Distillation versus Contrastive Learning: How to Train Your Rerankers

Anonymous ACL submission

Abstract

Training effective text rerankers is crucial for information retrieval. Two strategies are widely used: contrastive learning (optimizing directly on ground-truth labels) and knowledge distillation (transferring knowledge from a larger reranker). While both have been studied extensively, a clear comparison of their effectiveness for training cross-encoder rerankers under practical conditions is needed.

This paper empirically compares these strategies by training rerankers of different sizes and architectures using both methods on the same data, with a strong contrastive learning model acting as the distillation teacher. Our results show that knowledge distillation generally yields better in-domain and out-of-domain ranking performance than contrastive learning when distilling from a larger teacher model. This finding is consistent across student model sizes and architectures. However, distilling from a teacher of the same capacity does not provide the same advantage, particularly for out-of-domain tasks. These findings offer practical guidance for choosing a training strategy based on available teacher models. We recommend using knowledge distillation to train smaller rerankers if a larger, more powerful teacher is accessible; in its absence, contrastive learning remains a robust baseline.

1 Introduction

Modern information retrieval (IR) systems often rely on a two-stage process: an initial retriever quickly finds candidate texts, and a more powerful reranker re-orders them by relevance to improve the final ranking quality (Schütze et al., 2008; Ma et al., 2023; Asai et al., 2024; Singh et al., 2025, *inter alia*). This reranking stage, which involves scoring and sorting texts by their relevance to a given query, commonly employs powerful cross-encoders (Yates et al., 2021; Zhuang et al., 2024b).

Effective training is key to building a high-performing cross-encoder reranker. Two main strategies have emerged for this purpose. The first, contrastive learning (CL), trains the model directly using ground-truth relevance labels, learning to distinguish positive (relevant) examples from negative (irrelevant) ones (Oord et al., 2018; Gao et al., 2021; Ma et al., 2023). The second, knowledge distillation (KD), instead involves training a smaller "student" model to replicate the outputs of a larger, more capable "teacher" model (Buciluă et al., 2006; Hinton et al., 2015; Hofstätter et al., 2020; Schlatt et al., 2025). This method is often employed to create efficient models that can approximate the performance of larger ones.

Both strategies are widely used to train rerankers. But for practical deployments, is one preferable to the other? Prior work has explored related, but distinct, questions. For instance, Hofstätter et al. (2020) explored distilling from rerankers to retrievers; while Baldelli et al. (2024) and Schlatt et al. (2025) focused on training rerankers via distilling from large language models (LLMs) like GPT-4 (Achiam et al., 2023). However, there is no direct comparison of these two strategies.

In this paper, we address this gap. We present an empirical comparison of contrastive learning and knowledge distillation as training strategies for cross-encoder rerankers. To this end, we conduct a suite of controlled experiments. We train cross-encoder models of various sizes (0.5B, 1.5B, 3B, 7B) and different architectures (Transformer and Recurrent) using both strategies. For knowledge distillation, we employ a performant 7B Qwen2.5 model (Yang et al., 2024) trained with contrastive learning as the teacher. We focus on modern decoder-only models, as recent works have reported their effectiveness over smaller encoder-based models like BERT (Ma et al., 2023; Muennighoff et al., 2024a; Zhang et al., 2025, *inter alia*). We train all models on the same dataset and evaluate their ranking performance on standard in-domain and out-of-domain benchmarks.

Our findings consistently show that training with knowledge distillation yields better ranking performance than direct contrastive learning when the student model is smaller than the teacher. For instance, distilling from our 7B teacher significantly improved the performance of 0.5B, 1.5B, and 3B student models on both in-domain and out-of-domain benchmarks compared to training them with contrastive learning. This advantage holds across both Transformer and Recurrent model architectures. However, this benefit diminishes when the student and teacher have the same capacity; distilling from a 7B teacher to a 7B student yielded no significant improvements and, in some cases, harmed out-of-domain generalization (Gou et al., 2021; Gholami and Omar, 2023). To further test the robustness of our conclusions, we conduct an additional experiment where models are trained on a diverse, multi-domain dataset RLHN (Thakur et al., 2025) instead of only on hard negatives mined from a single source. In this setting, knowledge distillation again proves to be the more effective strategy, confirming its advantage even when the reranker's training data is decoupled from the first-stage retriever. Together, these findings position knowledge distillation as a powerful and robust strategy for training smaller rerankers, provided a strong, larger teacher is available. In its absence, we find contrastive learning remains a strong baseline.

2 Methodology

Training strategy is critical for cross-encoder reranker's ranking performance. Two primary strategies have been extensively studied in the literature: (1) direct optimization on ground-truth labels via contrastive learning (Gao et al., 2021; Yates et al., 2021; Ma et al., 2023), and (2) knowledge transfer from a larger model via knowledge distillation (Hofstätter et al., 2020; Schlatt et al., 2025, *inter alia*).

Objective: This paper aims to empirically compare contrastive learning and knowledge distillation for training cross-encoder rerankers. We aim to elucidate their respective strengths and provide clear guidance on which strategy is preferable under different practical constraints.

In the rest of this section, we first formally define the text reranking problem and our notation (§ 2.1). We then provide detailed technical descriptions of the training process using contrastive learning (§ 2.2) and knowledge distillation (§ 2.3).

2.1 The Text Reranking Problem

Modern IR systems often employ a two-stage retrieval-and-rerank pipeline (Schütze et al., 2008; Zhang et al., 2021; Asai et al., 2024, *inter alia*). An efficient first-stage retriever initially fetches a broad set of candidate texts. Subsequently, a more powerful reranker refines this initial list to optimize ranking metrics. Reranking is the task of ordering texts (e.g., passages or documents) by their relevance to a given query.

Let q be an input query, and d be a text from a corpus $\mathcal{D}$. We define a reranking model $f_\theta(q, d)$, parameterized by θ, which computes a scalar relevance score. This model is typically a cross-encoder: the query q and text d are concatenated and fed into a transformer-based language model, whose output is passed through a linear layer to produce the final score (Yates et al., 2021; Nogueira et al., 2019; Boytsov et al., 2022; Ma et al., 2023; Xu, 2024; Xu et al., 2025b, *inter alia*).

2.2 Training with Contrastive Learning

Contrastive learning is one strategy for learning representations (Oord et al., 2018; Weng, 2021). Its application to reranking builds on the principle of the InfoNCE loss (Oord et al., 2018), which is derived from Noise-Contrastive Estimation (Gutmann and Hyvärinen, 2010).

