Learning by Ranking: Data-Efficient Knowledge Distillation from Black-Box LLMs for Information Retrieval

Anonymous ACL submission

Abstract

Recent research has explored distilling knowledge from large language models (LLMs) to optimize retriever models, particularly within the retrieval-augmented generation (RAG) framework. However, most existing training methods rely on extracting supervision signals from LLMs' weights or output probabilities, which is not only resource-intensive but also incompatible with black-box LLMs. In this paper, we introduce *Intermediate Distillation*, a data-efficient knowledge distillation training scheme that treats LLMs as black boxes and distills their knowledge through an innovative LLM-ranker-retriever pipeline, using only the LLMs' ranking generation as the supervision signal. Extensive experiments demonstrate that our proposed method can significantly improve the performance of retriever models with just 1,000 training instances. Moreover, our distilled retriever model greatly enhances performance in question-answering tasks within the RAG framework, demonstrating the potential of LLMs to train smaller models economically and effectively[1].

1 Introduction

The rapid growth and superior performance of large language models (LLMs) (Ouyang et al., 2022; OpenAI, 2023; Wang et al., 2024b) have made them the preferred choice for a wide range of NLP applications (Xi et al., 2023; Wang et al., 2024b; Wu et al., 2023; Zhang et al., 2023a,b). LLMs have demonstrated robust zero-shot ranking abilities in English and various low-resource languages (Adeyemi et al., 2023; Sun et al., 2023). Consequently, researchers have applied LLMs to the information retrieval tasks, which can outperform previous text search and similarity measurement methods (Ma et al., 2023; Xu et al., 2024).

The retrieval-augmented generation (RAG) framework has been widely adopted to alleviate hallucination problems in LLM generation, especially for knowledge-intensive tasks (Lewis et al., 2020). The RAG framework consists of two key components: a retriever to locate relevant information from a large corpus based on provided inputs, and a reader, typically a LLM, which integrates such retrieved information to generate its outputs (Izacard et al., 2023; Shi et al., 2023).

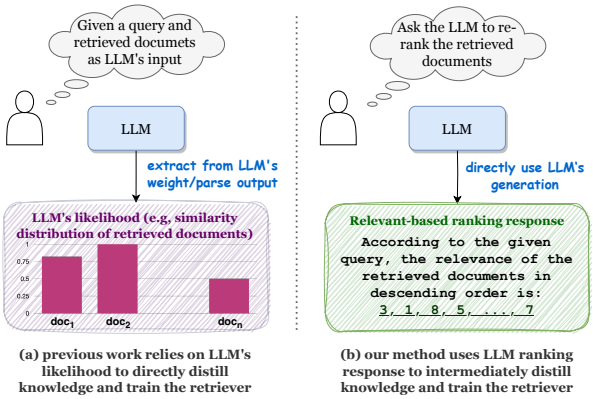

Figure 1: Previous distillation methods (left) rely on extracting supervision signals from LLM's weights or using LLM's output probabilities to train the retriever model. In contrast, our approach (right) bypasses the need for LLM's likelihood, directly using the LLM's ranking responses as supervision signals.

How to distill knowledge from LLMs to optimize the retriever in the RAG framework with in-domain data has been a great challenge. Early efforts proposed training the retriever with white-box LLM readers by extracting supervision signals directly from the LLMs' weights (Izacard et al., 2023; Rubin and Berant, 2023; Guu et al., 2020). However, as LLMs grow in size, this method becomes increasingly computationally demanding and time-consuming (Liu et al., 2023; Bai et al., 2024). Additionally, it is incompatible with closed-source LLMs, which do not share their parameters.

Recently, researchers have turned to integrating black-box LLMs for distillation training the retrievers. Shi et al. (2023) propose REPLUG that directly

[1]The code will be released after acceptance.

uses the generation log probabilities of correct answers as the distillation signal to train the retriever. However, this method may suffer from some inherent weaknesses: 1) Limited application scenarios, as the output probabilities are not always available for closed-source LLMs. 2) Discrepancy between retrieval and generation, where training based on LLMs' next-token prediction may not ideally suit information retrieval. 3) High computing costs, as it still requires hundreds of thousands of training instances.

