DeAR: Dual-Stage Document Reranking with Reasoning Agents via LLM Distillation

Abdelrahman Abdallah, Jamshid Mozafari, Bhawna Piryani, Adam Jatowt
University of Innsbruck
{abdelrahman.abdallah, jamshid.mozafari, bhawna.piryani, adam.jatowt}@uibk.ac.at

Abstract

Large Language Models (LLMs) have transformed listwise document reranking by enabling global reasoning over candidate sets, yet single models often struggle to balance fine-grained relevance scoring with holistic cross-document analysis. We propose **De**ep**A**gent**R**ank (**DEAR**), an open-source framework that decouples these tasks through a dual-stage approach, achieving superior accuracy and interpretability. In *Stage 1*, we distill token-level relevance signals from a frozen 13B LLaMA teacher into a compact {3, 8}B student model using a hybrid of cross-entropy, RankNet, and KL divergence losses, ensuring robust pointwise scoring. In *Stage 2*, we attach a second LoRA adapter and fine-tune on 20K GPT-4o-generated chain-of-thought permutations, enabling listwise reasoning with natural-language justifications. Evaluated on TREC-DL19/20, eight BEIR datasets, and NovelEval-2306, DEAR surpasses open-source baselines by +5.1 nDCG@5 on DL20 and achieves 90.97 nDCG@10 on NovelEval, outperforming GPT-4 by +3.09. Without fine-tuning on Wikipedia, DeAR also excels in open-domain QA, achieving 54.29 Top-1 accuracy on Natural Questions, surpassing baselines like MonoT5, UPR, and RankGPT. Ablations confirm that dual-loss distillation ensures stable calibration, making DEAR a highly effective and interpretable solution for modern reranking systems.[1].

1 Introduction

Document reranking refines top candidate documents retrieved by a first-stage system to improve relevance to a user query. It plays a critical role in tasks like web search (Bajaj et al., 2016; Abdallah et al., 2025e), open-domain QA (Chen et al., 2017; Gruber et al., 2024), fact verification (Thorne et al., 2018), and Retrieval-Augmented Generation (RAG) (Lewis et al., 2020), where ranking quality directly impacts downstream results. Transformer-based models (e.g., BERT (Devlin et al., 2019), T5 (Raffel et al., 2020)) and instruction-tuned LLMs (e.g., InstructGPT (Ouyang et al., 2022), LLaMA (Touvron et al., 2023), GPT-4 (Achiam et al., 2023)) have driven reranking progress. While pointwise (Sachan et al., 2022; Abdallah et al., 2025b,f) and pairwise (Qin et al., 2023) approaches dominate earlier work, listwise reranking with LLMs (Sun et al., 2023; Pradeep et al., 2023a) offers global ranking benefits. However, these often rely on expensive proprietary APIs and suffer from context-length limitations and brittle reasoning. As shown in Figure 1, our proposed DEAR-L achieves superior nDCG@5 performance across diverse datasets, outperforming baselines like RankZephyr and RankGPT.

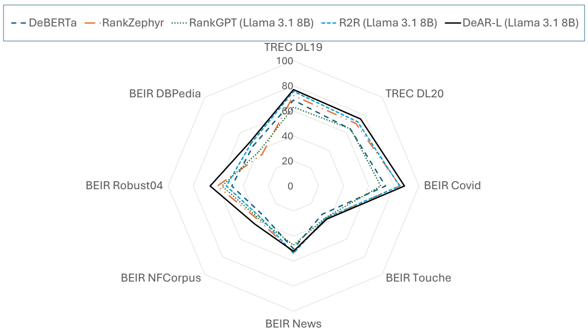

Figure 1: Radar chart comparing nDCG@5 performance of top reranking methods, including DeBERTa, RankZephyr, RankGPT (LLaMA 3.1 8B), R2R (LLaMA 3.1 8B), and DEAR-L (LLaMA 3.1 8B), across TREC DL19 and BEIR datasets (Covid, NFCorpus, Touche, DBPedia, News, Robust04).

To address these challenges, open-source efforts have explored knowledge distillation (KD) to transfer ranking abilities from large LLMs to smaller, more efficient student models (Sun et al., 2023; Pradeep et al., 2023a). However, existing

[1]Dataset and code available at https://github.com/DataScienceUIBK/DeAR-Reranking.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 5710–5723
November 4-9, 2025 ©2025 Association for Computational Linguistics

Synthetic training sample

System Prompt: You are RankLLM, an intelligent assistant that can rank passages based on their relevancy to the query.

Instruction: I will provide you with 3 passages, each indicated by a numerical identifier []. Rank the passages based on their relevance to the search query: *what is a shape of art*.
[1] In the visual arts, shape is a flat, enclosed area of an artwork created through line, texture, colour or an area enclosed by other shapes. [...]
[2] A shape in art is a closed line that is limited to two directions: width and length. [...]
[3] Form is a three-dimensional geometrical figure (i.e., sphere, cube, cylinder, etc.), as opposed to a shape, which is two-dimensional. [...]
Search Query: what is a shape of art
Steps to follow:

1. List the information requirements to answer the query.
2. For each requirement, find the passages that include the relevant information.
3. Rank the passages in descending order of relevance using only the identifiers (e.g., [2] > [1]).

The format of the final output should be ' ### Final Reranking: [] > []', " "e.g., ### Final Reranking: [2] > [1].

Final Reranking: [1] > [2] > [3]

Figure 2: Illustration of RankLLM training components. Synthetic training example with reasoning and final reranking. Passages and reasoning steps are abbreviated for brevity.

methods commonly depend on synthetic listwise permutations that risk propagating teacher errors such as hallucinations or misrankings. This motivates two key research questions: (1) *Can we balance KL divergence with ranking loss to effectively distill logit-level signals from the teacher while mitigating noise?* (2) *Can we incorporate synthetic reasoning chains to retain listwise reasoning benefits without overloading model context windows?*

We address these questions with a central insight: reranking performance can be enhanced by guiding the student with reasoning chains of thought (CoT) derived from synthetic data. We introduce **DEAR**, a novel dual-stage reranking framework that combines pointwise and listwise learning with reasoning-augmented supervision. Built on a frozen LLM backbone (e.g., LLaMA-13B) with lightweight LoRA adapters (Hu et al., 2022), DEAR trains on 20,000 synthetic reasoning examples and achieves performance comparable to GPT-4o, while surpassing RankZephyr in inference efficiency (see Section 5.3).

Our contributions are as follows:

- We propose **DEAR**, a dual-stage reranking framework that integrates pointwise cross-entropy learning with reasoning-augmented listwise ranking.
- We introduce a teacher–student pipeline where a LLM-based teacher transfers relevance signals to a student via logit-level distillation, combining cross-entropy, RankNet, and KL divergence losses.
- We construct $20K$ synthetic ranking examples with CoT reasoning.
- Extensive experiments on DL19, DL20, and BEIR-6 show that DEAR matches or outperforms larger baselines (e.g., GPT-4o, RankZephyr), improving NDCG@5 by up to +5.1 on DL20 while remaining lightweight and fast.

