

Don’t “Overthink” Pointwise Reranking: Is Reasoning Truly Necessary?

Nour Jedidi¹ Yung-Sung Chuang² James Glass² Jimmy Lin¹

¹University of Waterloo ²Massachusetts Institute of Technology
njedidi@uwaterloo.ca

Abstract

With the success of reasoning models across complex natural language tasks, researchers in the information retrieval (IR) community have begun exploring how similar reasoning capabilities can be integrated into passage rerankers built on large language models (LLMs). These methods typically employ an LLM to produce an explicit, step-by-step reasoning process before arriving at a final relevance prediction. But *does reasoning actually improve pointwise reranking?* We investigate this question through controlled comparisons of Rank1, a pointwise reasoning reranker, with two non-reasoning counterparts. Across in-domain and out-of-domain datasets, we find that Rank1 generally underperforms Rank0, an otherwise equivalent *non-reasoning* pointwise reranker trained using the same backbone LLM and underlying training data as Rank1. To isolate the contribution of Rank1’s inference-time reasoning, we then replace its generated reasoning process with a fixed dummy sequence. Surprisingly, this variant, Rank1-NoReason, can also outperform Rank1. Analysis of Rank1’s score distributions provides a likely explanation: once Rank1 deems a passage relevant, it assigns nearly identical relevance scores, limiting its ability to distinguish degrees of relevance and ultimately harming its ranking accuracy. We release all code at <https://github.com/nourj98/dont-overthink-reranking>.

1 Introduction

A common approach to improve the effectiveness of large language models (LLMs) is to “scale up test-time compute” by having the model generate an intermediate reasoning process where it “thinks” before returning a final response. As training LLMs to “reason then answer” has proven highly effective across complex natural language tasks, reasoning has become an increasingly standard capability in modern LLMs, many of which are now designed to do so by default.



Figure 1: Average nDCG@10 of reasoning pointwise rerankers (Rank1) compared to non-reasoning variants (Rank0 and Rank1-NoReason).

Motivated by this success, researchers in the information retrieval (IR) community have begun exploring methods to equip passage rerankers built on LLMs with similar reasoning capabilities (Weller et al., 2025; Zhuang et al., 2026; Liu et al., 2026; Yang et al., 2025), especially thanks to the introduction of reasoning-intensive retrieval benchmarks such as BRIGHT (Su et al., 2025). At a high level, given an input query and passage (or list of passages), these *reasoning rerankers* are trained to first generate an explicit reasoning process before producing a final relevance score for the query-passage pair (or, if the input is a list of passages, an ordering of such a list).

While various reasoning rerankers have been proposed and report strong gains, many of these works lack a thorough comparison to the *equivalent* – same backbone model, same training data – non-reasoning variants. Instead, the primary non-reasoning baselines are limited to passage rerankers built on previous generation LLMs – for example, RankLLaMA (Ma et al., 2024) (based on Llama-2), RankZephyr (Pradeep et al., 2023) (based on Zephyr), or MonoT5/RankT5 (Nogueira et al., 2020; Zhuang et al., 2023) (based on T5) – while the proposed models are leveraging the latest LLMs. Comparisons with equivalent, non-reasoning variants are typically relegated to abla-

tion studies or are only studied for a limited subset of presented datasets, raising the question of whether reasoning rerankers are *truly* more effective or if the reported gains are simply due to improvements from using stronger backbone LLMs.

In this paper, we address this gap by revisiting baselines for Rank1 (Weller et al., 2025), a reasoning pointwise reranker that generates a reasoning chain prior to producing a relevance score for an independent query-passage pair. Through controlled comparisons of Rank1 with otherwise equivalent – same backbone model, same training data – non-reasoning variants, we ask: *does reasoning actually improve pointwise reranking?* Our focus on Rank1 is simple: as the pioneering work in the reasoning reranking space, it has become a go-to baseline in follow-up research. Understanding its effectiveness in comparison to equivalent, non-reasoning counterparts is critical to ensure progress in developing more effective reasoning rerankers.

To answer our core research question, we decompose it into two sub-questions: (1) does a reasoning pointwise reranker outperform an otherwise equivalent non-reasoning pointwise reranker? and (2) does a reasoning pointwise reranker benefit from its own inference-time reasoning process? We investigate these questions through controlled comparisons of Rank1 with two simple yet effective non-reasoning counterparts which leverage the same backbone model and training data: (1) **Rank0**, a non-reasoning LLM-based pointwise reranker that directly classifies query-passage pairs as relevant or non-relevant (Nogueira et al., 2020; Ma et al., 2024); and (2) **Rank1-NoReason**, a modified version of Rank1, in which the explicit reasoning process is disabled at inference time, effectively transforming Rank1 into a standard non-reasoning pointwise reranker. The central findings of our experiments can be summarized as follows and are shown in Figure 1:

- Under equivalent conditions – same backbone model, same training data – reasoning pointwise rerankers are unable to outperform their non-reasoning counterparts. Across multiple LLM sizes, we find Rank0 to be more effective in-domain and out-of-domain versus Rank1.
- In fact, we find that reasoning pointwise rerankers *do not even benefit from their own inference-time reasoning process*. As shown in Figure 1, Rank1-NoReason outperforms Rank1 in-domain (MS MARCO) for the 7B and 14B

model sizes, and is always more effective out-of-domain (BRIGHT).

- Our analysis indicates that this likely stems from Rank1’s reasoning process pushing it toward polarized relevance scores, limiting its ability to distinguish degrees of relevance among passages. While improving Rank1’s ability to distinguish degrees of relevance consistently improves its effectiveness, it is still less effective than Rank0, suggesting that explicit reasoning may not be worth the “bang for the buck” for pointwise rerankers.

Our findings build upon research which questions the necessity of reasoning for LLMs (Ma et al., 2025; Sprague et al., 2025). We hope our work not only encourages future research in improving reasoning for pointwise reranking tasks, but also highlights the importance of comparing against strong, simple, and fair baselines when developing new methodologies.

2 Pointwise Reranking with LLMs

Preliminaries. The goal of information retrieval is to identify relevant passages from a large collection of n texts, denoted by $\mathcal{C} = \{P_1, P_2, \dots, P_n\}$, given a user-issued query, q . Current IR systems typically employ a multi-stage pipeline where a first-stage retriever fetches an initial set of k passages from $\mathcal{C}$ and a *reranker* reorders these top- k passages, where $k \ll n$, to produce a more accurate ranking. *Pointwise rerankers*, the family of rerankers we focus on in this study, perform this reordering task by independently producing a relevance score S for every query-passage pair in the top- k . The passages are then sorted in descending order of S .