The general goal is to learn a model that distinguishes a "positive" data sample from a set of "negative" (or noise) samples, given a certain context. Denote a context vector $\mathbf{c}$, and consider a set of N samples $X = \{\mathbf{x}_i\}_{i=1}^N$, where one sample $\mathbf{x}_{\text{pos}}$ is a positive sample drawn from the conditional distribution $p(\mathbf{x}|\mathbf{c})$, and the $N-1$ negative samples are drawn from a proposal distribution $p(\mathbf{x})$. Using Bayes' rule, the probability that a sample $\mathbf{x}_i$ is the positive one is:

$$p(C = \text{pos}|X, \mathbf{c}) = \frac{p(\mathbf{x}_{\text{pos}}|\mathbf{c}) \prod_{i \neq \text{pos}} p(\mathbf{x}_i)}{\sum_j \left[p(\mathbf{x}_j|\mathbf{c}) \prod_{i \neq j} p(\mathbf{x}_i) \right]} \tag{166}$$

$$= \frac{p(\mathbf{x}_{\text{pos}}|\mathbf{c}) / p(\mathbf{x}_{\text{pos}})}{\sum_j p(\mathbf{x}_j|\mathbf{c}) / p(\mathbf{x}_j)} = \frac{f(\mathbf{x}_{\text{pos}}, \mathbf{c})}{\sum_j f(\mathbf{x}_j, \mathbf{c})}$$

We can define the scoring function that is proportional to the density ratio $f(\mathbf{x}, \mathbf{c}) \propto \frac{p(\mathbf{x},\mathbf{c})}{p(\mathbf{c})}$. The

InfoNCE loss optimizes the negative log probability of classifying the positive sample correctly:

$$\mathcal{L}_{\text{InfoNCE}} = -\mathbb{E}\Big[\log \frac{f(\mathbf{x}, \mathbf{c})}{\sum_{\mathbf{x}' \in \mathbf{X}} f(\mathbf{x}', c)}\Big] \quad (1)$$

We map the abstract concepts to our reranking task:

- The context $\mathbf{c}$ is the query q_i.
- The positive sample $\mathbf{x}_{\text{pos}}$ is the relevant document d_i^+.
- The negative samples $\{\mathbf{x}_k\}$ are a set of irrelevant documents D_i^-.
- The scoring function f is the parameterized reranker model f_θ.

For a training instance consisting of a query q_i, a positive document d_i^+, and a set of negative documents D_i^-, the loss is:

$$-\frac{1}{|\mathcal{S}|} \sum_{(q_i, d_i^+) \in \mathcal{S}} \log \frac{\exp f_\theta(q_i, d_i^+)}{\exp f_\theta(q_i, d_i^+) + \sum_{j \in \mathcal{D}_i^-} \exp f_\theta(q_i, d_i^-)}$$

The total loss is averaged over all training instances. Following common practice (Gao et al., 2021; Xu, 2024), the negative documents are often "hard negatives" — documents that the first-stage retriever ranked highly but are not labeled as relevant. In practice, training instances are grouped into mini-batches, and the parameters θ are optimized jointly.

2.3 Training with Knowledge Distillation

Knowledge distillation (KD) is a technique for training a smaller, efficient "student" model by transferring knowledge from a larger, more capable "teacher" model (Hinton et al., 2015; Gou et al., 2021). In IR, KD is used to create fast rerankers that approximate the performance of slower, larger models, which is critical for production systems (Hofstätter et al., 2020; Santhanam et al., 2022, *inter alia*).

Let f_t denote the teacher reranker and f_s denote the student reranker. For a given query q and a list of candidate texts $\mathcal{D}_q = \{d_1, d_2, \ldots, d_k\}$, we first compute relevance scores from both models. This yields two score vectors (logits):

$$\mathbf{z}_t = [f_t(q, d_1), f_t(q, d_2), \ldots, f_t(q, d_k)]$$
$$\mathbf{z}_s = [f_s(q, d_1), f_s(q, d_2), \ldots, f_s(q, d_k)]$$

The student model f_s is trained to mimic the teacher's output distribution over the candidate texts. This is achieved by minimizing the Kullback-Leibler (KL) divergence between the two softened probability distributions:

$$\mathcal{L}_{\text{KD}} = D_{\text{KL}}\left(\text{softmax}\left(\frac{\mathbf{z}_t}{T}\right) \Big| \text{softmax}\left(\frac{\mathbf{z}_s}{T}\right)\right)$$

where T is the temperature hyperparameter. A higher temperature creates a softer probability distribution, which can help in transferring more nuanced information from the teacher. In practice, T is often set to 1 (Hinton et al., 2015).

Whereas contrastive learning optimizes a model on ground-truth labels, knowledge distillation mimics the outputs of a more capable teacher. The central goal of this paper is to empirically compare these distinct paradigms for training cross-encoder rerankers under controlled settings, which we detail in the following section.

3 Experimental Setup

Training setup. We construct our training set based on the well established MS MARCO passage retrieval dataset (Bajaj et al., 2016). The official training set contains 532k training pairs, and the corpus contains 8.8M passages. For first stage retriever, we reproduce the RepLlama experiment (Ma et al., 2023), but replace the Llama-2-7B backbone model (Touvron et al., 2023) with stronger Qwen2.5-7B model (Yang et al., 2024). The retriever — named RepQwen — is trained on `Tevatron/MSMARCO-passage-aug`[1] trainset, which consists of 486k (q_i, d_i^+) pairs with hard negatives mined from BM25 and CoCondenser (Gao and Callan, 2022).

To construct the training set for the rerankers, we follow the same strategy as Ma et al. (2023) to mine hard negatives from the first stage retriever. Specifically, for each (q_i, d_i^+) pair, we randomly sample k negatives from top-200 passages retrieved by the retriever, excluding d_i^+. As shown in the experimental results (§ 4), the models trained using this constructed dataset achieves on par or improved performance over RankLlama2 (Ma et al., 2023), suggesting the correctness of the pipeline. We did not further investigate hard negative mining strategies like prior works (Yu et al., 2024; Thakur et al., 2025; Lee et al., 2025) as they are orthogonal to the goal of this paper.

The (q_i, d_i^+) pairs, together with the mined hard negatives are then used to train rerankers

[1] `https://huggingface.co/datasets/Tevatron/msmarco-passage-aug`

with contrastive learning. We first train the most capable reranker — named RankQwen-7B — with Qwen2.5-7B backbone using contrastive learning. This RankQwen-7B is then used as the teacher model to label $\mathcal{D}_i = \{d_i^+, D_i^-\}$ to be subsequently used to train student rerankers. This way, we create a controlled experiment as the rerankers trained with contrastive learning and knowledge distillation are trained on the same $(q, d, \mathcal{D}^-)$ triples, and the only difference is the training strategy.