2 Related Work

Reranking methods (Abdallah et al., 2025c) are typically pointwise, pairwise, or listwise. Pointwise models like monoBERT (Nogueira et al., 2020) score each document independently using pretrained transformers (Devlin et al., 2019), but ignore inter-document context (Sachan et al., 2022). Pairwise approaches (e.g., duoBERT, duoT5 (Pradeep et al., 2021)) compare document pairs to infer preferences, at the cost of efficiency (Qin et al., 2023). Listwise methods, powered by LLMs like GPT-3.5 and GPT-4, enable zero-shot ranking through prompting (Sun et al., 2023), though reliance on proprietary APIs limits reproducibility. Open-source variants like RankVicuna and RankZephyr (Pradeep et al., 2023b) address this via distilled listwise models,

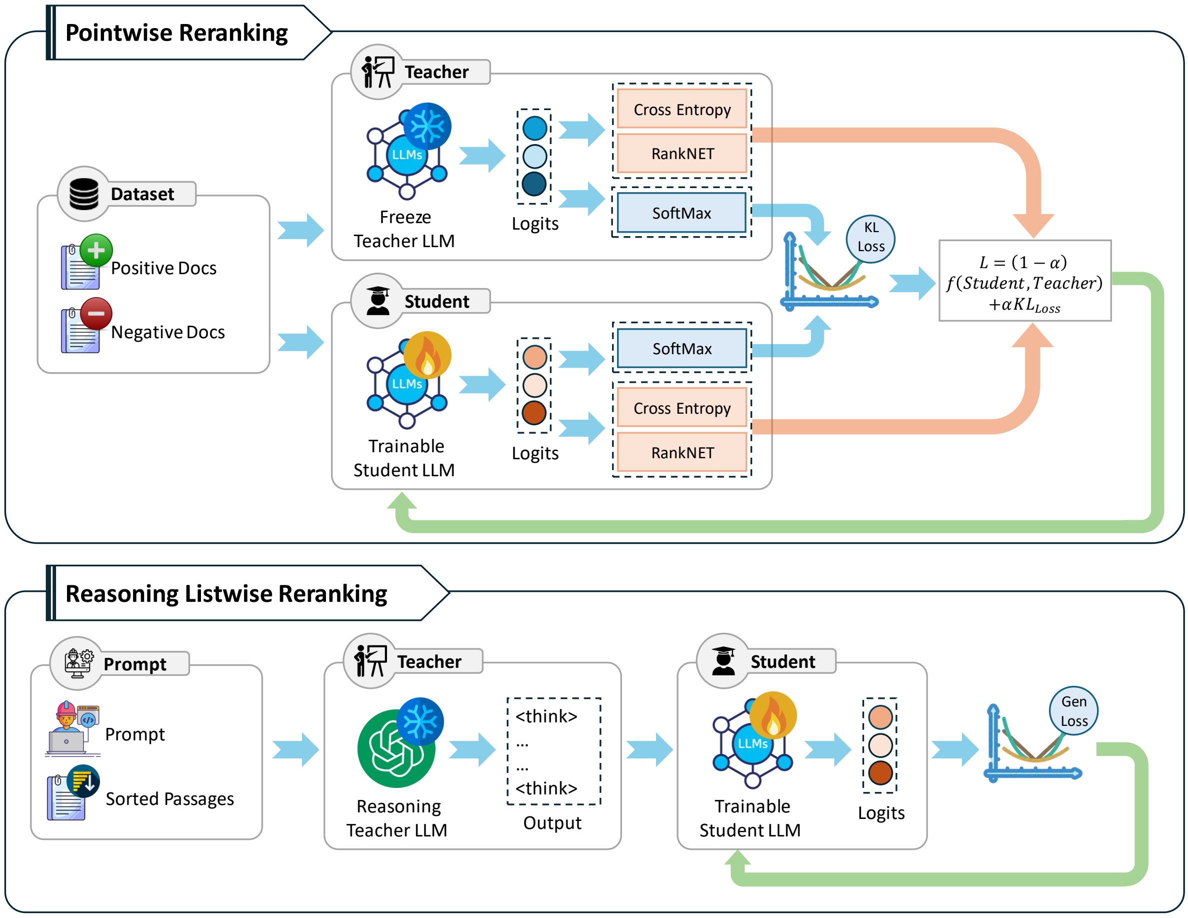

Figure 3: Overview of the **DEAR** dual-stage training pipeline. **Top:** In the pointwise stage, a frozen teacher LLM generates relevance logits for positive/negative documents, which are distilled into a student model using cross-entropy, RankNet, and KL divergence losses. **Bottom:** In the listwise stage, a reasoning teacher produces step-by-step chain-of-thought explanations and ranked outputs over candidate sets. The student is trained to generate coherent reasoning and rankings via generation loss.

but remain sensitive to hallucinations and prompt ordering.

Knowledge distillation (KD) compresses large models into smaller ones by transferring logits or intermediate signals (Hinton et al., 2015). In IR, it has been used for both retrieval (Guo et al., 2021) and reranking (Wang and Yoon, 2021). RankGPT and RankZephyr apply permutation-based KD using teacher rankings, but such discrete labels may discard confidence information.

Other work explores reasoning distillation (Magister et al., 2023; Fu et al., 2023), where small models benefit from teacher-generated explanations. Our method extends this by combining logit-level KD (Stage 1) with reasoning-based listwise training (Stage 2), preserving fine-grained supervision while improving stability. Reasoning improves reranking accuracy and interpretability. R2R (Ji et al., 2024) distills direct and comparative explanations from GPT-4 for MSMARCO and BEIR. RankGPT also leverages implicit LLM reasoning via prompt completions.

3 Method

3.1 Preliminaries

Task Definition. Given a query $q \in \mathcal{Q}$ and a corpus $\mathcal{C} = \{d_1, \ldots, d_n\}$, the reranking task aims to reorder a top-k candidate set ($k \ll n$), initially retrieved by a first-stage bi-encoder (Karpukhin et al., 2020), to maximize relevance to q. The reranker refines these candidates using a more expressive model to optimize metrics like nDCG (Wang et al., 2013; Järvelin and Kekäläinen, 2002).

Pointwise vs. Listwise Reranking. Pointwise rerankers score each (q, d) pair independently via $f(q, d)$ (Sachan et al., 2022), offering efficiency but no inter-document reasoning. We improve this via knowledge distillation from LLMs to LoRA-equipped student models (Hu et al., 2022). Listwise methods (e.g., ListNet (Cao et al., 2007), LambdaLoss (Burges, 2010)) model cross-document interactions (Ma et al., 2023; Pradeep et al., 2023b), but suffer from input order sensitivity and transformer context limits (Sun et al., 2023).

3.2 Pointwise Reranking with KL Distillation

The first stage of DEAR employs a pointwise reranking approach, leveraging KD to transfer the teacher model's nuanced relevance judgments to a compact student model. This stage produces calibrated relevance scores for individual documents, forming the foundation for the subsequent listwise refinement. For a query q_i and document d_{ij} from the candidate set $D_i = \{d_{i1}, \ldots, d_{im}\}$, the input is formatted as:

$$s_{ij} = [\text{query: } q_i, \text{ document: } d_{ij}, \texttt{</s>}], \tag{1}$$

where </s> denotes the end-of-sequence token. Both the student model $f_s(\cdot; \theta_s)$ and teacher model $f_t(\cdot; \theta_t)$, implemented as transformer-based sequence classifiers, process s_{ij} through L layers of self-attention and feed-forward networks. Let $\mathbf{H}^l \in \mathbb{R}^{l \times d}$ represent the hidden states at layer l for sequence length l and hidden dimension d. The final hidden state corresponding to </s>, denoted $\mathbf{h}^L_{\texttt{</s>}} \in \mathbb{R}^d$, is projected to a scalar relevance score:

$$\hat{y}^s_{ij} = \mathbf{w}_s^\top \mathbf{h}^L_{\texttt{</s>}}(s_{ij}; \theta_s) + b_s, \tag{2}$$

$$\hat{y}^t_{ij} = \mathbf{w}_t^\top \mathbf{h}^L_{\texttt{</s>}}(s_{ij}; \theta_t) + b_t, \tag{3}$$

where $\mathbf{w}_s, \mathbf{w}_t \in \mathbb{R}^d$ and $b_s, b_t \in \mathbb{R}$ are learnable projection parameters. For a batch of B queries, each associated with m documents, the student and teacher produce score matrices $\mathbf{S}^s, \mathbf{S}^t \in \mathbb{R}^{B \times m}$, where $\mathbf{S}^s_{bj} = \hat{y}^s_{ij}$ and $\mathbf{S}^t_{bj} = \hat{y}^t_{ij}$ for query q_i (batch index b) and document d_{ij} (index j). The training objective combines a ranking loss with a KD loss, parameterized by a weighting factor $\alpha \in [0, 1]$.