Data. Pointwise rerankers are fine-tuned to directly produce a relevance score given a query-passage pair. In this study, we fine-tune pointwise rerankers using the training data released by Weller et al. (2025).¹ The dataset augments MS MARCO (Bajaj et al., 2016) with reasoning chains and final relevance predictions generated by DeepSeek-R1 (Guo et al., 2025), consisting of approximately 386K quadruples of the form (query, passage, R1’s reasoning chain, R1’s relevance label). This dataset is well suited for our study as it provides a clean

¹<https://huggingface.co/datasets/jhu-clsp/rank1-training-data>

testbed for training pointwise rerankers and properly ablating the influence of reasoning while holding training data fixed.

Rank1. Rank1 is a pointwise reranker trained on the aforementioned quadruples to first generate a reasoning process r_i before predicting a “true” (relevant) or “false” (non-relevant) relevance label.

At inference time, for each query-passage pair (q, P_i) among the top- k passages, Rank1 computes a relevance score S by applying a softmax over the logits of the “true” and “false” tokens, conditioned on r_i :

$$S = \text{softmax}\left(\frac{z_{\text{true}}(q, P_i, r_i)}{z_{\text{false}}(q, P_i, r_i)}\right)_{\text{true}} \quad (1)$$

Here, $z_{\text{true}}(q, P_i, r_i)$ and $z_{\text{false}}(q, P_i, r_i)$ denote the logits assigned by the LLM to the “true” and “false” tokens, respectively, after generating r_i . The subscript “true” selects the normalized probability assigned to the “true” token, which is used as the relevance score S .

Rank1-NoReason. To isolate the contribution of Rank1’s generated reasoning at inference time, we introduce Rank1-NoReason, an ablation that replaces r_i with a fixed dummy reasoning process d_r (Ma et al., 2025). Rank1-NoReason uses the exact same trained model and scoring procedure as Rank1; the only difference is that its relevance score is conditioned on d_r rather than r_i :

$$S = \text{softmax}\left(\frac{z_{\text{true}}(q, P_i, d_r)}{z_{\text{false}}(q, P_i, d_r)}\right)_{\text{true}} \quad (2)$$

Unless otherwise specified, we set d_r to <think> Okay, I have finished thinking. </think>.

Rank0. To understand how Rank1 compares to non-reasoning pointwise rerankers trained under equivalent training conditions (same backbone model, same training data), we also train Rank0, a pointwise reranker which uses the same underlying examples as Rank1 but discards the reasoning process from the quadruples, retaining only (query, passage, R1’s relevance label) triples. As such, Rank0 is fine-tuned to directly produce the “true” or “false” token given (q, P_i) :

$$S = \text{softmax}\left(\frac{z_{\text{true}}(q, P_i)}{z_{\text{false}}(q, P_i)}\right)_{\text{true}} \quad (3)$$

We note that methods like RankLLaMA (Ma et al., 2024) train non-reasoning pointwise rerankers with

hard negatives sampled from the top-ranking results of a first-stage retriever. However, as our goal is to keep the training setup as close as possible to that used by the original Rank1 authors (Weller et al., 2025), we do not consider hard negatives in our Rank0 training.

3 Does Reasoning Improve Pointwise Rerankers?

Now with Rank0 and Rank1-NoReason in hand, we can fairly study the impact of reasoning on pointwise rerankers. In this section, we do so by focusing on the following comparisons: Rank0 vs. Rank1 and Rank1 vs. Rank1-NoReason.

3.1 Experimental Setup

As the primary backbones for Rank0 and Rank1, we leverage the Qwen2.5 base models (Yang et al., 2024) ranging in size from 3B to 14B parameters. To train Rank0, we fine-tune each model using LoRA (Hu et al., 2022) for one epoch with rank 32 and alpha 64.² For Rank1, we primarily consider the Qwen2.5-based model checkpoints released by Weller et al. (2025); however, to ensure that any differences in the effectiveness of Rank0 and Rank1 are not due to differences in how the models were trained, we additionally train our own variant of Rank1, Rank1 (our impl.), at the 3B and 7B model sizes, following the same training process as Rank0, except that it utilizes the full quadruples.

We evaluate all rerankers on in-domain and out-of-domain retrieval datasets. For in-domain evaluation, we leverage passage ranking datasets based on MS MARCO v1 – TREC DL19 and TREC DL20 (Craswell et al., 2020, 2021b) – and based on MS MARCO v2 – TREC DL21, TREC DL22, and TREC DL23 (Craswell et al., 2021a, 2022, 2023). For out-of-domain evaluation, we focus on BRIGHT (Su et al., 2025), a reasoning-intensive retrieval benchmark. We report nDCG@10, the official metric for MS MARCO and BRIGHT.

All models rerank the top-100 passages from BM25. For BRIGHT, we construct the candidate set using BM25 with GPT-4-generated chain-of-thought (CoT) reasoning steps as queries (BM25 + GPT-4 CoT). Following Weller et al. (2025), however, the rerankers themselves receive only the original BRIGHT query – and *not* the GPT-4-generated CoT – when scoring the candidate passages. For MS MARCO, we implement BM25 using Pyserini

²These parameters follow Weller et al. (2025).

	MS MARCO v1		MS MARCO v2			Avg.
	DL19	DL20	DL21	DL22	DL23	
BM25	50.6	48.0	44.6	26.9	26.3	39.3
+ Qwen2.5-3B						
Rank0	72.5	68.9[†]	69.4[†]	51.4[†]	45.5[†]	61.5
Rank1 (our impl.)	70.4	66.4	65.9	45.2	41.3	57.8
Rank1	70.1	64.5	65.3	45.2	40.7	57.2
+ Qwen2.5-7B						
Rank0	74.6[†]	70.0[†]	70.9[†]	50.3	46.3[†]	62.4
Rank1 (our impl.)	70.3	64.3	65.9	45.6	41.1	57.4
Rank1	68.4	65.1	65.8	44.7	41.6	57.1
+ Qwen2.5-14B						
Rank0	73.3[†]	68.7[†]	70.7[†]	49.6[†]	47.3[†]	61.9
Rank1	66.3	65.0	63.1	44.8	41.7	56.2

Table 1: Rank0 versus Rank1 on MS MARCO. **Bold** denotes best reranker under each Qwen2.5 model. [†] denotes a statistically significant improvement over Rank1 based on a paired Student’s t-test ($p < 0.05$) (Bassani, 2022).

