a memory that accumulates verified facts while penalizing redundancy. Training such an agent suggests long-horizon RL with session-level rewards (e.g., final answer quality, citation faithfulness, evidence coverage and diversity) and shaping signals based on per-turn retrieval gains, novelty, calibration, and compute cost. This setting also motivates studying credit assignment between planning and reranking policies, uncertainty-aware decisions (including abstention when evidence is insufficient), and robustness to domain shifts. Architecturally, we will explore Rank-R1 as a reusable module within multi-turn tool-using agents, examining how its GRPO-trained reasoning interacts with query planning and memory.

Acknowledgments

This work was partly supported by the Institute of Information Communications Technology Planning Evaluation (IITP) grant funded by the Korean Government (MSIT) (No. RS-2024-00457882, National AI Research Lab Project).

References

- [1] Haodong Chen, Shengyao Zhuang, Zheng Yao, Guido Zuccon, and Teerapong Leelanupab. 2026. Where Relevance Emerges: A Layer-Wise Study of Internal Attention for Zero-Shot Re-Ranking. arXiv:2602.22591 [cs.IR] doi:10.1145/3805712.3808558
- [2] Shijie Chen, Bernal Jimenez Gutierrez, and Yu Su. 2025. Attention in Large Language Models Yields Efficient Zero-Shot Re-Rankers. In *The Thirteenth International Conference on Learning Representations*. <https://openreview.net/forum?id=yzloNYH3QN>
- [3] Yiqun Chen, Qi Liu, Yi Zhang, Weiwei Sun, Xinyu Ma, Wei Yang, Daiting Shi, Jiaxin Mao, and Dawei Yin. 2025. Tourrank: Utilizing large language models for documents ranking with a tournament-inspired strategy. In *Proceedings of the ACM on Web Conference 2025*. 1638–1652.
- [4] Zijian Chen, Ronak Pradeep, and Jimmy Lin. 2025. Accelerating Listwise Re-ranking: Reproducing and Enhancing FIRSt. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 3165–3172.
- [5] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, and Daniel Campos. 2021. Overview of the TREC 2020 deep learning track. arXiv:2102.07662 [cs.IR] <https://arxiv.org/abs/2102.07662>
- [6] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2020. Overview of the TREC 2019 deep learning track. arXiv:2003.07820 [cs.IR] <https://arxiv.org/abs/2003.07820>
- [7] Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2023. Tevatron: An Efficient and Flexible Toolkit for Neural Retrieval. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (Taipei, Taiwan) (SIGIR '23)*. Association for Computing Machinery, New York, NY, USA, 3120–3124. doi:10.1145/3539618.3591805
- [8] Daya Guo, Dejia Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv:2501.12948 (2025). <https://arxiv.org/abs/2501.12948>
- [9] Fang Guo, Wenyu Li, Honglei Zhuang, Yun Luo, Yafu Li, Le Yan, Qi Zhu, and Yue Zhang. 2025. MCRanker: Generating diverse criteria on-the-fly to improve pointwise LLM rankers. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*. 944–953.
- [10] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=nZevKeeFy9>
- [11] Yuxuan Huang, Yihang Chen, Haozheng Zhang, Kang Li, Huichi Zhou, Meng Fang, Linyi Yang, Xiaoguang Li, Lifeng Shang, Songcen Xu, Jianye Hao, Kun Shao, and Jun Wang. 2025. Deep Research Agents: A Systematic Examination And Roadmap. arXiv:2506.18096 [cs.AI] <https://arxiv.org/abs/2506.18096>
- [12] Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. Search-R1: Training LLMs to Reason and Leverage Search Engines with Reinforcement Learning. arXiv:2503.09516 [cs.CL] <https://arxiv.org/abs/2503.09516>
- [13] Dawn Lawrie, Efsun Kayi, James Mayfield, Eugene Yang, Andrew Yates, and Douglas W Oard. 2025. A Reproducibility Study of LLM Setwise Reranker with Heapsort. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 3145–3154.
- [14] Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. 2022. Holistic evaluation of language models. arXiv:2211.09110 (2022). <https://arxiv.org/abs/2211.09110>
- [15] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: A Python Toolkit for Reproducible Information Retrieval Research with Sparse and Dense Representations. In *Proceedings of the 44th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2021)*. 2356–2362.
- [16] Wenhan Liu, Xinyu Ma, Weiwei Sun, Yutao Zhu, Yuchen Li, Dawei Yin, and Zhicheng Dou. 2026. ReasonRank: Empowering Passage Ranking with Strong Reasoning Ability. arXiv:2508.07050 [cs.IR] <https://arxiv.org/abs/2508.07050>
- [17] Xueguang Ma, Liang Wang, Nan Yang, Furu Wei, and Jimmy Lin. 2024. Fine-Tuning LLaMA for Multi-Stage Text Retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (Washington DC, USA) (SIGIR '24)*. Association for Computing Machinery, New York, NY, USA, 2421–2425. doi:10.1145/3626772.3657951
- [18] Xueguang Ma, Xinyu Zhang, Ronak Pradeep, and Jimmy Lin. 2023. Zero-Shot Listwise Document Reranking with a Large Language Model. arXiv:2305.02156 [cs.IR] <https://arxiv.org/abs/2305.02156>