We support multiple ranking loss functions to accommodate different supervision signals: Softmax Cross-Entropy (PointCE): Treating reranking as a classification task, we assign binary labels $\mathbf{y}_i = (y_{i1}, \ldots, y_{im})$, where $y_{ij} = 1$ for the most relevant document and $y_{ij} = 0$ otherwise. The loss is:

$$\begin{aligned}\mathcal{L}_{\text{PointCE}}(\mathbf{y}_i, \mathbf{S}^s_i) = &-\sum_{j \mid y_{ij}=1} \log(\sigma(\mathbf{S}^s_{ij})) \\ &-\sum_{j \mid y_{ij}=0} \log(1 - \sigma(\mathbf{S}^s_{ij}))\end{aligned} \tag{4}$$

where $\sigma(x) = (1 + e^{-x})^{-1}$ is the logistic function (Nogueira et al., 2020).

RankNet Loss: To model pairwise preferences, we use the RankNet loss (Burges et al., 2005). Given relevance ranks r_{ij} derived from $\mathbf{y}_i$ (lower r_{ij} indicates higher relevance), the loss is:

$$\begin{aligned}\mathcal{L}_{\text{RankNet}}(\mathbf{y}_i, \mathbf{S}^s_i) = &\sum_{j=1}^{m} \sum_{j'=1}^{m} \mathbb{1}_{r_{ij} < r_{ij'}} \\ &\cdot \log(1 + \exp(\mathbf{S}^s_{ij'} - \mathbf{S}^s_{ij}))\end{aligned} \tag{5}$$

Knowledge Distillation Loss (KD) aligns the student's softened score distribution with the teacher's using KL divergence. For query q_i, scores $\mathbf{S}^s_i, \mathbf{S}^t_i \in \mathbb{R}^m$ are normalized with temperature τ:

$$\mathbf{P}^s_i = \text{softmax}(\mathbf{S}^s_i / \tau), \quad \mathbf{P}^t_i = \text{softmax}(\mathbf{S}^t_i / \tau) \tag{6}$$

The KD loss is:

$$\begin{aligned}\mathcal{L}_{\text{KD}}(\mathbf{S}^s_i, \mathbf{S}^t_i) &= \tau^2 \cdot \text{KL}(\mathbf{P}^s_i || \mathbf{P}^t_i) \\ &= \tau^2 \sum_{j=1}^{m} \mathbf{P}^s_{ij} \log\left(\frac{\mathbf{P}^s_{ij}}{\mathbf{P}^t_{ij}}\right)\end{aligned} \tag{7}$$

The total loss combines the ranking loss and KD loss:

$$\begin{aligned}\mathcal{L}_{\text{total}}(\mathbf{y}_i, \mathbf{S}^s_i, \mathbf{S}^t_i) = &(1 - \alpha) \cdot \mathcal{L}_{\text{rank}}(\mathbf{y}_i, \mathbf{S}^s_i) \\ &+ \alpha \cdot \mathcal{L}_{\text{KD}}(\mathbf{S}^s_i, \mathbf{S}^t_i)\end{aligned} \tag{8}$$

where $\mathcal{L}_{\text{rank}}$ is either $\mathcal{L}_{\text{PointCE}}$ or $\mathcal{L}_{\text{RankNet}}$, and $\alpha \in [0, 1]$ balances the losses. Figure 3 illustrates the pointwise reranking stage of DEAR, where the student model learns from the frozen teacher using a combination of cross-entropy, RankNet, and KL-divergence losses. We train two student models: one with $\mathcal{L}_{\text{PointCE}} + \mathcal{L}_{\text{KD}}$ and another with $\mathcal{L}_{\text{RankNet}} + \mathcal{L}_{\text{KD}}$. The teacher provides logits without updates. LoRA ensures efficiency, producing a top-100 ranked list for the listwise stage.

Method	prev.	Top-K	DL19	DL20	Covid	NFCorpus	Touche	DBPedia	SciFact	Signal	News	Robust04	BEIR (Avg)
BM25	-	-	50.58	47.96	59.47	30.75	**44.22**	31.80	67.89	33.05	39.52	40.70	43.42
Supervised													
monoBERT (340M)	BM25	100	70.50	67.28	70.01	36.88	31.75	41.87	71.36	31.44	44.62	49.35	47.16
monoT5 (220M)	BM25	100	71.48	66.99	78.34	37.38	30.82	42.42	73.40	31.67	46.83	51.72	49.07
monoT5 (3B)	BM25	100	71.83	68.89	80.71	38.97	32.41	44.45	76.57	32.55	48.49	56.71	51.36
Cohere Rerank-v2	BM25	100	73.22	67.08	81.81	36.36	32.51	42.51	74.44	29.60	47.59	50.78	49.45
Unsupervised													
UPR (3B)	BM25	100	53.85	56.02	68.11	35.04	19.69	30.91	72.69	31.91	43.11	42.43	42.99
InPars (3B)	-	100	-	66.12	78.35	-	-	-	-	-	-	-	-
Promptagator++	-	100	-	-	76.2	37.0	38.1	43.4	73.1	-	-	-	-
RankGPT (llama 3.1 8B)	BM25	100	58.46	59.68	69.61	33.62	37.98	37.25	69.82	32.95	43.90		
RankGPT-3.5	BM25	100	65.80	62.91	76.67	35.62	36.18	44.47	70.43	32.12	48.85	50.62	49.37
RankGPT-4	RankGPT-3.5	30	75.59	70.56	85.51	38.47	38.57	47.12	74.95	34.40	52.89	57.55	53.68
DEAR-Pointwise (DEAR-P)													
Llama3.1-8B (RL)†	BM25	100	72.17	68.93	85.21	37.01	34.88	45.56	77.43	30.16	52.05	54.42	52.09
Llama3.1-8B (BC)‡	BM25	100	74.50	68.71	84.14	36.57	37.23	46.27	77.39	29.91	51.71	52.43	51.95
Llama3.1-3B (RL)§	BM25	100	72.94	69.21	83.01	36.30	35.76	45.96	74.45	28.64	50.84	49.78	50.59
Llama3.1-3B (BC)¶	BM25	100	74.49	69.02	82.91	35.78	36.17	45.28	75.48	29.14	48.99	50.93	50.58
DEAR-Listwise (DEAR-L)													
GPT-4	†	30	75.74	**72.18**	86.28	**40.56**	31.41	46.15	77.58	31.13	50.77	57.91	52.72
GPT-4	‡	30	75.68	72.73	86.12	40.42	31.60	45.99	78.36	32.40	52.10	62.18	53.65
GPT-4	§	30	74.72	72.21	85.13	40.30	33.95	46.17	78.04	31.79	53.28	60.25	53.61
GPT-4	¶	30	**76.29**	70.88	85.79	40.34	32.43	45.79	76.71	33.00	52.76	**60.39**	53.40
Llama3.1-8B	†	30	74.86	71.06	86.43	39.08	33.76	46.61	78.08	**33.10**	53.17	59.55	**53.72**
Llama3.1-8B	‡	30	75.54	70.39	86.53	38.48	34.32	46.20	**78.47**	31.69	**53.79**	59.43	53.61
Llama3.1-8B	§	30	75.29	71.17	88.10	39.05	31.47	**46.69**	77.77	32.64	53.09	57.93	53.35
Llama3.1-8B	¶	30	75.33	70.02	**88.36**	38.80	**35.04**	46.34	77.34	32.56	52.24	58.63	53.66

Table 1: nDCG@10 performance on TREC Deep Learning Tracks (DL19, DL20) and BEIR datasets (CovidQA, NFCorpus, Touche, DBPedia, SciFact, Signal, News, Robust04).

3.3 Listwise Reranking with Reasoning

The second stage of DEAR performs listwise reranking on the top-k candidates (e.g., $k = 30$) from the pointwise stage. This stage enhances the ranking by reasoning over the candidate set, guided by synthetic reasoning chains generated by GPT-4o, addressing the limitations of pointwise methods in capturing inter-document dependencies.