	StackExchange							Coding		Theorem-based			Avg.
	Bio.	Earth.	Econ.	Psy.	Rob.	Stack.	Sus.	Leet.	Pony	AoPS	TheoQ.	TheoT.	
BM25 + GPT-4 CoT	53.6	54.1	24.3	38.7	18.9	27.7	26.3	19.3	17.6	3.9	19.2	20.8	27.0
+ Qwen2.5-3B													
Rank0	41.6	27.1	20.9	31.9	22.2	16.9	30.3	13.2	42.0	2.7	16.2	30.6	24.6
Rank1 (our impl.)	37.3	27.8	20.7	33.1	18.3	24.3	25.2	11.3	26.2	4.7	20.7	34.0	23.6
Rank1	41.8	25.6	18.4	29.3	15.5	18.4	25.8	16.1	24.9	4.7	21.7	32.7	22.9
+ Qwen2.5-7B													
Rank0	47.1	38.0	28.1	44.1	26.1	29.5	36.5	19.3	37.5	4.6	22.4	39.4	31.0
Rank1 (our impl.)	47.0	35.4	24.0	35.2	20.0	25.2	31.0	15.1	36.0	5.9	22.2	36.6	27.8
Rank1	48.8	36.7	20.8	35.0	22.0	18.7	36.2	12.7	31.2	6.3	23.7	37.8	27.5
+ Qwen2.5-14B													
Rank0	52.9	45.5	30.6	46.1	28.5	32.3	38.1	24.1	33.1	8.0	26.8	40.7	33.9
Rank1	49.3	37.7	22.6	35.2	22.5	20.8	33.6	17.7	33.2	8.4	22.5	41.4	28.7

Table 2: Rank0 versus Rank1 on BRIGHT. **Bold** denotes best reranker under each Qwen2.5 model. The original BRIGHT queries – and *not* the GPT-4-generated CoT – are used when reranking.

(Lin et al., 2021) and for BRIGHT, we follow the implementation from the BRIGHT codebase. LLM training was performed using Hugging Face (Wolf et al., 2019) and LLM inference was performed with vLLM (Kwon et al., 2023).

3.2 Rank0 vs. Rank1

We begin by comparing Rank0 to Rank1. Recall that Rank0 and Rank1 are both pointwise rerankers fine-tuned under equivalent training conditions (same backbone model, same training data), but differ in whether they are trained to reason.

Does a reasoning pointwise reranker outperform an otherwise equivalent non-reasoning pointwise reranker? In short, the answer is *no*: our results suggest that training a Rank1-style pointwise reranker provides no general effectiveness advantage over Rank0. We now describe our findings in more detail, which are presented in Tables 1 and 2 over MS MARCO and BRIGHT.

We begin by noting that Rank1 is comparable to Rank1 (our impl.), achieving a similar average

nDCG@10 across MS MARCO and BRIGHT for the 3B and 7B model sizes. This validates that any differences in effectiveness between Rank0 and Rank1 cannot be attributed to potential differences in training conditions.³

On MS MARCO, we find that Rank0 outperforms Rank1 by an average of 3.7, 5.0, and 5.7 points for the 3B, 7B, and 14B model sizes, respectively. These gains are consistent, with Rank0 being more effective on each individual dataset. On BRIGHT – a benchmark built on reasoning-intensive queries – we find a similar story where removing reasoning improves accuracy: Rank0 outperforms Rank1, achieving 1.0, 3.2, and 5.2 points higher average nDCG@10 across the 3B, 7B, and 14B model sizes. Surprisingly, this effectiveness gap appears to *increase* as the backbone LLM gets larger, suggesting that scaling up benefits Rank0 more than Rank1. At the 3B and 7B scales, Rank1 is consistently stronger on the AoPS and TheoQ

³Unless explicitly noted, in this section we compare against the more effective Rank1 implementation.

Qwen2.5	Method	MS MARCO	BRIGHT
3B	Rank1 (our impl.)	57.8	23.6
	Rank1-NoReason	58.3	23.4
	Rank1	57.2	22.9
	Rank1-NoReason	55.0	24.2
7B	Rank1 (our impl.)	57.4	27.8
	Rank1-NoReason	58.8	30.8
	Rank1	57.1	27.5
	Rank1-NoReason	58.6	31.5
14B	Rank1	56.2	28.7
	Rank1-NoReason	57.3	34.7

Table 3: Studying the effect of removing the reasoning process from Rank1. Results on MS MARCO and BRIGHT are averages across the given datasets.

datasets, which are math-based tasks to retrieve relevant passages based on theorem-based questions. This result is consistent with [Sprague et al. \(2025\)](#), who found that reasoning is most helpful on problems which require mathematical reasoning. However, this gap appears to close at the 14B reranker scale.

3.3 Rank1 vs. Rank1-NoReason

We next compare Rank1 to Rank1-NoReason. Recall that Rank1 and Rank1-NoReason are the *exact same model*, but differ in whether the model computes its relevance score conditioned on its own reasoning or on a fixed dummy reasoning process.

Does a reasoning pointwise reranker benefit from its own inference-time reasoning process? Surprisingly, we find, again, that the answer is *no*: Rank1 can actually be more effective when using a fixed dummy reasoning process instead of its own generated reasoning process.

The results are shown in Table 3. Across the 7B and 14B Rank1 models, Rank1-NoReason consistently outperforms Rank1 on both MS MARCO and BRIGHT. For the 7B models, the gains are as large as 1.5 and 4.0 points on MS MARCO and BRIGHT, respectively; for the 14B model, the corresponding gains are 1.1 and 6.0 points. For the 3B model, the gains from Rank1-NoReason are less consistent, but we do find that Rank1-NoReason is generally around the same average nDCG@10 as Rank1. Similar to the result shown in Section 3.2, as Rank1’s size increases, the benefits of removing the reasoning process *increase*. In fact, on the BRIGHT dataset, Rank1-NoReason at the 7B and 14B model sizes is on par with Rank0 in terms of effectiveness, on average.