Evaluation setup. We evaluate the reranking performance following prior works' practices (Ma et al., 2023; Xu et al., 2025c, *inter alia*). For in-domain evaluation, we use MS MARCO passage dev set, which includes 6,980 queries. We also include TREC DL19 and DL20 (Craswell et al., 2020, 2021) consisting of 43 and 54 queries respectively. For out-of-domain evaluation, we adopt BEIR benchmark (Thakur et al., 2021). We evaluate performances on 13 subsets that are publicly available. Refer to Appx. A for details of datasets.

We report the official evaluation metrics for all benchmarks, i.e., MRR@10 for MS MARCO passage dev, NDCG@10 for DL19 and DL20, NDCG@10 for BEIR benchmarks.

Models used. Our main experiments are based on RepQwen, the retriever with a Qwen2.5-7B backbone following the RepLlama (Ma et al., 2023) training strategy. We also report the performance of a retriever with a Llama-3.1-8B backbone.

The objects of our study are reranker models. We evaluate Qwen2.5 models of different sizes, including 0.5B, 1.3B, 3B and 7B, to allow us to observe the scaling trend. For each model size, we train two reranker models: one via direct contrastive learning and one via distillation; we denote with CL/KD suffix, e.g., RankQwen-0.5B-CL means 0.5B model trained with contrastive learning. For knowledge distillation training, we distil from the RankQwen-7B-CL teacher.

We also experiment with RecurrentGemma (De et al., 2024), a recurrent language model based on the Griffin architecture instead of quadratic complexity Transformers (Vaswani et al., 2017). We use the 2B variant and refer to the trained model as RankRGemma-2B. Recent works have explored recurrent language models' efficacy for IR tasks, such as state space models like Mamba (Gu and Dao, 2023; Dao and Gu, 2024) for retrieval (Zhang et al., 2024) and reranking (Xu, 2024; Xu et al., 2025c). We follow this direction to examine recurrent language models' efficacy under different training strategies.

For all the models used in our experiments, we use the pretrained base models.

Baselines. Our baselines are prior results under the same training setting. RepLlama and RankLlama (Ma et al., 2023) are the closest baselines which use the same trainset for contrastive learning, using Llama-2 backbone (Touvron et al., 2023). CSPLADE (Xu et al., 2025a) is a learned sparse retrieval model with Llama-3-8B backbone (Dubey et al., 2024), and achieves competitive performance compared to dense retrieval models. RankMamba-2 (Xu et al., 2025c) trains Mamba-2-based rerankers for passage reranking, though their results are based on BGE retriever (Xiao et al., 2023). We compare against the original numbers reported by the authors.

Implementation details. Our implementation is based on packages including PyTorch, Huggingface Transformers and Tevatron-v2 (Ma et al., 2025). For all our models, we train with LoRA (Hu et al., 2021) to balance in-domain and out-of-domain performance and reduce overfitting. For scalable training, we use DeepSpeed stage 2 (Aminabadi et al., 2022), activation checkpointing, FlashAttention-2 (Dao, 2024), mixed precision and gradient accumulation. Appx. B gives details about hyperparameters.

4 Results and Analysis

4.1 Main Results

In-domain results. Table 1 reports the reranking performance on MS MARCO Dev and DL19+DL20. We observe our trained retrieval models, i.e., RepLlama3 and RepQwen performs on par or better than RepLlama2 and CSPLADE, the two models trained with same trainset. Similarly, RankQwen-7B-CL is comparable to RankLlama2-7B, suggesting the correctness of our training pipeline.

We now compare performances between rerankers trained with contrastive learning, and rerankers trained with knowledge distillation, with RankQwen-7B-CL as teacher. We notice that with the same model sizes, knowledge distillation achieves better ranking performance compared to contrastive learning. For example, RankQwen-0.5B-KD achieves 43.5 MRR@10 on Dev, and average 75.8 NDCG@10 on DL19+20's 97 queries,

Model	Size	Source		DEV	DL19	DL20
		prev.	top-k	MRR@10	NDCG@10	
Retrieval						
BM25 (Lin et al., 2021)	-	-	$\lvert\mathcal{D}\rvert$	18.4	50.6	48.0
CoCondenser (Gao and Callan, 2022)	110M	-	$\lvert\mathcal{D}\rvert$	38.2	71.7	68.4
RepLlama2 (Ma et al., 2023)	7B	-	$\lvert\mathcal{D}\rvert$	41.2	74.3	72.1
CSPLADE (Xu et al., 2025a)	8B	-	$\lvert\mathcal{D}\rvert$	41.3	74.1	**72.8**
RepQwen♣	7B	-	$\lvert\mathcal{D}\rvert$	42.2	73.2	72.5
RepLlama3♣	8B	-	$\lvert\mathcal{D}\rvert$	**42.6**	**74.4**	**72.8**
Reranking						
cross-SimLM (Wang et al., 2022a)	110M	bi-SimLM	200	43.7	74.6	72.7
RankT5 (Zhuang et al., 2023)	220M	GTR	1000	43.4	-	-
RankMamba (Xu et al., 2025c)	1.3B	BGE	100	38.6	75.8	74.0
RankLlama-7B (Ma et al., 2023)	7B	RepLlama2	200	**44.9**†	75.6	**77.4**†
RankQwen-7B-CL♣	7B	RepQwen	200	44.8‡	**77.4**	77.1
Contrastive Learning						
RankQwen-0.5B-CL	0.5B	RepQwen	200	42.1	75.7	72.9
RankQwen-1.5B-CL	1.5B	RepQwen	200	43.5	75.8	**75.4**
RankQwen-3B-CL	3B	RepQwen	200	**43.9**	**76.8**	**75.4**
RankRGemma-2B-CL	2B	RepQwen	200	43.0	76.0	74.7
Knowledge Distillation						
RankQwen-0.5B-KD	0.5B	RepQwen	200	43.5	76.1	75.5
RankQwen-1.5B-KD	1.5B	RepQwen	200	43.9	76.1	76.8
RankQwen-3B-KD	3B	RepQwen	200	**44.7**	77.4	**77.1**‡
RankQwen-7B-KD	7B	RepQwen	200	**44.7**	**77.5**†‡	77.0
RankRGemma-2B-KD	2B	RepQwen	200	43.6	76.1	75.0

Table 1: Results for passage reranking in-domain evaluation. We mark best results in each section bold; † indicates the overall best result and ‡ indicates the best result among our trained models; for models in *Retrieval* and *Reranking* baseline sections, ♣ denotes the model we trained. Note RankQwen-7B-CL♣ is used as the teacher for distillation.

improving over RankQwen-0.5B-CL's 42.1 and 74.1. The similar observation applies for 1.5B and 3B scale Qwen models as well as RankRGemma — the recurrent model with Griffin architecture. These observations suggest the efficacy of knowledge distillation: the performance improvement is consistent across different model sizes (0.5B, 1.5B, 3B) and model architectures (Transformer, Griffin).