To address those aforementioned limitations, we propose *Intermediate Distillation*, a data-efficient training scheme that utilizes ranking responses generated by LLMs to guide the training of retrievers. Our training operates on a *rerank-then-retrieve* pipeline, where LLMs indirectly influence retriever training via an intermediate ranker model. We chose this pipeline for three main reasons: 1) There are no restrictions on accessing this generated ranking order. 2) LLMs' robust zero-shot ranking capabilities establish a strong foundation for knowledge distillation. Furthermore, using LLMs to generate a relevance-based ranking order is closer and more suitable for retriever training than depending on LLMs' output probabilities, making the supervision signals more reliable. 3) The overhead on computational resources has been greatly reduced.

Specifically, we begin by training a ranker model using the re-ranked orders of initially retrieved documents, determined by LLMs, as supervision signals. We then employ this trained ranker to further distill the retriever model by aligning their judgments with the relevance assessments between the input query and the retrieved information. We conduct a series of experiments using advanced, closed-source LLMs that restrict access to output probabilities. The empirical results demonstrate the effectiveness of our method, requiring 100 times less data than previous method (Shi et al., 2023), thereby significantly reducing computational costs. Our main contributions are:

- We introduce Intermediate Distillation, a data-efficient knowledge distillation scheme that optimizes retrieval models from black-box LLMs via an intermediate ranker model in a two-stage straightforward training process.
- We conduct extensive experiments with cutting-edge LLMs and demonstrate the efficacy and efficiency of the proposed method in enhancing information retrieval performance compared to other supervision signals and previous related work.
- We deploy our distilled retriever model within the RAG framework and demonstrate its effectiveness in open-domain question-answering downstream tasks.

2 Related Work

2.1 Retrieval-Augmented Generation

Information retrieval plays a crucial role in various knowledge-intensive NLP tasks, including question-answering (Siriwardhana et al., 2023; Zhang et al., 2024), fact-verification (Hang et al., 2024; Khaliq et al., 2024) and open-domain dialogue (Wang et al., 2024a; Shuster et al., 2021). A prevalent approach in information retrieval is the multi-stage retrieval process (Nogueira et al., 2020), which utilizes a retriever model to select the most relevant documents from a large corpus and a ranker model to refine the ranking and finally return the top documents.

Recently, the retriever models are increasingly used to enhance the generation quality of language models for knowledge-intensive tasks due to their flexibility and effectiveness, leading to the development of the retrieval-augmented generation (RAG) framework (Guu et al., 2020; Izacard et al., 2023; Borgeaud et al., 2022; Ram et al., 2023). This framework enhances the language model's generation and addresses its limitations like hallucination through the use of external, up-to-date retrieved information. Meanwhile, it is applicable during both the pre-training (Zhong et al., 2022; Min et al., 2022) and inference stages (Menick et al., 2022; Min et al., 2023).

In this paper, we use the advanced knowledge from LLMs as the supervision signal to train the retriever models through a two-stage (i.e., rerank-then-retrieve) training scheme. We then integrate our well-trained retriever model into the RAG framework to demonstrate its effectiveness in question-answering downstream tasks.

2.2 Knowledge Distillation in LLMs.

Knowledge distillation is widely used to transfer knowledge from complex, large teacher models to smaller student models (Hinton et al., 2015). Driven by the superior performance of LLMs, increasing research is employing them as teachers to distill knowledge into smaller, task-specific models (Brown et al., 2023). The distillation meth-