Dataset Construction. From MS MARCO (Bajaj et al., 2016), we sample 40K queries, as utilized by Pradeep et al. (2023b). For each query q_i, we retrieve the top-20 candidate passages $\{d_{i1}, d_{i2}, \ldots, d_{i20}\}$ using GPT-4 via RankZephyr (Pradeep et al., 2023b; Lin et al., 2021b). To generate teacher-labeled data, we employ GPT-4o with a CoT reasoning prompt, producing both a ranked list and corresponding reasoning for each query-passage set, as illustrated in Figure 2. This process yields $20K$ synthetic reasoning examples, each consisting of a query, candidate passages, CoT reasoning, and a teacher-generated ranking, which are used to train the student model. The prompt instructs the model to: (1) extract requirements, (2) match passages to them, and (3) rank documents using their IDs, as shown in Figure 7.

Training Objective. We train an instruction-tuned student model using supervised fine-tuning. For a query q_i and its candidate set D_i, the input is the prompt containing q_i and D_i, and the target output is the teacher-generated ranked list (e.g., `[1] > [2] > [3]`). Let $\pi_i = (\pi_{i1}, \pi_{i2}, \ldots, \pi_{ik})$ denote the target permutation, where $\pi_{ij} \in \{1, \ldots, k\}$ indicates the position of passage d_{ij} in the ranked list (e.g., $\pi_{i1} = 2$ means d_{i1} is ranked second). The student model generates a predicted permutation $\hat{\pi}_i$, which is optimized to align with the teacher's ranking through supervised fine-tuning on the synthetic dataset.

4 Experiments

We evaluate DEAR on standard reranking benchmarks and settings.

4.1 Implementation Details

Datasets and Evaluation Metrics. We evaluate our approach on TREC-DL (Bajaj et al., 2016), BEIR (Thakur et al., 2021), and open-domain QA tasks such as NQ (Kwiatkowski et al., 2019) and

Method	prev.	Top-K	nDCG@1	nDCG@5	nDCG@10	Avg
BM25	-	-	33.33	45.96	55.77	45.02
monoBERT (340M)	BM25	100	78.57	70.65	77.27	75.50
monoT5 (220M)	BM25	100	83.33	77.46	81.27	80.69
monoT5 (3B)	BM25	100	83.33	78.38	84.62	82.11
RankGPT-3.5	BM25	100	76.19	74.15	75.71	75.35
RankGPT-4	RankGPT-3.5	20	85.71	87.49	90.45	87.88
DEAR-Pointwise (DEAR-P)						
Llama3.1-8B (RL)†	BM25	100	85.71	75.59	82.34	81.21
Llama3.1-8B (BC)‡	BM25	100	88.10	79.48	85.03	84.20
Llama3.1-3B (RL)§	BM25	100	85.71	78.24	82.56	82.17
Llama3.1-3B (BC)¶	BM25	100	85.71	76.94	82.12	81.59
DEAR-Listwise (DEAR-L)						
Llama3.1-8B	†	30	92.86	88.04	92.01	90.97
Llama3.1-8B	‡	30	92.86	88.04	90.98	90.63
Llama3.1-8B	§	30	90.48	90.32	92.05	90.95
Llama3.1-8B	¶	30	90.48	88.79	90.62	89.96

Table 2: Reranking results on NovelEval-2306. We compare BM25, monoT5, GPT baselines, and DEAR (pointwise and listwise).

WebQA (Berant et al., 2013). For TREC, we use DL19 and DL20. From BEIR, we select eight diverse datasets: Covid, NFCorpus, Signal, News, Robust04, Touche, DBPedia, and SciFact. Following standard reranking pipelines (Nogueira et al., 2019a; Sun et al., 2023), we retrieve the top-100 candidate documents per query using BM25 via Pyserini and Rankify (Lin et al., 2021a; Abdallah et al., 2025d). We use NDCG@1, NDCG@5, and NDCG@10 to measure reranking performance, focusing on top-ranked relevance quality. We integrated our results with RankArena (Abdallah et al., 2025a) to show effective of our model compare with other reranker model. For open-domain QA, we also report top-1, top-10, and top-20 accuracy.

Models. We experiment with both Qwen and LLaMA model families. For the **pointwise stage**, we use: (1) **Qwen:** Student models include Qwen 1.7B, 1.5B, 3B, 4B, and 7B, with Qwen3 14B as the teacher (Bai et al., 2023). (2) **LLaMA:** Student models include LLaMA3.2 3B, 1B, and LLaMA3.1 8B, trained using LLaMA2 13B as teacher (Touvron et al., 2023). (3) For the **listwise stage**, we use a single student: LLaMA3.1 8B with reasoning supervision.

Training Infrastructure. All experiments are conducted on 4×V100 GPUs (32GB). The teacher models remain frozen during distillation. All models are fine-tuned using LoRA adapters. For the pointwise stage, we set `lora_alpha` = 64. For the listwise stage, we reduce `lora_alpha` to 8 to reflect reasoning efficiency. We use the Adam optimizer (Kingma, 2014) with a batch size of 8, and a learning rate of 2e-5 with linear decay, weight decay of 0.1, and train for 3 epochs.

4.2 Superior Performance

We evaluate DEAR 's dual-stage reranking on TREC DL19/20 and BEIR datasets, using nDCG@10. Table 1 presents full results, with key findings summarized below. **Strong Pointwise Performance.** In the pointwise stage, DEAR-P re-ranks Top-100 BM25 candidates. LLaMA3.1-8B with RankLoss (RL) achieves a BEIR average of 52.09, improving +8.67 over BM25 (43.42) and surpassing monoT5-3B (51.36). Binary Cross-Entropy (BCE) excels on DL19 (74.50) and Touche (37.23), while RL outperforms overall (52.09 vs. 51.95). The compact LLaMA3.1-3B RL scores 50.59, confirming scalability. **Enhanced with Listwise CoT Reranking.** The listwise stage refines the Top-20 pointwise outputs using CoT reasoning. DEAR-L (LLaMA3.1-8B RL) reaches 53.72 BEIR average, outperforming RankGPT-4 (53.68) on Covid (88.36 vs. 85.51), NFCorpus (40.56 vs. 38.47), and Robust04 (62.18 vs. 57.55), leveraging efficient open-source models. **Benefits of CoT Reasoning.** CoT enhances inter-document understanding, with LLaMA3.1-8B RL (53.72) surpassing GPT-4-BCE (53.65). It boosts weaker pointwise models, e.g., LLaMA3.1-3B-BCE improves from 50.58 to 53.61, closing performance gaps without increasing model size. **Scalability and Efficiency.** Both 8B and 3B models excel in listwise reranking. LLaMA3.1-3B RL achieves 88.10 on Covid and 53.35 BEIR average, nearing 8B performance. RL consistently outperforms BCE (53.77 vs. 53.67), optimizing permutations effectively.

4.3 NovelEval Reranking Result

NovelEval-2306 tests DEAR on novel queries to ensure generalisation. In pointwise reranking, DEAR-P (LLaMA3.1-8B BCE) scores 84.20 average across nDCG@1/5/10, outperforming monoT5-3B (82.11) and RankGPT-3.5 (75.35), and closely trailing RankGPT-4 (87.88). The 3B RL model achieves 82.17, underscoring DEAR 's efficiency with smaller models. Listwise reranking with CoT, applied to the Top-20 pointwise outputs, significantly enhances performance: DEAR-L (LLaMA3.1-8B RL) reaches 90.97 average, surpassing RankGPT-4 by +3.09, with nDCG@10 of 92.01. RL aids listwise optimization, as seen in the consistent gains over BCE (90.97 vs. 89.96).