We also note RankQwen-7B-KD achieves similar in-domain performance as RankQwen-3B-KD and the RankQwen-7B-CL teacher, which suggests that the student model is not benefiting from knowledge distillation training when the teacher model is of the same capacity. We will revisit this problem in our discussion of out-of-domain results.

Out-of-domain results. We report the out-of-domain evaluation results in Table 2. The baseline methods' results are deferred to Appx. C.

We note similar observations as in the in-domain setting. Directly comparing contrastive learning to knowledge distillation, RankQwen-0.5B-KD achieves similar performance as RankQwen-0.5B-CL (53.1 average NDCG@10 v.s. 53.3); while RankQwen-1.5B-KD, RankQwen-3B-KD, RankRGemma-2B have better performance over their contrastive learning counterparts.

Among all models with KD training strategy, RankQwen-3B-KD achieves strong out-of-domain performance, averaging 57.5 NDCG@10 over 13 BEIR datasets, improving over RankQwen-7B-CL teacher. However, we notice that RankQwen-7B-CL's underperformance is mainly due to Quora duplicate question retrieval — a symmetric retrieval task different from the asymmetric MS MARCO trainset. The performance is 55.5 versus 56.2 excluding Quora, suggesting that the student model still cannot improve over the teacher (Gou et al., 2021; Gholami and Omar, 2023).

An important observation is that RankQwen-7B-KD leads to performance degradation in out-of-domain evaluation (55.8 vs RankQwen-3B-KD's

Dataset	Teacher 7B	Contrastive Learning 0.5B	1.5B	3B	2B	Knowledge Distillation 0.5B	1.5B	3B	7B	2B
Arguana	55.9	50.6	55.8	**56.2**	52.3	51.3	**57.6**†‡	56.5	55.2	54.7
Climate-FEVER	27.1	23.8	27.6	**30.5**†‡	24.3	23.8	**28.5**	31.6	28.7	26.6
DBPedia	48.7†	44.5	46.3	**48.5**‡	47.0	46.4	47.7	**48.5**‡	48.1	47.6
FEVER	88.2	86.0	86.2	**89.1**†	85.1	86.7	87.3	**88.1**	87.3	86.6
FiQA	45.9	37.9	42.2	43.2	**44.1**	34.7	45.4	**46.6**†‡	44.2	44.7
HotpotQA	76.1	72.5	74.8	**76.2**	74.4	74.4	76.2	**76.7**†‡	72.5	76.6
NFCorpus	31.5	33.0	29.3	25.9	**34.4**†‡	32.2	32.3	33.3	32.3	**34.2**
NQ	65.9†	59.2	63.2	**64.9**	62.8	60.9	64.4	**65.5**‡	64.2	64.2
Quora	79.3	82.3	**84.9**†‡	84.6	79.6	80.2	81.0	**81.9**	78.5	80.3
SCIDOCS	19.1	16.9	16.2	17.3	**18.4**	17.8	**19.2**†	**19.2**†‡	19.1	18.6
SciFact	75.0	72.2	72.1	71.1	**73.7**	75.6	**75.8**†‡	75.5	74.6	75.2
TREC-COVID	85.0	81.9	86.3	**86.8**	85.1	80.7	85.6	86.3	84.7	**87.4**†‡
Touche-2020	36.7	31.9	34.6	**36.5**	36.4	25.9	35.9	**38.6**†‡	35.5	37.8
Average	56.5	53.3	55.3	**56.2**	55.2	53.1	56.7	**57.6**†‡	55.8	56.5
Average w/o Quora	56.2†	50.9	52.9	**53.8**	53.2	50.8	54.6	**55.5**‡	53.9	54.5

Table 2: Results for passage reranking out-of-domain evaluation. 0.5B, 1.5B, 3B, 7B 2B correspond to RankQwen-0.5B, RankQwen-1.5B, RankQwen-3B, RankQwen-7B and RankRGemma-2B, respectively. We also report results without Quora dataset as Quora duplicate question detection is a symmetric retrieval task, not aligning with the asymmetric web search task of the training set. We mark best results in each section bold; † indicates the overall best result including the teacher model, ‡ indicates the best result excluding teacher.

57.6), while their performances on in-domain MS MARCO datasets are comparable. Prior studies have also noted that knowledge distillation suffers from overfitting and poor out-of-domain generalization (Gou et al., 2021; Yuan et al., 2020; Yun et al., 2020). Our experimental results suggest that when distilling from a larger, more capable teacher models (RankQwen-7B-CL in our case), the student still achieve performance improvement compared to contrastive training from scratch.

Scaling model sizes. Scaling has been proven effective in IR tasks (Neelakantan et al., 2022; Muennighoff et al., 2024b; Xu et al., 2025b; Zhu et al., 2023, *inter alia*). We also compare the model size scaling trend between CL versus KD for passage reranking in Fig. 1. We make two observations:

1. We observe that contrastive learning demonstrates a clear scaling trend, i.e., performance improves with increased model size when trained on the same data, as also reported by prior works (Ma et al., 2023; Neelakantan et al., 2022; Muennighoff, 2022; Zhuang et al., 2024a).

2. We also note that when distilling from a larger, more capable teacher model, knowledge distillation also demonstrates a scaling trend. For example, RankQwen's performance on 12 BEIR datasets (excluding Quora) improves from 50.8 to 54.6, 55.5 when scaling from 0.5B to 1.5B and 3B model sizes.

We hypothesize that with our knowledge distillation training strategy, the passage reranking performance can be further improved by scaling up the student model's model sizes, and distilling from stronger teacher models.