- [19] Tong Niu, Shafiq Joty, Ye Liu, Caiming Xiong, Yingbo Zhou, and Semih Yavuz. 2024. JudgeRank: Leveraging Large Language Models for Reasoning-Intensive Reranking. arXiv:2411.00142 [cs.CL] <https://arxiv.org/abs/2411.00142>
- [20] Rodrigo Nogueira, Zhiying Jiang, and Jimmy Lin. 2020. Document Ranking with a Pretrained Sequence-to-Sequence Model. arXiv:2003.06713 [cs.IR] <https://arxiv.org/abs/2003.06713>
- [21] OpenAI et al. 2024. GPT-4 Technical Report. arXiv:2303.08774 [cs.CL] <https://arxiv.org/abs/2303.08774>
- [22] Jakub Podolak, Leon Perić, Mina Janićijević, and Roxana Petcu. 2025. Beyond reproducibility: Advancing zero-shot LLM reranking efficiency with setwise insertion. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 3205–3213.
- [23] Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. 2023. RankVicuna: Zero-Shot Listwise Document Reranking with Open-Source Large Language Models. arXiv:2309.15088 [cs.IR] <https://arxiv.org/abs/2309.15088>
- [24] Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. 2023. RankZephyr: Effective and Robust Zero-Shot Listwise Reranking is a Breeze! arXiv:2312.02724 [cs.IR] <https://arxiv.org/abs/2312.02724>
- [25] Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Le Yan, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, and Michael Bendersky. 2024. Large Language Models are Effective Text Rankers with Pairwise Ranking Prompting. In *Findings of the Association for Computational Linguistics: NAACL 2024*, Kevin Duh, Helena Gomez, and Steven Bethard (Eds.). Association for Computational Linguistics, Mexico City, Mexico, 1504–1518. doi:10.18653/v1/2024.findings-naacl.97
- [26] Owen et al. 2025. Qwen2.5 Technical Report. arXiv:2412.15115 [cs.CL] <https://arxiv.org/abs/2412.15115>
- [27] Devendra Sachan, Mike Lewis, Mandar Joshi, Armen Aghajanyan, Wen-tau Yih, Joelle Pineau, and Luke Zettlemoyer. 2022. Improving Passage Retrieval with Zero-Shot Question Generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 3781–3797. doi:10.18653/v1/2022.emnlp-main.249
- [28] Rulin Shao, Rui Qiao, Varsha Kishore, Niklas Muennighoff, Xi Victoria Lin, Daniela Rus, Bryan Kian Hsiang Low, Sewon Min, Wen tau Yih, Pang Wei Koh, and Luke Zettlemoyer. 2025. ReasonIR: Training Retrievers for Reasoning Tasks. In *Second Conference on Language Modeling*. <https://openreview.net/forum?id=kkBCNLMbGj>
- [29] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. DeepSeek-Math: Pushing the Limits of Mathematical Reasoning in Open Language Models. arXiv:2402.03300 [cs.CL] <https://arxiv.org/abs/2402.03300>
- [30] Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. 2025. R1-Searcher: Incentivizing the Search Capability in LLMs via Reinforcement Learning. arXiv:2503.05592 [cs.AI] <https://arxiv.org/abs/2503.05592>
- [31] Hongjin Su, Howard Yen, Mengzhou Xia, Weijia Shi, Niklas Muennighoff, Han yu Wang, Liu Haisu, Quan Shi, Zachary S Siegel, Michael Tang, Ruoxi Sun, Jinsung Yoon, Sercan O Arik, Danqi Chen, and Tao Yu. 2025. BRIGHT: A Realistic and Challenging Benchmark for Reasoning-Intensive Retrieval. In *The Thirteenth International Conference on Learning Representations*. <https://openreview.net/forum?id=ykucs5q381b>
- [32] Shuoqi Sun, Shengyao Zhuang, Shuai Wang, and Guido Zuccon. 2025. An investigation of prompt variations for zero-shot LLM-based rankers. In *European Conference on Information Retrieval*. Springer, 185–201.
- [33] Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. 2023. 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*, Houa Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 14918–14937. doi:10.18653/v1/2023.emnlp-main.923
- [34] Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. 2023. Zephyr: Direct Distillation of LM Alignment. arXiv:2310.16944 [cs.LG] <https://arxiv.org/abs/2310.16944>
- [35] Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Galouédec. 2020. TRL: Transformer Reinforcement Learning. <https://github.com/huggingface/trl>.
- [36] Orion Weller, Kathryn Ricci, Eugene Yang, Andrew Yates, Dawn Lawrie, and Benjamin Van Durme. 2025. Rank1: Test-Time Compute for Reranking in Information Retrieval. arXiv:2502.18418 [cs.IR] <https://arxiv.org/abs/2502.18418>
- [37] Zhichao Xu. 2024. RankMamba: Benchmarking Mamba’s Document Ranking Performance in the Era of Transformers. arXiv:2403.18276 [cs.IR] <https://arxiv.org/abs/2403.18276>
- [38] Eugene Yang, Andrew Yates, Kathryn Ricci, Orion Weller, Vivek Chari, Benjamin Van Durme, and Dawn Lawrie. 2025. Rank-K: Test-Time Reasoning for Listwise Reranking. arXiv:2505.14432 [cs.IR] <https://arxiv.org/abs/2505.14432>
- [39] Weihao Zeng, Yuzhen Huang, Wei Liu, Keqing He, Qian Liu, Zejun Ma, and Junxian He. 2025. 7B Model and 8K Examples: Emerging Reasoning with Reinforcement Learning is Both Effective and Efficient. <https://hkust-nlp.notion.site/simplerl-reason>. Notion Blog.