How does the text used for d_r impact the ac-

Reasoning Process	MS MARCO
Rank1 (14B) generated reasoning	56.2
<i>Okay, I have finished thinking</i>	57.3
<i>Blank (empty string)</i>	55.7
<i>Passage</i>	56.6
<i>Query & Passage</i>	58.1

Table 4: Variations of d_r for Rank1-NoReason (14B).

curacy of Rank1-NoReason? In Table 4, we study different dummy reasoning variations for Rank1-NoReason (14B) on MS MARCO. The results show that completely leaving the reasoning blank – i.e., `<think></think>` – results in the largest drop in effectiveness compared to Rank1 (56.2 vs. 55.7). On the other hand, simply repeating the query-passage pair – i.e., `<think>{query}\n{passage}</think>` – results in a 1.9-point gain in nDCG@10, on average. In other words, repeating the input in the reasoning chain is more helpful than the reasoning chain generated by Rank1.

The results in this section suggest that it is in fact the generated reasoning process that is limiting Rank1’s effectiveness: If Rank1-NoReason at larger model sizes is competitive with Rank0, this means Rank1 is in fact learning how to predict relevance, but the reasoning it generates is not guiding it toward *better reranking ability*. We explore potential explanations in the next section.

4 Why Does Reasoning Hurt Pointwise Rerankers?

One reason why Rank1 may perform worse than Rank0 is because Rank1 has poorly calibrated probabilities for ranking due to conclusions made by its reasoning process. For example, Rank1 may almost always assign very high probabilities when its reasoning concludes that a passage is relevant, and thus may not be able to reflect that one passage is *more* relevant than another passage. On the other hand, as Rank0 is trained to only output “true” or “false”, it may implicitly learn to output scores that account for one passage being more relevant than another passage. Due to this, we hypothesize that Rank0 can better model the *relative* relevance of query-passage pairs. In this section, we investigate this hypothesis.

4.1 Relevance Classification Comparison

To begin, we study how different reranking methods compare as simple relevance classifiers, ignor-

	MS MARCO v1						MS MARCO v2								
	DL19			DL20			DL21			DL22			DL23		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Rank0	71.4	80.3	75.6	54.5	79.3	64.6	56.4	87.1	68.5	49.1	70.0	57.6	39.7	66.9	49.8
Rank1	65.6	78.8	71.6	50.6	81.7	62.5	54.8	88.3	67.6	41.2	72.0	52.4	34.7	62.3	44.6
+ Self-Consistency	65.1	83.6	73.2	49.7	83.2	62.2	54.4	90.3	67.9	42.7	74.2	54.2	34.5	63.2	44.6
Rank1-NoReason	61.6	81.2	70.1	46.0	83.4	59.2	52.8	89.9	66.5	37.6	77.5	50.6	33.6	69.6	45.3

Table 5: Relevance classification effectiveness (Precision, Recall, and F1-score) across reranker variants. Rank1 + Self-Consistency is discussed in Section 5.

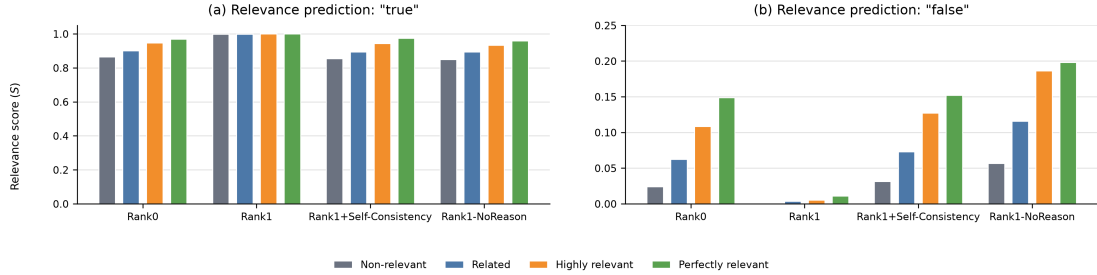


Figure 2: Average relevance score S by ground-truth grade on MS MARCO, conditioned on predicting the passage as (a) relevant (“true”) or (b) non-relevant (“false”). Rank1 + Self-Consistency is discussed in Section 5.

ing their reranking accuracy measured by metrics like nDCG@10, which, ultimately, is what we *really* care about. Doing so will allow us to better understand how much we can attribute differences in effectiveness to simply being worse relevance classifiers. In most cases, better relevance classification *should* result in better reranking accuracy.

For this experiment, we set $y_{pred} = 1$ if $S > 0.5$, and $y_{pred} = 0$ otherwise. For the ground truth relevance judgements, we set judgements ≥ 2 (corresponding to highly relevant and perfectly relevant) as positive labels, and the rest as negative labels, following standard practice used for binary measures in IR (MacAvaney et al., 2022). The results are in Table 5.

Comparing Rank0 to Rank1, we find that in terms of F1-score and precision, Rank0 is consistently stronger than Rank1. Interestingly, Rank1-NoReason generally underperforms Rank1 as a binary relevance classifier, yielding lower precision on all five datasets and lower F1 on four, yet achieves higher average nDCG@10, as discussed in Section 3.3. This discrepancy suggests that differences in retrieval effectiveness cannot *only* be attributed to relevance classification effectiveness.

4.2 S Across Relevance Grades

We next take a closer look at the relevance scores emitted by Rank0, Rank1, and Rank1-NoReason across the different ground-truth relevance grades

(perfectly relevant, highly relevant, related, and non-relevant). These results are shown in Figure 2 and are split according to whether the model predicts the passage as relevant (“true”) or non-relevant (“false”).

Focusing on panel (a), which considers passages that the reranker predicts as relevant, Rank0 exhibits a desirable behavior: the average relevance score S decreases monotonically as the ground-truth label moves from perfectly relevant to non-relevant. That is, among passages that Rank0 scores highly – and thus ranks toward the top of the list – the ideal ordering is preserved on average, i.e., perfectly relevant $>$ highly relevant $>$ related $>$ non-relevant. On the other hand, Rank1 does not exhibit this behavior: whenever it predicts a passage to be relevant, it produces nearly identical S scores (≈ 1). In practice, this suggests that once Rank1 deems a passage relevant, there is minimal separation among the relevance scores it produces.

Put another way, when Rank0 produces a false positive, its score is, on average, still lower than the score assigned to a true positive. For example, across the MS MARCO datasets, the average S is 0.9555 for true positives, compared with 0.8881 for false positives. In contrast, Rank1 exhibits almost no separation between the two, assigning average scores of 0.9997 and 0.9982 to true positives and false positives, respectively.