4.2 Scaling Training Data

Experiment setup. In § 4.1, we focus on a controlled experiment with various model sizes. Now we examine the efficacy of knowledge distillation when scaling the axis of training data. Notice that doing so leads to the decoupling of the retriever and reranker as we can no longer mine hard negatives from the retriever on MS MARCO passage trainset. We use a recently released trainset, RLHN (Thakur et al., 2025), which consists of 649K (q, d^+, D^-) training samples pruned from the larger BGE trainset (Xiao et al., 2023). Specifically, RLHN comprises train samples from the following dataset: MS MARCO passage (Bajaj et al., 2016), Arguana (Wachsmuth et al., 2018), FEVER (Thorne et al., 2018), FiQA (Maia et al., 2018), SCIDOCS (Cohan et al., 2020), HotpotQA (Yang et al., 2018) and NQ (Kwiatkowski et al., 2019). In this case, BEIR benchmark as a whole is no longer considered out-of-domain evaluation. We use a lightweight `intfloat/e5-base-v2` re-

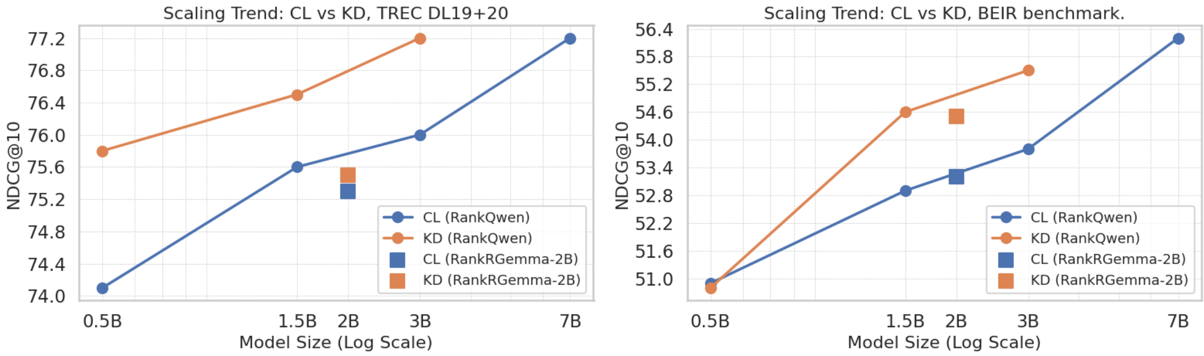

Figure 1: Scaling trend for contrastive learning vs knowledge distillation, for passage reranking. Left figure shows TREC DL19+20 results while right figure shows BEIR results averaged over 12 datasets (excluding Quora). We skip results of RankQwen-7B-KD model.

Dataset	Retriever 110M	Teacher 7B	Contrastive Learning 0.5B	Contrastive Learning 3B	Knowledge Distillation 0.5B	Knowledge Distillation 3B
MSMARCO	42.7	**46.0**	45.5	45.6†	45.3	45.6†
Arguana	44.5	**79.1**	65.2	76.7†	63.7	75.6
Climate-FEVER	26.6	40.5	39.0	40.1	40.2	**42.6**†
DBPedia	42.2	**54.3**	49.9	52.7	50.1	53.7†
FEVER	85.0	94.0	93.6	93.9	93.8	**94.3**†
FiQA	39.9	**55.9**	46.7	55.1†	47.3	54.2
HotpotQA	69.1	**84.8**	82.9	84.7†	82.7	84.6
NFCorpus	35.4	**42.4**	37.0	41.6†	36.8	41.6†
NQ	58.2	**74.5**	67.5	72.8	68.2	73.5†
Quora	86.6	77.6	78.3	78.6	**81.1**†	77.1
SCIDOCS	18.7	**27.1**	22.5	25.7†	22.2	25.7†
SciFact	71.9	**81.9**	78.6	81.3	78.6	81.6†
TREC-COVID	69.5	88.3	85.5	**89.0**†	86.1	88.7
Touche-2020	26.4	32.9	33.1	32.8	35.1	**35.3**†
Average	51.2	**62.8**	58.9	62.2	59.4	62.4†
Average w/o Quora	48.5	**61.7**	57.4	60.9	57.7	61.3†

Table 3: Results with on BEIR benchmark (including MS MARCO passage Dev) with RLHN training mixture. We report NDCG@10 as the performance metric. We mark best result in each row bold; † indicates the overall best result excluding the 7B-sized teacher model.

triever (Wang et al., 2022b) pretrained on unlabeled text pairs, then finetuned with the combined MS MARCO passage and NQ training mixture.

We use the similar training strategy as § 3 (the only difference is the ranker training data): we first train 7B-scale reranker with Qwen2.5-7B and contrastive learning, then train smaller rerankers with knowledge distillation, with the reranking scores on RLHN dataset labeled by the teacher, as well as the contrastive learning counterparts. We train 0.5B and 3B rerankers, as we notice Qwen2.5-1.5B fail to converge in the contrastive learning setting. We name the trained rerankers as RankQwen-{0.5B, 3B, 7B}-CL-RLHN and RankQwen-{0.5B, 3B}-KD-RLHN, respectively.

Results and analysis. We report the results in Table 3. Compared to Table 2, we notice rerankers trained with RLHN mixture show significant performance boost, suggesting the effectiveness of in-domain training data.

Similar to Table 2, the 7B model achieves the best overall performance, average 62.8 NDCG@10 on 14 BEIR benchmark datasets including MS MARCO Dev. We notice that knowledge distillation still outperforms contrastive learning when distilling from the strong 7B teacher model, though the margin is small (<1%). This observation suggests the robustness of knowledge distillation: it can achieve performance improvement compared to contrastive learning when the reranker training data is not coupled with the retriever.

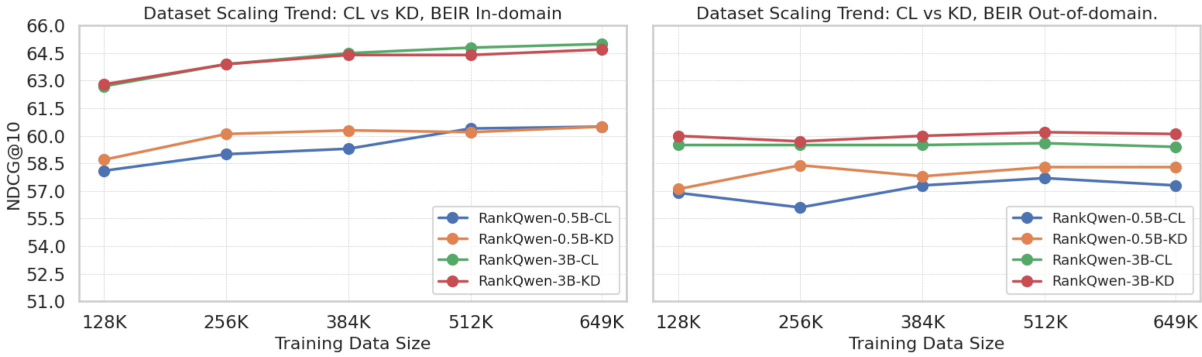

Figure 2: Dataset scaling trend for CL versus KD. Left figure shows average results for 7 in-domain datasets from BEIR benchmark, while the right figure shows average results for 7 out-of-domain datasets.