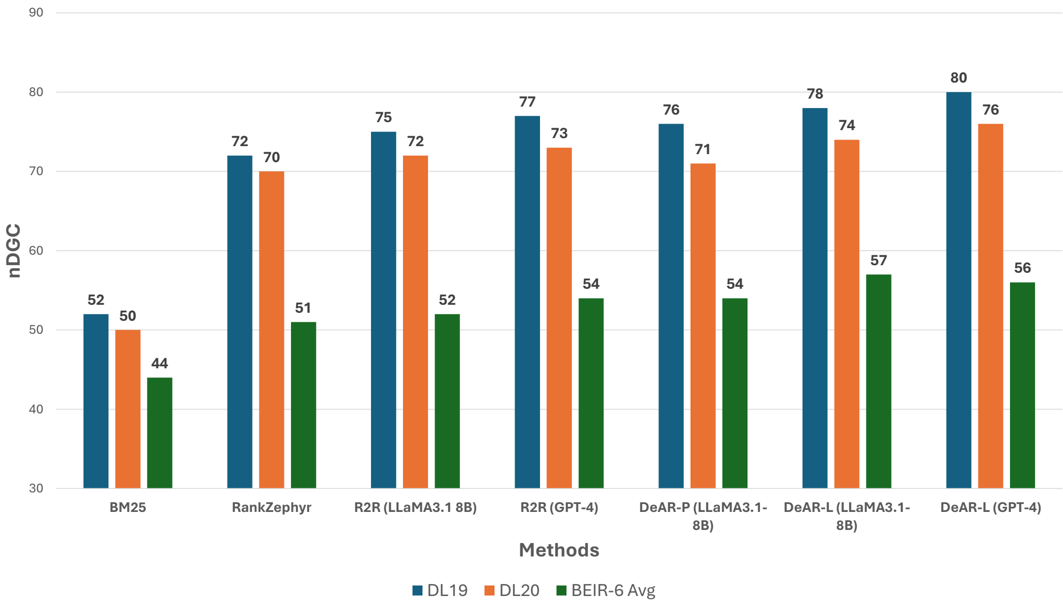

Figure 4: nDCG@5 performance of DEAR-Listwise vs. Reason-to-Rank (R2R) on TREC DL19/20 and BEIR datasets. See Appendix 4.5 and Table 6 for full comparison.

Method	@1	@5	@10	Avg
BM25	54.26	52.78	50.58	52.54
RankGPT (text-davinci-003)	70.54	61.90	57.24	63.23
RankGPT (gpt-3.5-turbo)	75.58	66.19	60.89	67.55
RankGPT (gpt-4)	79.46	71.65	65.68	72.26
RankGPT (rerank-english-v2.0)	79.46	71.56	64.78	71.27
RankGPT (claude-2)	66.66	59.33	55.91	60.63
RankGPT (claude-instant-1)	81.01	66.71	62.23	69.98
RankGPT (text-bison-001)	69.77	64.46	58.67	64.30
RankGPT (bard-2023.10.21)	81.01	65.57	60.11	68.90
RankGPT (flan-t5-xxl)	52.71	51.63	50.26	51.53
RankGPT (ChatGLM-6B)	54.26	52.77	50.58	52.54
RankGPT (Vicuna-13B)	54.26	51.55	49.08	51.63
DEAR-Pointwise (DEAR-P)				
Llama3.1-8B (RL)	80.23	69.86	64.16	71.42
Llama3.1-8B (BC)	79.46	71.45	65.48	72.13
Llama3.1-3B (RL)	78.68	70.19	64.26	71.04
Llama3.1-3B (BC)	83.33	72.30	65.43	73.69
DEAR-Listwise (DEAR-L)				
Llama3.1-8B	77.91	72.32	66.27	72.17
Llama3.1-8B	77.13	71.70	66.05	71.63
Llama3.1-8B	77.13	72.00	65.66	71.60
Llama3.1-8B	81.00	72.98	66.56	73.51

Table 3: Results of NDGC for different LLMs on re-ranking top-20 passages on DL-19.

Table 2 details these results, confirming DEAR 's strong generalization on unseen queries using open-source models, avoiding reliance on proprietary APIs.

4.4 Performance on TREC DL-19

On TREC DL-19, a standard IR benchmark, DEAR competes with proprietary LLM APIs like GPT-4 and Claude. In the pointwise stage, DEAR-P (LLaMA3.1-3B BCE) re-ranks Top-100 BM25 candidates, achieving 73.69 average (nDCG@1/5/10), outperforming RankGPT-4 (72.26) and other APIs like Claude-2 (60.63) and Bard (68.90). LLaMA3.1-8B RL scores 71.42, remaining competitive. The listwise stage, re-ranking the Top-20 pointwise outputs with CoT, further improves performance: DEAR-L (LLaMA3.1-8B) reaches 73.51, exceeding RankGPT-4 and demonstrating superior refinement. Table 3 presents these results, validating DEAR 's state-of-the-art performance with compact, open-source models in real-world IR settings.

4.5 Comparison with Reasoning Methods

We compare DEAR 's CoT reasoning with Reason-to-Rank (R2R) (Ji et al., 2024), which employs direct relevance and comparison reasoning, using nDCG@5 on TREC DL19/20 and BEIR datasets. In the pointwise stage, DEAR-P (LLaMA3.1-8B RL) provides a strong foundation, e.g., 89.13 on Covid. Listwise reranking with CoT, applied to the Top-20 pointwise outputs, significantly outperforms R2R: DEAR-L (LLaMA3.1-8B BCE) achieves 80.71 on

Teacher	Student	dl19	dl20	Avg BEIR
Qwen3-14B	Qwen3-1.7B	73.31	68.00	49.58
Qwen3-14B	Qwen3-4B	74.04	66.94	50.59
Qwen2.5-14B	Qwen2.5-7B	74.06	66.15	51.24
Qwen2.5-14B	Qwen1.5-1.5B	73.63	65.85	50.00
Qwen2.5-14B	Qwen1.5-3B	73.47	66.42	50.20
LLaMA2-13B	LLaMA-3.2-3B	74.49	69.02	50.58
LLaMA2-13B	LLaMA-3.2-1B	72.82	68.08	49.72
LLaMA2-13B	LLaMA-3.1-8B	74.50	68.71	51.95

Table 4: nDCG@10 performance (in percentage) of DEAR-Pointwise with different teacher and student model pairs using binary cross-entropy across TREC and BEIR datasets. See Appendix C and Table 7 for full comparison.

DL19, surpassing R2R-GPT-4 (77.7) and R2R-LLaMA3.1-8B (75.4). On BEIR, DEAR-L scores 91.94 on Covid and 69.49 on Robust04, outpacing R2R-GPT-4 (85.3 and 58.6). The 3B RL model reaches 91.94 on Covid, rivaling R2R's larger models. On NFCorpus, DEAR 's 45.76 exceeds the result by R2R-LLaMA3.1-8B (36.4). Figure 4 visualizes these gains, emphasizing DEAR 's robust multi-document reasoning with open-source efficiency, enhancing pointwise outputs through CoT-guided global ranking decisions.

5 Additional Analysis

5.1 Impact of Teacher–Student

We assess DEAR-P's pointwise reranking with various teacher-student pairs. LLaMA2-13B to LLaMA3.1-8B yields the highest BEIR average (51.95), while Qwen2.5-14B to Qwen2.5-7B scores 51.24. Smaller students, like LLaMA-3.2-1B (49.72) and Qwen3-1.7B (49.58), remain competitive, showcasing DEAR 's flexibility. Listwise reranking, applied to the Top-20 pointwise outputs, was not evaluated here but is expected to further enhance these results, as seen in prior subsections. Table 4 details the pointwise performance, highlighting DEAR 's adaptability across diverse model architectures and sizes for efficient knowledge distillation.