Interestingly, when we remove Rank1’s reason-

Query	how to help a jammed finger
Passage	Jammed finger vs. broken finger. Share on Pinterest. A broken finger is often more painful than a jammed finger, although both may be treated using a splint. A broken finger will cause severe pain and swelling that lasts for hours or even days. While a jammed finger is painful, it is not usually severe.
Reasoning	Okay, let's see. The user is asking how to help a jammed finger. The passage provided talks about the difference between a jammed finger and a broken finger. It mentions that both might be treated with a splint, but a broken finger is more painful and has more swelling. First, the query is about treatment for a jammed finger. The passage does mention that a jammed finger can be treated with a splint, which is a direct answer to the query. However, the main focus of the passage is comparing jammed and broken fingers, not giving detailed treatment steps. It does say that both might use a splint, which is relevant. But the passage doesn't go into other treatment methods like icing, elevation, or when to see a doctor. So, the passage is somewhat relevant because it mentions a treatment (splint) for a jammed finger. But it's not comprehensive. The user might need more information, but the passage does provide a basic answer. However, the question is whether the passage is relevant, not if it's complete. Since it does address treatment (splint) for a jammed finger, even briefly, it's relevant. The comparison to a broken finger might also help in understanding when a jammed finger isn't severe. Therefore, the answer is true.
S	Rank0: 0.374 Rank1: 0.999 Rank1-NoReason: 0.810

Table 6: An example of Rank1’s (7B) reasoning process where it scores a partially relevant passage as highly relevant. We additionally provide the relevance score S for the provided query-passage pair.

ing process at inference time, as done with Rank1-NoReason, the monotonic decrease in the average S score across relevance grades is “recovered”. This further suggests that the reasoning process compresses Rank1’s positive relevance scores into a narrow range and thus weakens its ability to distinguish among passages that it deems relevant.

Based on these results, we believe that Rank1’s weaker ordering of passages it deems relevant may help explain the mismatch between relevance-classification effectiveness and nDCG@10 observed for Rank1 and Rank1-NoReason. Whereas binary classification only requires assigning each passage the correct label, effective reranking must also preserve relative degrees of relevance among passages predicted to be relevant. We further evaluate this explanation quantitatively in Section 4.4.

4.3 Qualitative Study

We next examine Rank1’s reasoning process for an example in which it scores a partially relevant passage as highly relevant. To select the example, we randomly select a passage that Rank0 produces a partially relevant prediction for.

The qualitative example can be found in Table 6. Looking at the reasoning process, we find that Rank1 explicitly mentions that the passage is “somewhat relevant”, but because it has to select between two binary options (relevant or non-relevant), it reasonably selects the passage as relevant. This subsequently makes S high (0.999) as Rank1 finishes its reasoning stating “Therefore, the answer is true”. This example shows how the reasoning

process can restrict the ability of Rank1 to capture varying degrees of relevance in its output probabilities even when the model explicitly reasons that the passage is not fully addressing the query.

4.4 Takeaways

The findings from Sections 4.1 and 4.2 suggest that distinguishing among different degrees of relevance within passages predicted to be relevant is an important factor in the effectiveness of pointwise rerankers, beyond binary relevance classification. This makes sense as maximizing nDCG@10 requires that rankers place the most relevant documents ahead of documents that are “less relevant”. To quantify this observation and examine whether score separation among passages predicted to be relevant is associated with nDCG@10, we fit a regression model to the per-query nDCG@10 scores for Rank0 (7B), Rank1 (7B), and Rank1-NoReason (7B) across all MS MARCO datasets.

In our regression, we consider two variables: Score Gap and precision (i.e., P@10). Score Gap denotes the difference between the average S scores assigned to true positives and false positives. A higher Score Gap indicates that the reranker does a better job at distinguishing labeled relevant documents from non-relevant ones that were incorrectly classified as relevant (which reflects a better ability to distinguish between degrees of relevance). P@10 measures the proportion of the top-10 ranked passages that are judged relevant. For this regression, Score Gap and P@10 are standardized. The R-squared of the regression is 0.684, indicating that

Score Gap and P@10 account for $\approx 68\%$ of the variability of nDCG@10.

$$\begin{aligned} \text{nDCG@10} = & \underset{(p<0.001)}{0.5641} + \\ & \underset{(p<0.001)}{0.0449} \cdot \text{Score Gap} + \\ & \underset{(p<0.001)}{0.1845} \cdot \text{P@10} \end{aligned} \quad (4)$$

As expected, the regression shows that having higher precision contributes the most to higher nDCG@10. When controlling for P@10, the ability to better separate the scores of true positives from the false positives has a positive and *significant* effect. This means that the ability to capture relative relevance is positively and significantly correlated with higher nDCG. This regression provides quantitative support for our hypothesis: As Rank1 is less capable of differentiating the nuances in relevance (always assigning a similar S score to all positive predictions), it results in lower nDCG@10.

5 Discussion and Related Work

Is improving Rank1 as simple as sampling more reasoning chains? A common approach to improve reasoning models and measure the uncertainty of their predictions is by making multiple inference calls to the model and selecting its most consistent answer as the final response. Such an approach can potentially provide a solution for enabling Rank1 to produce S scores which better differentiate across different degrees of relevance.

We explore this in Table 7 by implementing Rank1 using self-consistency (Wang et al., 2023), which we denote as Rank1 + Self-Consistency. Unlike the majority vote approach used by Wang et al. (2023), we average the predicted S values across eight sampled outputs from Rank1 to produce a continuous score suitable for reranking.

As shown in Figure 2, Rank1 begins to produce S scores that decrease monotonically as the ground-truth label moves from perfectly relevant to non-relevant. In turn, nDCG@10 improves by 1.8 points on MS MARCO and 2.5 points on BRIGHT. While this improves Rank1’s effectiveness, it is just as effective as simply injecting a dummy reasoning process into the same model (i.e., Rank1-NoReason), while being more compute-intensive. These results suggest that naive inference-time solutions are not enough to close the effectiveness gap between Rank1 and Rank0 or to consistently outperform Rank1-NoReason.

	MS MARCO	BRIGHT
Rank0 (7B)	62.4	31.0
Rank1-NoReason (7B)	58.8	30.8
Rank1 (7B)	57.1	27.5
+ Self-Consistency	58.9	30.0
w/ Inverse Training (our impl.)	61.6	30.5

Table 7: Exploring approaches for improving Rank1: self-consistency sampling and inverse training.