In Fig. 2 we further analyze BEIR in-domain (7 datasets used in training) and out-of-domain (7 unseen datasets) performance with varying training data sizes. As the amount of training data increases, both CL and KD show improved in-domain performance, with CL eventually outperforming KD. In contrast, KD consistently outperforms CL on the OOD datasets, although the benefits of increasing training data are diminishing. We leave a more in-depth investigation of the in-domain versus OOD performance gap to future work.

5 Related Works

In this paper, we focus exclusively on the architecture of cross-encoder pointwise reranker (Nogueira and Cho, 2019; Nogueira et al., 2020; Zhuang et al., 2024b). The reranker model processes a query and a document simultaneously, allowing the self-attention mechanism to explicitly model the interactions between their tokens throughout the layers. Nogueira and Cho (2019) first demonstrated the efficacy of this approach by fine-tuning BERT as a text pair classifier. The fine-tuning of the BERT-type model as cross-encoder reranker has later been extended to encoder-decoder architectures, such as T5 (Raffel et al., 2020; Nogueira et al., 2020), and decoder-only models like Llama (Touvron et al., 2023; Ma et al., 2023).

The training objective for a cross-encoder is to accurately discriminate between relevant and non-relevant documents for a given query. In practice, the non-relevant documents are usually hard negatives mined from first-stage retrievers (Nogueira et al., 2019; Gao et al., 2021; Boytsov et al., 2022, *inter alia*). Two primary training strategies are widely used to train cross-encoder reranker. The first is contrastive learning, which trains the model directly on ground-truth labels to maximize the distinction between positive and negative examples (Gao et al., 2021; Zhuang et al., 2023). The second strategy is knowledge distillation, which involves training a student reranker to mimic the behavior and performance of a larger, more capable teacher model (Hofstätter et al., 2020; Baldelli et al., 2024; Schlatt et al., 2025). Knowledge distillation (KD, Buciluă et al., 2006; Hinton et al., 2015) facilitates the transfer of knowledge from a large, complex "teacher" model to a smaller, more efficient "student" model. The goal is to enable the student to mimic the teacher's output, thereby inheriting its predictive capabilities at a reduced computational cost. Within the context of training neural IR models, knowledge distillation has been effectively applied to enhance both bi-encoder retriever (Hofstätter et al., 2020; Formal et al., 2021, *inter alia*) and cross-encoder reranker (Baldelli et al., 2024; Schlatt et al., 2025). In this work, we aim to compare these two training strategies in a controlled experimental setting to evaluate their strengths and weaknesses.

6 Conclusion and Future Work

In this paper, we compared the effectiveness of two training strategies, i.e., contrastive learning and knowledge distillation in the context of training text rerankers. With a rigorously controlled experimental setup, we find that when distilling from a larger, more capable teacher model, rerankers trained with knowledge distillation achieve better in-domain and out-of-domain reranking performances compared to contrastive training strategy, and the observation is consistent across different model scales and language model architectures. Our future work will investigate a more optimized way to combine two training strategies, as well as to improve the robustness and out-of-domain generalization of knowledge distillation training.

Limitations

Given the limited computational resources, we are unable to scale the reranker training to >10B models such as Qwen2.5-14B. The observation made in this work may be subject to change for stronger base models. How to improve robustness and out-of-domain generalization of the knowledge distillation training has been extensively investigated by prior works in other NLP domains (Utama et al., 2020; Stacey and Rei, 2024; Wang et al., 2023), we leave investigation for reranking to future works.

This paper focuses on empirical experiments on public benchmarks. We believe this paper do not incur potential risks.

References

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

Reza Yazdani Aminabadi, Samyam Rajbhandari, Ammar Ahmad Awan, Cheng Li, Du Li, Elton Zheng, Olatunji Ruwase, Shaden Smith, Minjia Zhang, Jeff Rasley, et al. 2022. Deepspeed-inference: enabling efficient inference of transformer models at unprecedented scale. In *SC22: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–15. IEEE.

Akari Asai, Jacqueline He, Rulin Shao, Weijia Shi, Amanpreet Singh, Joseph Chee Chang, Kyle Lo, Luca Soldaini, Sergey Feldman, Mike D'arcy, et al. 2024. Openscholar: Synthesizing scientific literature with retrieval-augmented lms. *arXiv preprint arXiv:2411.14199*.

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 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.

Alexander Bondarenko, Maik Fröbe, Meriem Beloucif, Lukas Gienapp, Yamen Ajjour, Alexander Panchenko, Chris Biemann, Benno Stein, Henning Wachsmuth, Martin Potthast, et al. 2020. Overview of touché 2020: argument retrieval. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction: 11th International Conference of the CLEF Association, CLEF 2020, Thessaloniki, Greece, September 22–25, 2020, Proceedings 11*, pages 384–395. Springer.

Vera Boteva, Demian Gholipour, Artem Sokolov, and Stefan Riezler. 2016. A full-text learning to rank dataset for medical information retrieval. In *Advances in Information Retrieval: 38th European Conference on IR Research, ECIR 2016, Padua, Italy, March 20–23, 2016. Proceedings 38*, pages 716–722. Springer.

Leonid Boytsov, Tianyi Lin, Fangwei Gao, Yutian Zhao, Jeffrey Huang, and Eric Nyberg. 2022. Understanding performance of long-document ranking models through comprehensive evaluation and leaderboarding. *arXiv preprint arXiv:2207.01262*.

Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. 2006. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel S Weld. 2020. Specter: Document-level representation learning using citation-informed transformers. *arXiv preprint arXiv:2004.07180*.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, and Daniel Campos. 2021. Overview of the trec 2020 deep learning track. corr abs/2102.07662 (2021). *arXiv preprint arXiv:2102.07662*.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M Voorhees. 2020. Overview of the trec 2019 deep learning track. *arXiv preprint arXiv:2003.07820*.

Tri Dao. 2024. Flashattention-2: Faster attention with better parallelism and work partitioning. In *The Twelfth International Conference on Learning Representations*.

Tri Dao and Albert Gu. 2024. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 10041–10071. PMLR.

Soham De, Samuel L Smith, Anushan Fernando, Aleksandar Botev, George Cristian-Muraru, Albert Gu, Ruba Haroun, Leonard Berrada, Yutian Chen, Srivatsan Srinivasan, et al. 2024. Griffin: Mixing gated linear recurrences with local attention for efficient language models. *arXiv preprint arXiv:2402.19427*.