5.2 Alpha Selection for KL Divergence

We analyze now the impact of the alpha coefficient, which balances KL divergence and binary cross-entropy (BCE) during pointwise reranking in DEAR, using the LLaMA2-13B $\rightarrow$ LLaMA-3.1-8B teacher-student pair. Varying α from 0.1 to 0.5 on MS MARCO, we evaluate nDCG@10 across eight BEIR datasets. Figure 5 shows the average BEIR score peaking at 52.5 with $\alpha = 0.1$,

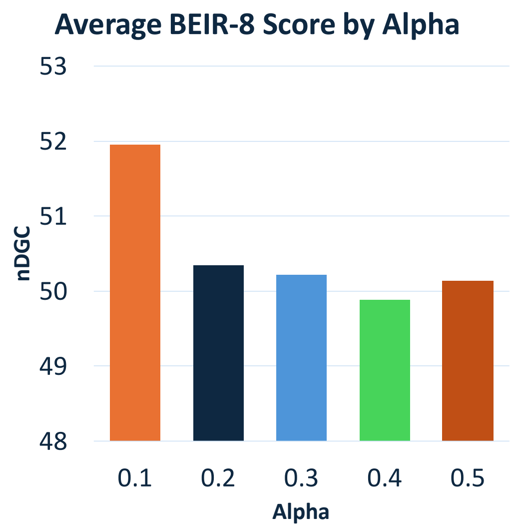

Figure 5: Average BEIR-8 score (nDCG@10, in percentage) across alpha values for DEAR-Pointwise (Teacher: LLaMA2-13B, Student: LLaMA-3.1-8B) with KL divergence and binary cross-entropy loss.

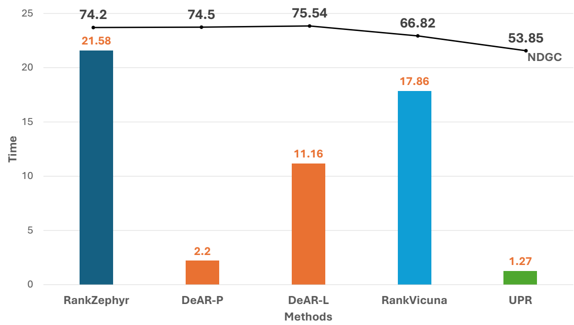

Figure 6: nDCG@10 performance vs. inference time (seconds) Per Query for DEAR-P, DEAR-L, RankZephyr, RankVicuna, and UPR on TREC DL19.

then declining to 48.5 at $\alpha = 0.5$. Lower α prioritizes the BCE ranking objective, optimizing relevance, while higher α overemphasizes teacher logit alignment, reducing ranking quality. Thus, we adopt $\alpha = 0.1$ for all KL-based distillation in DEAR.

5.3 Inference Time vs. Performance

We next compare DEAR 's inference time and nDCG@10 on TREC DL19 (Figure 6). DEAR-P (LLaMA3.1-8B) achieves 74.5 nDCG in 2.2s, outperforming RankZephyr (74.2, 21.58s) and RankVicuna (66.82, 17.86s), with only UPR faster (1.27s, 53.85 nDCG). DEAR-L (LLaMA3.1-8B) reaches 75.54 nDCG in 11.16s, balancing speed and CoT-enhanced accuracy, surpassing slower baselines with open-source efficiency.

Reranking/	Model	NQ Top-1	NQ Top-10	NQ Top-50	WebQ Top-1	WebQ Top-10	WebQ Top-50
BM25	-	23.46	56.32	74.57	19.54	53.44	72.34
UPR (Sachan et al., 2022)	T0-3B	35.42	67.56	76.75	32.48	64.17	73.67
	gpt-neo-2.7B	28.75	64.81	76.56	24.75	59.64	72.63
RankGPT (Sun et al., 2023)	LLaMAv3.1-8b	41.55	66.17	75.42	38.77	62.69	73.12
RankT5 (Zhuang et al., 2023)	3b	47.17	70.85	76.89	40.40	66.58	74.45
Inranker (Laitz et al., 2024)	3b	15.90	48.06	69.00	14.46	46.11	69.34
MonoBert (Nogueira et al., 2019b)	large	39.05	67.89	76.56	34.99	64.56	73.96
Twolar (Baldelli et al., 2024)	twolar-xl	46.84	70.22	76.86	41.68	67.07	74.40
Echorank (Rashid et al., 2024)	flan-t5-xl	41.68	59.05	62.38	36.22	57.18	61.51
Incontext Reranker (Chen et al., 2024)	LLaMAv3.1-8b	15.15	57.11	76.48	18.89	52.16	71.70
Lit5 (Tamber et al., 2023)	LiT5-Distill-xl-v2	47.92	69.03	76.17	41.53	65.69	73.27
Sentence Transformer	GTR-xxl	42.93	68.55	77.00	39.41	65.89	74.01
	T5-xxl	38.89	67.78	76.64	35.82	65.20	74.01
DEAR-P	Llama3.1 8B	48.92	73.35	78.78	41.93	67.67	75.05
DEAR-L	Llama3.1 8B	54.29	73.07	78.78	46.60	68.11	75.05

Table 5: Performance of re-ranking methods on BM25-retrieved documents for NQ Test and WebQ Test. Results are reported in terms of Top-1, Top-10, and Top-50 accuracy. Note that some results (e.g., UPR) differ from original papers due to re-ranking top-100 documents instead of 1,000.

5.4 Open Domain QA

We finally evaluate DEAR 's generalization to Natural Questions (NQ) and Web Questions (WebQ) without Wikipedia fine-tuning, re-ranking Top-100 BM25 passages (Table 5). DEAR-P (LLaMA3.1-8B, RankLoss) scores 48.92 Top-1 on NQ and 41.93 on WebQ, outperforming RankGPT (41.55, 38.77) and Twolar (46.84, 41.68). DEAR-L with CoT boosts Top-1 to 54.29 (NQ) and 46.60 (WebQ), surpassing RankT5-3B (47.17, 40.40). Top-10 (73.07 NQ, 68.11 WebQ) and Top-50 (78.78 NQ, 75.05 WebQ) gains are smaller, but CoT enhances precision under domain shift, excelling with open-source models trained on MS MARCO.

6 Conclusion

DeAR introduces a dual-stage reranking framework that decouples pointwise scoring and listwise reasoning, achieving high accuracy and interpretability. Stage 1 distills relevance signals from a 13B LLaMA teacher into 3, 8B students using hybrid losses, ensuring robust calibration. Stage 2 fine-tunes with GPT-4o-generated CoT permutations for global reasoning. DEAR achieves 90.97 nDCG@10 on NovelEval, surpassing GPT-4 by +3.09, and 54.29 Top-1 accuracy on Natural Questions, outperforming MonoT5 and RankGPT. With an inference time of 2.2s (pointwise) and 11.16s (listwise), DEAR offers an efficient, open-source solution for advanced reranking.

Limitations

While DEAR achieves state-of-the-art performance in document reranking, it has several limitations. First, the dual-stage training pipeline relies on synthetic data generated by GPT-4o for listwise reasoning, which may introduce biases or errors from the teacher model, such as hallucinations or misrankings, potentially affecting generalization. Second, the framework's performance is evaluated on top-100 candidates retrieved by BM25, making it dependent on the quality of the initial retrieval stage. Third, the listwise stage processes smaller candidate sets (e.g., top-20), which may limit its ability to handle larger sets due to context window constraints in LLMs. Finally, while DEAR is efficient compared to baselines like RankZephyr, the two-stage process increases computational complexity compared to single-stage rerankers, which may pose challenges for resource-constrained environments.

Acknowledgments

We acknowledge the EuroHPC Joint Undertaking for awarding this project access to the EuroHPC supercomputer LEONARDO, hosted by CINECA (Italy) and the LEONARDO consortium through an EuroHPC

References

Abdelrahman Abdallah, Mahmoud Abdalla, Bhawna Piryani, Jamshid Mozafari, Mohammed Ali, and Adam Jatowt. 2025a. Rankarena: A unified platform for evaluating retrieval, reranking and rag with human and llm feedback. *arXiv preprint arXiv:2508.05512*.

Abdelrahman Abdallah, Jamshid Mozafari, Bhawna Piryani, and Adam Jatowt. 2025b. Asrank: Zero-shot re-ranking with answer scent for document retrieval. *arXiv preprint arXiv:2501.15245*.

Abdelrahman Abdallah, Bhawna Piryani, Jamshid Mozafari, Mohammed Ali, and Adam Jatowt. 2025c. How good are llm-based rerankers? an empirical analysis of state-of-the-art reranking models. *arXiv preprint arXiv:2508.16757*.