Can reasoning help pointwise rerankers? We believe our study supports the following conclusion: In Rank1’s current state, reasoning is not necessary for pointwise rerankers at inference time. *But is there potential for it to help?* This is unclear.

Past results have shown that the benefits of reasoning come primarily on tasks which involve math or logic and that chain-of-thought reasoning is in fact unnecessary for most tasks it is applied to (Sprague et al., 2025). Other work in the IR community (Ferraretto et al., 2023) has demonstrated that training LLM-based pointwise rerankers with synthetic chain-of-thought reasoning *can* be helpful, but with a critical caveat: the reasoning needs to be produced *after* the relevance prediction. They found that producing the reasoning first dropped nDCG@10 by 10 points.

On the other hand, Samarinas and Zamani (2025) demonstrated encouraging results by using *discrete*, non-binary labels: Could redesigning Rank1 to generate non-binary relevance labels be sufficient?

To answer this, we investigate a simple strategy to train Rank1 to produce non-binary labels in Appendix E. We find that while this underperforms Rank1 if the non-binary relevance labels are used directly as the relevance score (e.g., S), when the relevance labels are combined with first-stage retrieval scores, similar to what is done in Shao et al. (2025), using non-binary relevance labels can yield a 3.4-point improvement in nDCG@10 over Rank1 on BRIGHT, but is notably *still* less effective than Rank0 in the same hybrid setup. While promising, future work is needed to train a Rank1-style reranker that produces non-binary relevance labels and can outperform Rank0.

Can reasoning be useful as a data augmentation technique? Rank1-NoReason’s strong effectiveness raises an interesting question: Can reasoning be useful as an additional training signal, where, for example, the reranker is trained to generate a reasoning process, but at inference time, the reasoning process gets “turned off”? Our results sug-

	BRIGHT
ReasonIR	30.8
+ Rank0 (14B)	34.8
+ Rank-R1 (14B) [†]	29.7
+ Rank-K (32B) [†]	32.6
+ ReasonRank (32B) [†]	38.0

Table 8: Comparison of Rank0 to listwise[†] and setwise[‡] reasoning rerankers. Except for Rank0, numbers are copied directly from Liu et al. (2026).

gest this is a promising direction for future work, particularly for larger LLMs, as we found Rank1-NoReason to perform on par with Rank0 at the 7B and 14B scales. Could larger models (e.g., 32B+) see even more improvement?

Alternatively, what if Rank1 is trained to produce the reasoning chain *after* generating a relevance label (Ferraretto et al., 2023)? In Table 7, we test this “inverse training” approach: Rank1 is trained to first produce a relevance label followed by its reasoning process, but at inference time we only allow Rank1 to produce the relevance label. With inverse training, Rank1 can achieve similar effectiveness to Rank0 on MS MARCO (61.6 vs. 62.4) and BRIGHT (30.5 vs. 31.0).

One advantage of reasoning rerankers is their potential to improve explainability of ranking results (Weller et al., 2025; Zhuang et al., 2026). Rank1 with inverse training can serve as a middle ground between Rank1 and Rank0, where one gets the explainability benefits of a reasoning model (if, say, requested by the user) while preserving the stronger ranking capabilities of Rank0.

What about reasoning for other reranking paradigms? Since the release of Rank1 (Weller et al., 2025), various follow-up methods have been developed, focusing on injecting a reasoning process into listwise (Liu et al., 2026; Yang et al., 2025) and setwise (Zhuang et al., 2026) LLM rerankers. In Table 8, we compare Rank0 to such methods. For a fair comparison to the results reported in Liu et al. (2026), we rerank the top-100 passages from ReasonIR (Shao et al., 2025), a dense retriever trained for reasoning tasks.

How does Rank0 compare? *Surprisingly well.* Rank0 is only outperformed by ReasonRank, which leverages a sophisticated automated training data curation strategy to mine data from domains similar to those in BRIGHT. Compared to Rank-R1 and Rank-K, rerankers which were both trained on the MS MARCO dataset, Rank0 per-

forms 5.1 and 2.2 points better, on average. While not fully conclusive if reasoning is more helpful for non-pointwise rerankers, these results suggest that non-reasoning baselines remain strong and that it is imperative to properly evaluate them under equivalent settings. We leave this to future work.

6 Conclusion

In this paper, we set out to answer a simple question: *does reasoning actually improve pointwise reranking?* To answer this question, we evaluated Rank1 against two simple but effective pointwise reranking baselines: Rank0 and Rank1-NoReason. Through experiments across in-domain and out-of-domain datasets, we found that the reasoning process generally *harms* the accuracy of pointwise rerankers, with the effectiveness gap increasing as model size increased. Investigating the root cause of this result, we observed that the reasoning process hurts Rank1’s ability to distinguish degrees of relevance among passages it deems relevant.

Our findings suggest that, at least as currently implemented, the reasoning process is unnecessary for pointwise rerankers and that practitioners are better served with simpler methods like Rank0. We believe that in order to fully realize the benefits of reasoning, it is essential to redesign how reasoning is utilized by Rank1. However, we emphasize that any improvements in the training of Rank1 should also be properly compared against strong and simple baselines.

Limitations

First, while we demonstrated that reasoning hurts pointwise rerankers, the influence reasoning may have on other reranking approaches such as listwise (Sun et al., 2023) and setwise (Zhuang et al., 2024) remains an open question. However, as shown in Zhuang et al. (2024), pointwise rerankers are much more efficient as they can rerank candidate passages in parallel (Ma et al., 2024) and thus, our study covers a very popular and commonly used approach for passage reranking.

Second, our experiments were also limited to LLMs with ≤ 14 B parameters. While our results showed that increasing LLM size benefited Rank0 more than Rank1, the influence at scales larger than 14B remains an open question. We note, however, that Rank0 at the 7B and 14B scales still outperforms or is competitive with the reported results for Rank1-14B and Rank1-32B on BRIGHT.

Lastly, while we demonstrate methods to enhance Rank1—i.e., self-consistency, inverse training, and Rank1 (Non-Binary Labels)—these methods are still less effective than our Rank0. We hope our results and analysis can help future work in developing new reasoning pointwise rerankers.