Thomas Diggelmann, Jordan Boyd-Graber, Jannis Bulian, Massimiliano Ciaramita, and Markus Leippold. 2020. Climate-fever: A dataset for verification of real-world climate claims. *arXiv preprint arXiv:2012.00614*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Thibault Formal, Carlos Lassance, Benjamin Piwowarski, and Stéphane Clinchant. 2021. Splade v2: Sparse lexical and expansion model for information retrieval. *arXiv preprint arXiv:2109.10086*.

Luyu Gao and Jamie Callan. 2022. Unsupervised corpus aware language model pre-training for dense passage retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2843–2853.

Luyu Gao, Zhuyun Dai, and Jamie Callan. 2021. Rethink training of bert rerankers in multi-stage retrieval pipeline. In *Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021, Virtual Event, March 28–April 1, 2021, Proceedings, Part II 43*, pages 280–286. Springer.

Sia Gholami and Marwan Omar. 2023. Can a student large language model perform as well as it's teacher? *ArXiv*, abs/2310.02421.

Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819.

Albert Gu and Tri Dao. 2023. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*.

Michael Gutmann and Aapo Hyvärinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 297–304, Chia Laguna Resort, Sardinia, Italy. PMLR.

Faegheh Hasibi, Fedor Nikolaev, Chenyan Xiong, Krisztian Balog, Svein Erik Bratsberg, Alexander Kotov, and Jamie Callan. 2017. Dbpedia-entity v2: a test collection for entity search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1265–1268.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *stat*, 1050:9.

Sebastian Hofstätter, Sophia Althammer, Michael Schröder, Mete Sertkan, and Allan Hanbury. 2020. Improving efficient neural ranking models with cross-architecture knowledge distillation. *arXiv preprint arXiv:2010.02666*.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2025. NV-embed: Improved techniques for training LLMs as generalist embedding models. In *The Thirteenth International Conference on Learning Representations*.

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.

Xueguang Ma, Luyu Gao, Shengyao Zhuang, Jiaqi Samantha Zhan, Jamie Callan, and Jimmy Lin. 2025. Tevatron 2.0: Unified document retrieval toolkit across scale, language, and modality. *arXiv preprint arXiv:2505.02466*.

Xueguang Ma, Liang Wang, Nan Yang, Furu Wei, and Jimmy Lin. 2023. Fine-tuning llama for multi-stage text retrieval. *arXiv preprint arXiv:2310.08319*.

Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. 2018. Www'18 open challenge: financial opinion mining and question answering. In *Companion proceedings of the the web conference 2018*, pages 1941–1942.

Niklas Muennighoff. 2022. Sgpt: Gpt sentence embeddings for semantic search. *arXiv preprint arXiv:2202.08904*.

Niklas Muennighoff, SU Hongjin, Liang Wang, Nan Yang, Furu Wei, Tao Yu, Amanpreet Singh, and Douwe Kiela. 2024a. Generative representational instruction tuning. In *ICLR 2024 Workshop: How Far Are We From AGI*.

Niklas Muennighoff, SU Hongjin, Liang Wang, Nan Yang, Furu Wei, Tao Yu, Amanpreet Singh, and Douwe Kiela. 2024b. Generative representational instruction tuning. In *ICLR 2024 Workshop: How Far Are We From AGI*.

Arvind Neelakantan, Tao Xu, Raul Puri, Alec Radford, Jesse Michael Han, Jerry Tworek, Qiming Yuan, Nikolas Tezak, Jong Wook Kim, Chris Hallacy, et al.

2022. Text and code embeddings by contrastive pre-training. *arXiv preprint arXiv:2201.10005*.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with bert. *arXiv preprint arXiv:1901.04085*.

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.

Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019. Document expansion by query prediction. *arXiv preprint arXiv:1904.08375*.

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

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.

Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. 2022. ColBERTv2: Effective and efficient retrieval via lightweight late interaction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3715–3734, Seattle, United States. Association for Computational Linguistics.

Ferdinand Schlatt, Maik Fröbe, Harrisen Scells, Shengyao Zhuang, Bevan Koopman, Guido Zuccon, Benno Stein, Martin Potthast, and Matthias Hagen. 2025. Rank-distillm: closing the effectiveness gap between cross-encoders and llms for passage re-ranking. In *European Conference on Information Retrieval*, pages 323–334. Springer.

Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. 2008. *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge.

Amanpreet Singh, Joseph Chee Chang, Chloe Anastasiades, Dany Haddad, Aakanksha Naik, Amber Tanaka, Angele Zamarron, Cecile Nguyen, Jena D Hwang, Jason Dunkleberger, et al. 2025. Ai2 scholar qa: Organized literature synthesis with attribution. *arXiv preprint arXiv:2504.10861*.

Joe Stacey and Marek Rei. 2024. Distilling robustness into natural language inference models with domain-targeted augmentation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2239–2258, Bangkok, Thailand. Association for Computational Linguistics.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. Beir: A heterogeneous benchmark for zero-shot evaluation of information retrieval models.

Nandan Thakur, Crystina Zhang, Xueguang Ma, and Jimmy Lin. 2025. Fixing data that hurts performance: Cascading llms to relabel hard negatives for robust information retrieval. *arXiv preprint arXiv:2505.16967*.

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, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. 2020. Mind the trade-off: Debiasing nlu models without degrading the in-distribution performance. In *Annual Meeting of the Association for Computational Linguistics*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Ellen Voorhees, Tasmeer Alam, Steven Bedrick, Dina Demner-Fushman, William R Hersh, Kyle Lo, Kirk Roberts, Ian Soboroff, and Lucy Lu Wang. 2021. Trec-covid: constructing a pandemic information retrieval test collection. In *ACM SIGIR Forum*, volume 54, pages 1–12. ACM New York, NY, USA.

Henning Wachsmuth, Shahbaz Syed, and Benno Stein. 2018. Retrieval of the best counterargument without prior topic knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 241–251.

David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or fiction: Verifying scientific claims. *arXiv preprint arXiv:2004.14974*.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022a. Simlm: Pre-training with representation bottleneck for dense passage retrieval. *arXiv preprint arXiv:2207.02578*.

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

Xin-Yue Wang, Zitao Wang, and Wei Hu. 2023. Serial contrastive knowledge distillation for continual few-shot relation extraction. *ArXiv*, abs/2305.06616.

Lilian Weng. 2021. Contrastive representation learning. *lilianweng.github.io*.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding. *Preprint*, arXiv:2309.07597.

Zhichao Xu. 2024. Rankmamba, benchmarking mamba's document ranking performance in the era of transformers. *arXiv preprint arXiv:2403.18276*.