- [40] Yuxiang Zheng, Dayuan Fu, Xiangkun Hu, Xiaojie Cai, Lyumanshan Ye, Pengrui Lu, and Pengfei Liu. 2025. DeepResearcher: Scaling Deep Research via Reinforcement Learning in Real-world Environments. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (Eds.). Association for Computational Linguistics, Suzhou, China, 414–431. doi:10.18653/v1/2025.emnlp-main.22
- [41] Honglei Zhuang, Zhen Qin, Kai Hui, Junru Wu, Le Yan, Xuanhui Wang, and Michael Bendersky. 2024. Beyond Yes and No: Improving Zero-Shot LLM Rankers via Scoring Fine-Grained Relevance Labels. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, Kevin Duh, Helena Gomez, and Steven Bethard (Eds.). Association for Computational Linguistics, Mexico City, Mexico, 358–370. doi:10.18653/v1/2024.naacl-short.31
- [42] Shengyao Zhuang, Bing Liu, Bevan Koopman, and Guido Zuccon. 2023. Open-source Large Language Models are Strong Zero-shot Query Likelihood Models for Document Ranking. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 8807–8817. doi:10.18653/v1/2023.findings-emnlp.590
- [43] Shengyao Zhuang, Honglei Zhuang, Bevan Koopman, and Guido Zuccon. 2024. A Setwise Approach for Effective and Highly Efficient Zero-shot Ranking with Large Language Models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Washington DC, USA) (SIGIR '24). Association for Computing Machinery, New York, NY, USA, 38–47. doi:10.1145/3626772.3657813
- [44] Guido Zuccon, Shengyao Zhuang, and Xueguang Ma. 2025. R2LLMs: Retrieval and Ranking with LLMs. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 4106–4109.