Ethical Considerations

Our research solely uses publicly available datasets, and no personal information is collected. All datasets and models are used in accordance with their intended use and licenses. The goal of our study is to better understand factors which influence the accuracy of LLM rerankers, which we hope has a positive impact on building better search engines and other applications built on retrieval systems. We recognize, however, that such systems still rely on LLMs and may inherit their potential to produce biased, harmful, or offensive outputs.

Acknowledgments

Research was sponsored by the Department of the Air Force Artificial Intelligence Accelerator and was accomplished under Cooperative Agreement Number FA8750-19-2-1000. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Department of the Air Force or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein. Nour was also supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada. Additional funding was provided by Snowflake and 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

- Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. MS MARCO: A Human Generated MACHine Reading COMprehension Dataset. *arXiv:1611.09268*.
- Elias Bassani. 2022. Ranx: A Blazing-Fast Python Library for Ranking Evaluation and Comparison. In *ECIR (2)*, Lecture Notes in Computer Science, pages 259–264.
- Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Fernando Campos, and Jimmy Lin. 2021a. Overview of the TREC 2021 Deep Learning Track. In *Text Retrieval Conference*.
- Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Fernando Campos, Jimmy Lin, Ellen M. Voorhees, and Ian Soboroff. 2022. Overview of the TREC 2022 Deep Learning Track. In *Text Retrieval Conference*.
- Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Fernando Campos, and Ellen M. Voorhees. 2020. Overview of the TREC 2019 Deep Learning Track. *arXiv:2003.07820*.
- Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Fernando Campos, and Ellen M. Voorhees. 2021b. Overview of the TREC 2020 Deep Learning Track. *arXiv:2102.07662*.
- Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Hossein A. Rahmani, Daniel Campos, Jimmy Lin, Ellen M. Voorhees, and Ian Soboroff. 2023. Overview of the TREC 2023 Deep Learning Track. In *Text Retrieval Conference*.
- Fernando Ferraretto, Thiago Laitz, Roberto Lotufo, and Rodrigo Nogueira. 2023. ExaRanker: Synthetic Explanations Improve Neural Rankers. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2409–2414.
- Daya Guo, Dejian 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*.
- Edward 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*.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pages 611–626.
- 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 International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2356–2362.
- Wenhan Liu, Xinyu Ma, Weiwei Sun, Yutao Zhu, Yuchen Li, Dawei Yin, and Zhicheng Dou. 2026. ReasonRank: Empowering Passage Ranking with

- Strong Reasoning Ability. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 22007–22031.
- Wenjie Ma, Jingxuan He, Charlie Snell, Tyler Griggs, Sewon Min, and Matei Zaharia. 2025. Reasoning Models Can Be Effective Without Thinking. *arXiv:2504.09858*.
- 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*, pages 2421–2425.
- Sean MacAvaney, Nicola Tonellotto, and Craig Macdonald. 2022. Adaptive Re-Ranking with a Corpus Graph. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 1491–1500.
- Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document Ranking with a Pre-trained Sequence-to-Sequence Model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 708–718.
- Ronak Pradeep, Sahel Sharifmoghaddam, and Jimmy Lin. 2023. RankZephyr: Effective and Robust Zero-Shot Listwise Reranking is a Breeze! *arXiv:2312.02724*.
- Chris Samarinas and Hamed Zamani. 2025. Distillation and Refinement of Reasoning in Small Language Models for Document Re-ranking. In *Proceedings of the 2025 International ACM SIGIR Conference on Innovative Concepts and Theories in Information Retrieval (ICTIR)*, pages 430–435.
- 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*.
- Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. 2025. To CoT or not to CoT? Chain-of-Thought Helps Mainly on Math and Symbolic Reasoning. In *The Thirteenth International Conference on Learning Representations*.
- Hongjin Su, Howard Yen, Mengzhou Xia, Weijia Shi, Niklas Muennighoff, Han-yu Wang, Haisu Liu, 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*.
- Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang 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*, pages 14918–14937.
- Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *The Eleventh International Conference on Learning Representations*.
- Orion Weller, Kathryn Ricci, Eugene Yang, Andrew Yates, Dawn Lawrie, and Benjamin Van Durme. 2025. Rank1: Test-Time Compute for Reranking in Information Retrieval. In *Second Conference on Language Modeling*.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. HuggingFace’s Transformers: State-of-the-art Natural Language Processing. *arXiv:1910.03771*.
- An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2.5 Technical Report. *arXiv:2412.15115*.
- 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*.
- Honglei Zhuang, Zhen Qin, Rolf Jagerman, Kai Hui, Ji Ma, Jing Lu, Jianmo Ni, Xuanhui Wang, and Michael Bendersky. 2023. RankT5: Fine-Tuning T5 for Text Ranking with Ranking Losses. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2308–2313.
- 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*, pages 4419–4425.
- 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*, pages 38–47.

Dataset	# Queries
TREC DL19	43
TREC DL20	54
TREC DL21	53
TREC DL22	76
TREC DL23	82
Biology	103
Earth Science	116
Economics	103
Psychology	101
Robotics	101
StackOverflow	117
Sustainable Living	108
LeetCode	142
Pony	112
AoPS	111
TheoremQA Questions	194
TheoremQA Theorems	76

Table 9: Dataset details.

Model	Hugging Face ID	License
Qwen2.5-3B	Qwen/Qwen2.5-3B	Qwen Research
Qwen2.5-7B	Qwen/Qwen2.5-7B	Apache 2.0
Qwen2.5-14B	Qwen/Qwen2.5-14B	Apache 2.0
Qwen3-4B-Base	Qwen/Qwen3-4B-Base	Apache 2.0
Llama-3.1-8B	meta-llama/Llama-3.1-8B	Llama 3.1 Community
Mistral-7B-v0.2	mistralai/Mistral-7B-v0.2	Apache 2.0
rank1-3b	jhu-clsp/rank1-3b	MIT
rank1-7b	jhu-clsp/rank1-7b	MIT
rank1-14b	jhu-clsp/rank1-14b	MIT

Table 10: Model details and licenses.

A AI Usage

LLMs were solely utilized to assist in making grammatical edits for sections of this work, including the text and tables.

B Dataset and Model Details

We show the number of test queries for each dataset used for evaluation in Table 9. Model details and licenses can be found in Table 10. We also leverage Pyserini (Lin et al., 2021) and vLLM (Kwon et al., 2023) which are under the Apache 2.0 License.