Abdelrahman Abdallah, Bhawna Piryani, Jamshid Mozafari, Mohammed Ali, and Adam Jatowt. 2025d. Rankify: A comprehensive python toolkit for retrieval, re-ranking, and retrieval-augmented generation. *arXiv preprint arXiv:2502.02464*.

Abdelrahman Abdallah, Bhawna Piryani, Jonas Wallat, Avishek Anand, and Adam Jatowt. 2025e. Tempretriever: Fusion-based temporal dense passage retrieval for time-sensitive questions. *arXiv preprint arXiv:2502.21024*.

Abdelrahman Elsayed Mahmoud Abdallah, Jamshid Mozafari, Bhawna Piryani, Mohammed M Abdelgwad, and Adam Jatowt. 2025f. Dynrank: Improve passage retrieval with dynamic zero-shot prompting based on question classification. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 4768–4778.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, and 1 others. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, and 1 others. 2016. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*.

Davide Baldelli, Junfeng Jiang, Akiko Aizawa, and Paolo Torroni. 2024. Twolar: a two-step llm-augmented distillation method for passage reranking. In *European Conference on Information Retrieval*, pages 470–485. Springer.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1533–1544, Seattle, Washington, USA. Association for Computational Linguistics.

Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. 2005. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96.

Christopher JC Burges. 2010. From ranknet to lambdarank to lambdamart: An overview. *Learning*, 11(23-581):81.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051*.

Shijie Chen, Bernal Jiménez Gutiérrez, and Yu Su. 2024. Attention in large language models yields efficient zero-shot re-rankers. *Preprint*, arXiv:2410.02642.

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

Yao Fu, Hao-Chun Peng, Litu Ou, Ashish Sabharwal, and Tushar Khot. 2023. Specializing smaller language models towards multi-step reasoning. In *ICML 2023*.

Raphael Gruber, Abdelrahman Abdallah, Michael Färber, and Adam Jatowt. 2024. Complextempqa: A large-scale dataset for complex temporal question answering. *arXiv preprint arXiv:2406.04866*.

Jianyuan Guo, Kai Han, Yunhe Wang, Han Wu, Xinghao Chen, Chunjing Xu, and Chang Xu. 2021. Distilling object detectors via decoupled features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2154–2164.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

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

Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446.

Yuelyu Ji, Zhuochun Li, Rui Meng, and Daqing He. 2024. Reasoningrank: Teaching student models to rank through reasoning-based knowledge distillation. *arXiv preprint arXiv:2410.05168*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick SH Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *EMNLP (1)*, pages 6769–6781.

Diederik P Kingma. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, and 1 others. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Thiago Laitz, Konstantinos Papakostas, Roberto Lotufo, and Rodrigo Nogueira. 2024. Inranker: Distilled rankers for zero-shot information retrieval. *Preprint*, arXiv:2401.06910.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021a. 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)*, pages 2356–2362.

Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021b. Pyserini: An easy-to-use python toolkit to support replicable ir research with sparse and dense representations. *arXiv preprint arXiv:2102.10073*.

Xueguang Ma, Xinyu Zhang, Ronak Pradeep, and Jimmy Lin. 2023. Zero-shot listwise document reranking with a large language model. *arXiv preprint arXiv:2305.02156*.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. 2023. Teaching small language models to reason. In *ACL 2023*.

Rodrigo Nogueira, Zhiying Jiang, and Jimmy Lin. 2020. Document ranking with a pretrained sequence-to-sequence model. *arXiv preprint arXiv:2003.06713*.

Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. 2019a. Multi-stage document ranking with bert. *arXiv preprint arXiv:1910.14424*.

Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. 2019b. Multi-stage document ranking with bert. *arXiv preprint arXiv:1910.14424*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Ronak Pradeep, Rodrigo Nogueira, and Jimmy Lin. 2021. The expando-mono-duo design pattern for text ranking with pretrained sequence-to-sequence models. *arXiv preprint arXiv:2101.05667*.

Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. 2023a. Rankvicuna: Zero-shot listwise document reranking with open-source large language models. *arXiv preprint arXiv:2309.15088*.

Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. 2023b. Rankzephyr: Effective and robust zero-shot listwise reranking is a breeze! *arXiv preprint arXiv:2312.02724*.

Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Le Yan, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, and 1 others. 2023. Large language models are effective text rankers with pairwise ranking prompting. *arXiv preprint arXiv:2306.17563*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Muhammad Shihab Rashid, Jannat Ara Meem, Yue Dong, and Vagelis Hristidis. 2024. Ecorank: Budget-constrained text re-ranking using large language models. *arXiv preprint arXiv:2402.10866*.

Devendra Singh Sachan, Mike Lewis, Mandar Joshi, Armen Aghajanyan, Wen-tau Yih, Joelle Pineau, and Luke Zettlemoyer. 2022. Improving passage retrieval with zero-shot question generation. *arXiv preprint arXiv:2204.07496*.

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. *arXiv preprint arXiv:2304.09542*.

Manveer Singh Tamber, Ronak Pradeep, and Jimmy Lin. 2023. Scaling down, litting up: Efficient zero-shot listwise reranking with seq2seq encoder-decoder models. *arXiv preprint arXiv: 2312.16098*.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. Fever: a large-scale dataset for fact extraction and verification. *arXiv preprint arXiv:1803.05355*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Lin Wang and Kuk-Jin Yoon. 2021. Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. *TPAMI*.

Yining Wang, Liwei Wang, Yuanzhi Li, Di He, and Tie-Yan Liu. 2013. A theoretical analysis of ndcg type ranking measures. In *Conference on learning theory*, pages 25–54. PMLR.

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.

A Synthetic Data Generation Prompt

Figure 7 illustrates the prompt used to generate synthetic chain-of-thought (CoT) reasoning examples for the listwise reranking stage of DEAR. This prompt, provided to GPT-4o, instructs the model to: (1) identify information requirements for a given query, (2) match candidate passages to these requirements, and (3) produce a ranked list of passage identifiers with step-by-step reasoning. The resulting 20K synthetic examples, each comprising a query, candidate passages, CoT reasoning, and ranked output, are used to fine-tune the student model (LLaMA3.1-8B) for listwise reranking, enhancing its ability to reason globally over document sets and generate interpretable rankings.

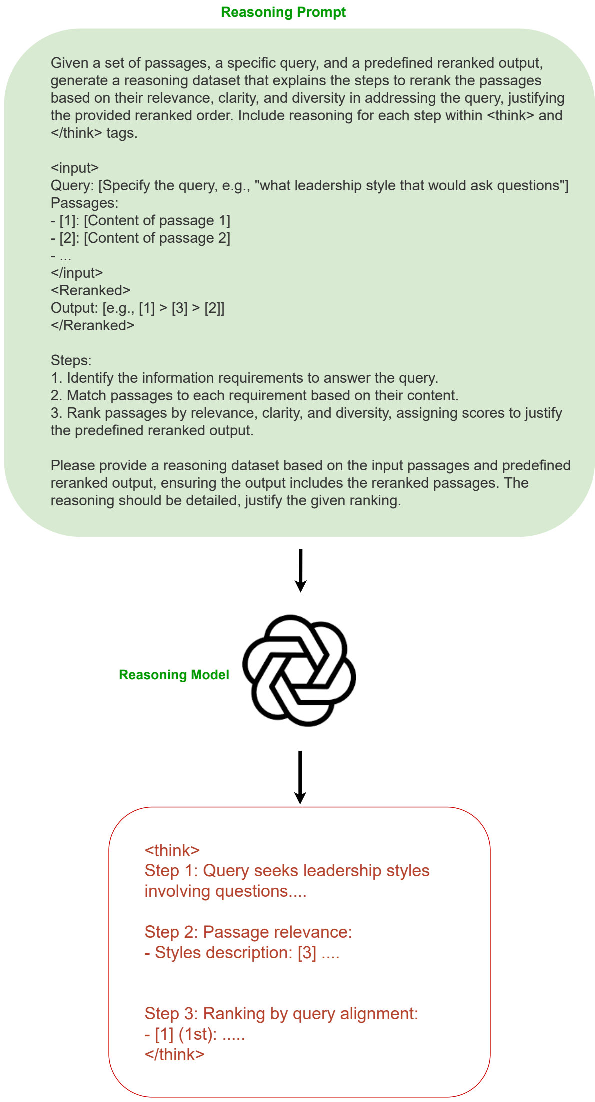

Figure 7: Prompt used to generate synthetic chains.