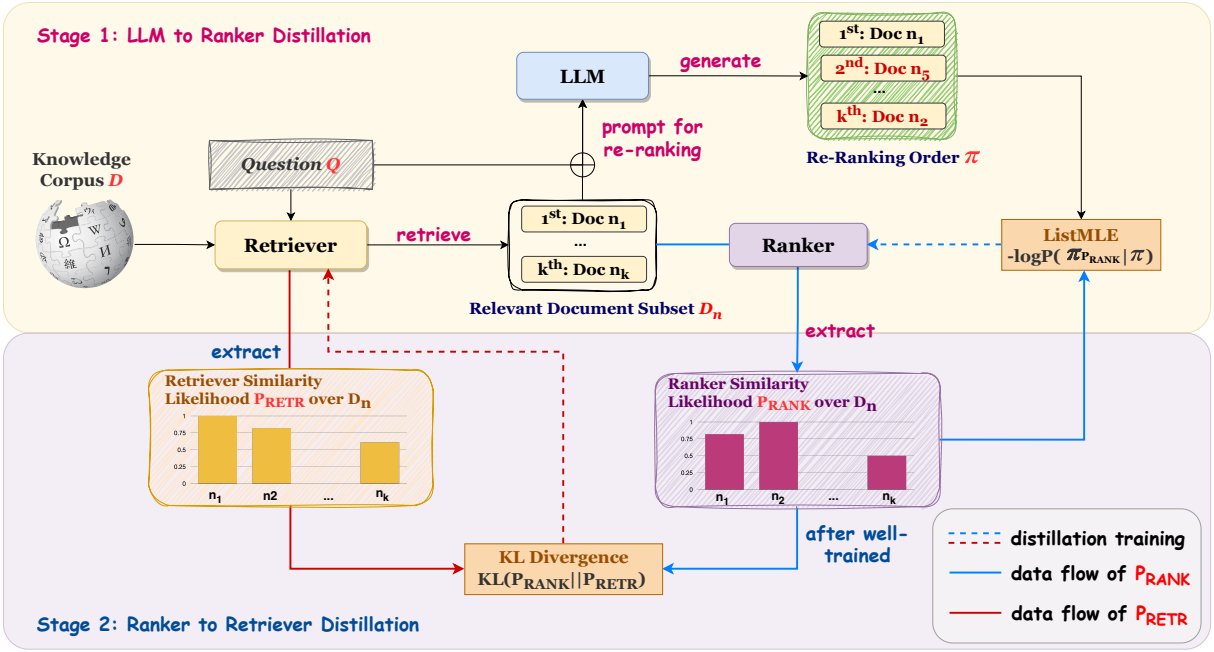

Figure 2: The two-stage knowledge distillation process of our proposed Intermediate Distillation scheme. In Stage 1, we use re-ranking order π (highlighted in the green background color) as the supervisory signal to train a ranker model. In Stage 2, this distilled ranker unsupervised trains the retriever model to enhance its performance.

ods can be categorized into two types: white-box (Gu et al., 2023; Agarwal et al., 2023; Udagawa et al., 2023) and black-box (Li et al., 2022; Ho et al., 2022; Hsieh et al., 2023; Dai et al., 2022; Lee et al., 2024). Specifically, white-box training utilizes both the generated tokens and the parameters of LLM for supervised signals, which can be memory-intensive and computationally demanding. Conversely, black-box training depends solely on LLM-generated tokens, making it less resource-intensive.

Many studies have successfully integrated knowledge distillation within the RAG framework to train the retriever models. Previous researches employ LLM likelihood, such as attention scores, to assess the relevance distribution of retrieved documents (Izacard et al., 2023, 2022) in the white-box LLM distillation. Meanwhile, recent work has also explored methods for training RAG using black-box LLMs, such as RePLUG (Shi et al., 2023). The remaining problem is that it still relies on the generation log probabilities for the ground truth as the supervision signals in knowledge distillation, which tends to have a degree of randomness and is limited to the availability of the output probabilities. Furthermore, there are better choices than aligning LLM token predictions with the goals of retriever training since there remains a gap between retrieval and generation.

In contrast, our proposed distillation scheme only requires black-box LLMs to output relevance-based re-ranking orders of the initial retrieved documents, easily yielding more consistent and interpretable results than output probability-based methods. Moreover, our method is much more data efficient, requiring about 100 times less data compared to the previous approach (Shi et al., 2023), significantly saving computational resources and increasing the flexibility of the training process.

3 Method

Our two-stage distillation scheme uses a ranker model and a retriever model as the student models and an LLM as the teacher model. As shown in Figure 2, we initially employ an off-the-shelf retriever to select a subset of documents D_n from a large corpus D based on their relevance to a query Q. The LLM then re-ranks these documents, creating a ranking order π, which is used to train the ranker model in the distillation Stage 1. In Stage 2, this ranker enhances the original retriever by minimizing the KL divergence between their similarity likelihood. In detail, Section 3.1 provides the formal definitions of the related tasks. Section 3.2 and Section 3.3 show how knowledge is directly transferred from LLMs to a ranker model and then further conveyed to a retriever model, respectively.

3.1 Problem Formulation

Given a question Q, the goal of a retriever model is to find a subset of the most relevant documents $D_n = \{n_1, n_2, ..., n_k\} \subseteq D$ from a large knowledge corpus $D = \{d_1, d_2, ..., d_m\}$, where each d_i, n_i represents a unique document. In the retrieval-augmented generation (RAG) framework, this subset D_n is combined with the question Q to form the reader's prompt input and then generates the corresponding answer A.