C Training and Inference Details

To train Rank0 and Rank1 (our impl.) we fine-tune Qwen2.5 using LoRA (Hu et al., 2022) using the training hyperparameters described in Table 11. As noted in Section 2, when training Rank0 we leverage the same dataset as Rank1, but only use the (query, passage, R1’s relevance label) triples, ignoring R1’s reasoning process.

To train Rank1 w/ Inverse Training, we follow the same exact setup as Rank1 (our impl.), but switch around the order of R1’s relevance label and R1’s reasoning chain. In other words, we train with the following quadruples: (query, passage, R1’s relevance label, R1’s reasoning chain).

Hyperparameter	
Training Epochs	1
LoRA Rank	32
LoRA Alpha	64
Batch Size	128
Learning Rate	2×10^{-4}
LoRA Target Modules	All Linear Layers

Table 11: Training hyperparameters.

	First-Stage Retriever	
	BM25 + GPT-4 CoT	ReasonIR + GPT-4 CoT
Rank0 (7B)	31.0	31.2
Rank1 (our impl.) (7B)	27.8	25.8

Table 12: Impact of the first-stage retriever on the effectiveness of Rank0 versus Rank1 on BRIGHT.

Training for all rerankers took less than a day on an A100 GPU. Due to limited computational resources, each model is only trained once.

As the Rank0 and Rank1 outputs are using greedy decoding, all scores reported in our paper are from a single run. For Rank1 + Self-Consistency, we sample eight outputs from Rank1 using a temperature of 0.7. Due to limited computational resources, inference is run once per dataset.

In Figure 3, we report all prompts used across rerankers. Please refer to Weller et al. (2025) for the dataset-specific prompts. For Rank1 with inverse training, we leverage the same prompt as Rank0.

D Additional Results

Effect of initial retriever We explore how our results may be impacted when candidate documents are retrieved by different first-stage retrievers. The results for this experiment are in Table 12.

Effect of backbone LLM Lastly, we further test Rank0 and Rank1 with different LLMs. These results are found in Table 13.

E Details on Rank1 (Non-Binary Labels)

We explore a simple strategy to train Rank1 to produce graded, non-binary relevance labels, rather than binary labels. We refer to this Rank1 variation as Rank1 (Non-Binary Labels). To generate training data for Rank1 (Non-Binary Labels), for each example, we first prompt Qwen2.5-7B-Instruct to generate a score from 0 to 4 given the query, passage, and reasoning, using the “Non-Binary Label Prompt” in Figure 3.

This in turn generates graded relevance labels for each of Rank1’s training examples. We then

	MS MARCO
BM25	39.3
+ Qwen2.5-3B	
Rank0	61.5
Rank1 (our impl.)	57.8
+ Qwen3-4B	
Rank0	61.5
Rank1 (our impl.)	56.8
+ Llama3.1-8B	
Rank0	62.1
Rank1 (our impl.)	58.7
+ Mistral-7B-v0.2	
Rank0	61.8
Rank1 (our impl.)	56.1

Table 13: Investigating the generalizability of Rank0 and Rank1 across different LLM rerankers.

	BRIGHT
BM25 + GPT-4 CoT	27.0
+ Qwen2.5-7B	
Rank0	31.0
Hybrid (w/ BM25 + GPT-4 CoT)	34.2
Rank1 (our impl.)	27.8
Hybrid (w/ BM25 + GPT-4 CoT)	30.0
Rank1 (Non-Binary Labels)	24.8
Hybrid (w/ BM25 + GPT-4 CoT)	33.4
+ Qwen2.5-32B-Instruct	
QwenRerank (Shao et al., 2025)	34.9

Table 14: Training Rank1 to produce non-binary relevance labels.

train Rank1 exactly as described in Appendix C, using the graded labels generated by Qwen2.5-7B-Instruct rather than the original binary labels. Note that we *do not* adapt the reasoning chains provided in Rank1’s training data and instead leave them fixed. We, however, think this could be a good direction for future work.

The results for this experiment can be found in Table 14. The results show that Rank1 (Non-Binary Labels) is less effective than Rank1 when directly ranking with the generated non-binary relevance labels, likely due to there being many ties with passages being assigned the exact same label. To break ties, we investigate the effectiveness of the hybrid scoring approach utilized in Shao et al. (2025), which combines scores from BM25 + GPT-4 CoT with the generated relevance labels from Rank1 (Non-Binary Labels) with an interpolation weight of 0.5. Using the hybrid scoring strategy, Rank1 (Non-Binary Labels) improves nDCG@10 by 8.6 points and is even more effective than Rank1 under the same hybrid scoring strategy. Rank0 remains more effective than Rank1 (Non-Binary Labels) under equal conditions (34.2 vs. 33.4).

Rank0 prompt <pre> < im_start >system Determine if the following passage is relevant to the query. Answer only with 'true' or 'false'. < im_end > < im_start >user Query: {} Passage: {} < im_end > < im_start >assistant </pre>
Rank1 (our impl.) prompt <pre> < im_start >system Determine if the following passage is relevant to the query. Answer only with 'true' or 'false'. < im_end > < im_start >user Query: {} Passage: {} < im_end > < im_start >assistant <think> </pre>
Rank1-NoReason prompt <pre> < im_start >system Determine if the following passage is relevant to the query. Answer only with 'true' or 'false'. < im_end > < im_start >user Query: {} Passage: {} < im_end > < im_start >assistant <think> Okay, I have finished thinking. </think> </pre>
Rank1 (Non-Binary Labels) training data generation prompt <pre> < im_start >user Is the document below relevant to answering the query below? The answer should be 'Relevance score: X.' where X is a number from 0-4, with each score defined below: 0 - Not Relevant: The document does not address the query at all. 1 - Marginally Relevant: The document touches on the topic very lightly but provides almost no useful information. 2 - Partially Relevant: The document addresses some aspects of the query but is incomplete or superficial. 3 - Mostly Relevant: The document addresses the query well but may miss minor points or lack full detail. 4 - Highly Relevant: The document thoroughly addresses the query, providing detailed and accurate information. Here is the query: <start_query> {} <end_query> Here is the document: <start_document> {} <end_document> Here is your reasoning about the relevance of the document to the query: <start_reasoning> {} <end_reasoning> < im_end > < im_start >assistant Relevance score: </pre>

Figure 3: Prompt templates used in our experiments. The first three panels show the prompts for Rank0, Rank1, and Rank1-NoReason, respectively. The final panel shows the prompt used to generate graded relevance labels for Rank1 (non-binary labels) training.