Zhichao Xu, Aosong Feng, Yijun Tian, Haibo Ding, and Lin Lee Cheong. 2025a. Csplade: Learned sparse retrieval with causal language models. *arXiv preprint arXiv:2504.10816*.

Zhichao Xu, Fengran Mo, Zhiqi Huang, Crystina Zhang, Puxuan Yu, Bei Wang, Jimmy Lin, and Vivek Srikumar. 2025b. A survey of model architectures in information retrieval. *arXiv preprint arXiv:2502.14822*.

Zhichao Xu, Jinghua Yan, Ashim Gupta, and Vivek Srikumar. 2025c. State space models are strong text rerankers. In *Proceedings of the 10th Workshop on Representation Learning for NLP (RepL4NLP-2025)*, pages 152–169, Albuquerque, NM. Association for Computational Linguistics.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, et al. 2024. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*.

Andrew Yates, Rodrigo Nogueira, and Jimmy Lin. 2021. Pretrained transformers for text ranking: Bert and beyond. In *Proceedings of the 14th ACM International Conference on web search and data mining*, pages 1154–1156.

Puxuan Yu, Luke Merrick, Gaurav Nuti, and Daniel Campos. 2024. Arctic-embed 2.0: Multilingual retrieval without compromise. *arXiv preprint arXiv:2412.04506*.

Li Yuan, Francis E. H. Tay, Guilin Li, Tao Wang, and Jiashi Feng. 2020. Revisiting knowledge distillation via label smoothing regularization. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3902–3910.

Sukmin Yun, Jongjin Park, Kimin Lee, and Jinwoo Shin. 2020. Regularizing class-wise predictions via self-knowledge distillation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13873–13882.

Hanqi Zhang, Chong Chen, Lang Mei, Qi Liu, and Jiaxin Mao. 2024. Mamba retriever: Utilizing mamba for effective and efficient dense retrieval. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 4268–4272.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, et al. 2025. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*.

Yue Zhang, ChengCheng Hu, Yuqi Liu, Hui Fang, and Jimmy Lin. 2021. Learning to rank in the age of Muppets: Effectiveness–efficiency tradeoffs in multi-stage ranking. In *Proceedings of the Second Workshop on Simple and Efficient Natural Language Processing*, pages 64–73, Virtual. Association for Computational Linguistics.

Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Haonan Chen, Zheng Liu, Zhicheng Dou, and Ji-Rong Wen. 2023. Large language models for information retrieval: A survey. *arXiv preprint arXiv:2308.07107*.

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, Bevan Koopman, Jimmy Lin, and Guido Zuccon. 2024a. PromptReps: Prompting large language models to generate dense and sparse representations for zero-shot document retrieval. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4375–4391, Miami, Florida, USA. Association for Computational Linguistics.

Shengyao Zhuang, Honglei Zhuang, Bevan Koopman, and Guido Zuccon. 2024b. 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.

A Dataset Artifacts and Licenses

Four of the datasets we used in experiments (NFCorpus (Boteva et al., 2016), FiQA-2018 (Maia et al., 2018), Quora[2], Climate-Fever (Diggelmann et al., 2020)) do not report the dataset license in the

[2] https://www.kaggle.com/c/quora-question-pairs

paper or a repository. For the rest of the datasets, we list their licenses below:

- MS MARCO (Bajaj et al., 2016): MIT License for non-commercial research purposes.
- ArguAna (Wachsmuth et al., 2018): CC BY 4.0 license.
- DBPedia (Hasibi et al., 2017): CC BY-SA 3.0 license.
- FEVER (Thorne et al., 2018): CC BY-SA 3.0 license.
- HotpotQA (Yang et al., 2018): CC BY-SA 4.0 license.
- NQ (Kwiatkowski et al., 2019): CC BY-SA 3.0 license.
- SCIDOCS (Cohan et al., 2020): GNU General Public License v3.0 license.
- SciFact (Wadden et al., 2020): CC BY-NC 2.0 license.
- TREC-COVID (Voorhees et al., 2021): "Dataset License Agreement".
- Touche-2020 (Bondarenko et al., 2020): CC BY 4.0 license.

B Hyperparameters

For all our training runs, we use a similar set of optimized hyperparameters identified from prior works and our preliminary experiments (Ma et al., 2023; Xu et al., 2025a), only ablating learning rate to reduce the effect of overfitting. We use LoRA rank=16 and α=32. We use AdamW optimizer, learning rate ranging from 3e-5 to 1e-4 with linear warmup and cool down. As we train all models on the train dataset for 1 epoch, we find learning rate is the most important hyperparameter to control overfitting. We use 8 GPUs with per device batch 4, gradient accumulation steps 4, which leads to a global batch size of 128. For each (q_i, d_i^+) pair, we use $k = 15$ negatives as we find increasing to 31 negatives may lead to instable contrastive learning. For RepQwen training, we use in-batch negatives; while for reranker training, we focus solely on each (q_i, d_i^+) pair's own hard negatives.

C Baseline Results

We report the baseline results in Table 4.

Dataset	BM25 -	GTR-XXL 4.8B	RepLlama2 7B	CSPLADE 8B	RepQwen♣ 7B	RankT5 220M	RankMamba 1.3B	RankLlama2 7B
Arguana	39.7	54.0	48.6	48.9	54.7	33.0	34.4	56.0
Climate-FEVER	16.5	26.7	31.0	29.4	29.6	21.5	26.2	28.0
DBPedia	31.8	40.8	43.7	44.5	45.2	44.2	45.8	48.3
FEVER	65.1	74.0	83.4	86.5	79.9	83.2	81.9	83.9
FiQA	23.6	46.7	45.8	40.5	45.4	44.5	43.3	46.5
HotpotQA	63.3	59.9	68.5	69.8	68.9	71.0	76.3	75.3
NFCorpus	32.2	34.2	37.8	37.2	38.4	38.1	39.2	30.3
NQ	30.6	56.8	62.4	60.9	62.3	61.4	52.1	66.3
Quora	78.9	89.2	86.8	87.1	87.1	83.1	83.9	85.0
SCIDOCS	14.9	16.1	18.1	17.6	18.3	18.1	19.6	17.8
SciFact	67.9	66.2	75.6	73.9	75.0	75.0	76.8	73.2
TREC-COVID	59.6	50.1	84.7	83.2	85.3	80.7	79.9	85.2
Touche-2020	44.2	25.6	30.5	38.9	36.8	44.0	37.7	40.1
Average	43.7	49.3	55.1	55.3	55.9	53.7	53.6	56.6

Table 4: Baseline results on BEIR datasets. RepQwen♣ is a model we trained.