B Comparison with Reasoning Models

We compare DEAR 's chain-of-thought (CoT) reasoning approach with Reason-to-Rank (R2R) (Ji et al., 2024), which uses direct relevance and comparative reasoning, alongside other baselines. Table 6 reports nDCG@5 performance across TREC DL19/20 and six BEIR datasets (Covid, Touche, News, NFCorpus, Robust04, DBPedia). Figure 4 visualizes these results, highlighting DEAR 's gains.

In the pointwise stage, DEAR-P (LLaMA3.1-8B BC) achieves 76.52 on DL19 and 54.65 BEIR average, outperforming R2R-LLaMA3.1-8B (75.4 and 52.0) and competitive with R2R-GPT-4 (77.7 and 54.4). The 3B RL model scores 53.39 BEIR average, showcasing scalability. In the listwise stage, DEAR-L (LLaMA3.1-8B BC with CoT

GPT) reaches 80.71 on DL19 and 56.92 BEIR average, surpassing R2R-GPT-4 (77.7 and 54.4) and R2R-LLaMA3.1-8B (75.4 and 52.0). Notable gains occur on Covid (90.53 vs. 85.3), Robust04 (69.02 vs. 58.6), and NFCorpus (45.75 vs. 36.3). DEAR-L with CoT LLaMA achieves 57.78 BEIR average, with strong performance on Covid (91.44) and Robust04 (65.69), demonstrating robust multi-document reasoning.

These results, visualized in Figure 4, confirm DEAR 's CoT-guided listwise reranking enhances performance over R2R's reasoning strategies, leveraging open-source efficiency to outperform larger proprietary models while maintaining interpretability.

C Impact of Teacher–Student Pairing

We extend the ablation study from Section 5.1 to evaluate DEAR-Pointwise's performance across various teacher–student pairs using binary cross-entropy loss, as shown in Table 7. The table reports nDCG@10 across TREC DL19/20 and eight BEIR datasets (Covid, DBPedia, News, NFCorpus, Robust04, Scifact, Signal, Touche). The LLaMA2-13B to LLaMA3.1-8B pairing achieves the highest BEIR average (51.95), with strong performance on Covid (84.14) and Robust04 (52.43). Qwen2.5-14B to Qwen2.5-7B follows closely with a BEIR average of 51.24, excelling on Scifact (76.52) and DBPedia (46.30). Smaller models remain competitive: LLaMA-3.2-1B (49.72 BEIR average) and Qwen3-1.7B (49.58) perform robustly, particularly on Touche (37.18 and 31.91, respectively). These results, visualized for the LLaMA2-13B to LLaMA3.1-8B pair in Figure 5 for alpha selection, highlight DEAR 's flexibility across diverse model architectures and sizes. Listwise reranking, applied to the top-20 pointwise outputs, is expected to further enhance these results, as demonstrated in Section 4.2.

Table 6: NDCG@5 performance (in percentage) for student models and baseline comparisons across multiple datasets.

Models	prev.	Top-K	DL19	DL20	BEIR-6 Avg.	Covid	Touche	News	NFCorpus	Robust04	DBPedia
Baseline Models											
BM25	-	-	52.78	50.67	44.04	63.24	48.11	41.28	35.66	43.59	32.36
DeBERTa	BM25	100	68.5	64.2	47.3	73.4	32.1	50.2	33.7	49.2	45.4
MonoT5	BM25	100	74.5	70.4	–	80.0	34.1	–	–	46.0	35.2
RankVicuna	BM25	100	71.1	68.7	-	67.1	48.7	–	38.5	55.7	35.3
RankZephyr	BM25	100	72.2	70.5	51.7	85.1	36.5	53.3	38.9	60.7	35.5
APEER	BM25	100	74.6	72.3	51.1	83.9	35.3	52.1	33.4	56.0	46.1
R2R (GPT-4)	BM25	100	77.7	73.2	54.4	85.3	38.3	58.4	36.3	58.6	49.5
R2R (Claude)	BM25	100	72.1	70.0	51.8	84.0	37.3	55.5	36.4	52.7	44.9
R2R (Gemini)	BM25	100	71.4	68.6	51.0	83.5	37.0	53.8	36.1	51.8	43.7
R2R (LLaMA3.1 8B)	BM25	100	75.4	72.4	52.0	84.6	36.2	53.8	36.4	53.5	47.9
DEAR-Pointwise (DEAR-P)											
Llama3.1-8B (RL)†	BM25	100	75.86	70.41	54.93	89.13	38.73	52.17	41.19	60.43	47.90
Llama3.1-8B (BC)‡	BM25	100	76.52	71.85	54.65	87.11	39.47	53.40	41.42	57.95	48.53
Llama3.1-3B (RL)§	BM25	100	76.08	72.64	53.39	86.06	37.58	51.33	40.70	56.06	48.58
Llama3.1-3B (BC)¶	BM25	100	78.15	72.87	53.15	86.08	39.63	49.54	40.08	56.61	46.96
DEAR-Listwise (CoT GPT)											
GPT-4	†	30	78.88	76.18	56.26	90.29	32.44	51.74	45.52	69.49	48.07
GPT-4	‡	30	80.71	76.11	56.92	90.53	34.66	52.89	45.75	69.02	48.69
GPT-4	§	30	78.09	76.77	57.33	89.36	37.67	53.42	45.76	69.43	48.35
GPT-4	¶	30	80.09	74.98	56.99	89.53	37.65	52.53	45.54	68.89	47.79
DEAR-Listwise (CoT LLaMA)											
Llama3.1-8B	†	30	76.93	75.63	56.18	88.49	37.41	52.08	43.31	66.45	49.33
Llama3.1-8B	‡	30	78.23	74.27	57.12	89.91	38.94	55.10	42.93	66.49	49.36
Llama3.1-8B	§	30	76.91	75.40	56.69	91.94	35.73	52.68	43.92	66.26	49.63
Llama3.1-8B	¶	30	78.40	74.28	57.78	91.44	38.03	51.85	43.81	65.69	49.86

Teacher	Student	dl19	dl20	covid	dbpedia	news	nfc	robust04	scifact	signal	touche	Avg BEIR
Qwen3-14B	Qwen3-1.7B	73.31	68.00	83.25	43.69	49.25	35.70	49.96	74.88	28.07	31.91	49.58
Qwen3-14B	Qwen3-4B	74.04	66.94	82.41	45.47	50.92	35.56	52.07	76.09	28.57	33.68	50.59
Qwen2.5-14B	Qwen2.5-7B	74.06	66.15	83.21	46.30	49.88	35.83	52.86	76.52	29.09	36.29	51.24
Qwen2.5-14B	Qwen1.5-1.5B	73.63	65.85	83.78	44.33	50.81	34.83	48.40	73.30	28.33	36.26	50.00
Qwen2.5-14B	Qwen1.5-3B	73.47	66.42	83.42	45.08	49.24	35.59	49.49	74.74	27.27	36.81	50.20
LLaMA2-13B	LLaMA-3.2-3B	74.49	69.02	82.91	45.28	48.99	36.17	50.93	75.48	29.14	35.78	50.58
LLaMA2-13B	LLaMA-3.2-1B	72.82	68.08	79.00	43.60	47.65	35.79	47.46	75.88	31.21	37.18	49.72
LLaMA2-13B	LLaMA-3.1-8B	74.50	68.71	84.14	46.27	51.71	36.57	52.43	77.39	29.91	37.23	51.95

Table 7: nDCG@10 performance (in percentage) of DEAR-Pointwise with different teacher and student model pairs using binary cross-entropy across TREC and BEIR datasets.