For the re-ranking task in our distillation framework, the teacher ranker model (i.e., LLMs) is tasked with reordering the documents D_n according to their relevance to the question Q. The re-ranking order can be represented as π : $T_{sim}(n_i, Q) > T_{sim}(n_k, Q) > ... > T_{sim}(n_j, Q)$, indicating the descending order of relevance from the documents $n_i, n_j, ..., n_k$ in the subset D_n. This order information π is first transferred to the ranker model, which serves as an intermediary between the LLM and the retriever model. Subsequently, the ranker model conveys this knowledge to the retriever, thereby enhancing its performance.

3.2 Stage 1: LLM to Ranker Distillation

The first step of our knowledge distillation workflow is index initialization, where we find the relevant document subsets D_n from corpus D for each specified question Q. These subsets then serve as the input for the Stage 1 training. In practice, we employ a widely-used information retrieval model, Contriever (Izacard et al., 2022), as the retriever model for data initialization.

LLM Re-ranking. In this stage, we utilize LLM's guaranteed zero-shot ranking capabilities to generate high-quality re-ranking orders for each subset D_n based on the relevance to the corresponding question Q. We use a list-wise ranking prompt, which is adapted from the RankGPT (Sun et al., 2023), as our input ranking prompt. Figure 3 illustrates the LLM re-ranking process, where the LLM generates a re-ranking order $\pi_{D_n} : T_{sim}(n_1, Q) > T_{sim}(n_5, Q) > T_{sim}(n_4, Q) > T_{sim}(n_3, Q) > T_{sim}(n_2, Q)$. Following this, we use these re-ranking orders to transfer LLM's knowledge into a smaller but more efficient ranker model.

Ranker Distillation Training We initialize our ranker model by using the dual-encoder structure Contriever checkpoint. For each question Q and its retrieved document $n_i \subseteq D_n$, we represent the question and document using the average value from the last hidden layer of the ranker model, denoted as $\hat{Q}$ and $\hat{n}_i$, respectively. For each training data instance, once we have the representative embeddings $\hat{Q}$ and $\hat{n}_i$, we proceed to calculate the similarity likelihood P_{RANK} over D_n, which can be defined as:

$$P_{RANK}(n_i|Q) = \frac{exp(s(\hat{n}_i, \hat{Q})/\theta)}{\sum_{k=1}^{K} exp(s(\hat{n}_k, \hat{Q})/\theta)} \quad (1)$$

where s denotes the dot-product between the ranker's representation vectors of the question and each retrieved document, and θ is the temperature hyper-parameter.

After obtaining the ranker model's similarity likelihood $P_{RANK}(n_i|Q)$ over its relevant document subset D_n, and the LLM-generated re-ranking order π_{D_n}, we use ListMLE (Xia et al., 2008), a list-wise loss function for distillation training from LLMs to the ranker model. ListMLE considers the similarity likelihood $P_{RANK}(D_n)$ as the predicted list and π_{D_n} as the ground truth, aiming to minimize the following loss function:

$$\begin{aligned} L(P_{RANK}, \pi) &= -logP(\pi_{P_{RANK}}|\pi) \\ &= -log \prod_{j=1}^{k} \frac{exp(P_{RANK}(n_i))}{\sum_{m=j}^{k} exp(P_{RANK}(n_m))} \end{aligned} \quad (2)$$

where $\pi_{P_{RANK}}$ represents the permutation of documents ordered by the ground truth ranking π. The loss function calculates an exponential probability distribution over all elements of $P(\pi_{P_{RANK}}|\pi)$, expressing the loss as the negative log-likelihood of the ground truth order π.

3.3 Stage 2: Ranker to Retriever Distillation

In Stage 2, this well-trained ranker model is used to enhance the retriever model's performance by transferring knowledge from LLMs.

We initialize our retriever model using a dual-encoder Contriever checkpoint, similar to the ranker. For each question and its retrieved document, we also compute their representations $\widetilde{Q}$ and $\widetilde{n}_i$ by calculating the average value from the retriever model's last hidden layer and the similarity distribution P_{RETR} of the retriever model over D_n is defined as:

$$P_{RETR}(n_i|Q) = \frac{exp(s(\widetilde{n}_i, \widetilde{Q})/\theta)}{\sum_{k=1}^{K} exp(s(\widetilde{n}_k, \widetilde{Q})/\theta)} \quad (3)$$

where s represents the dot-product between the retriever's representation vectors of the question and

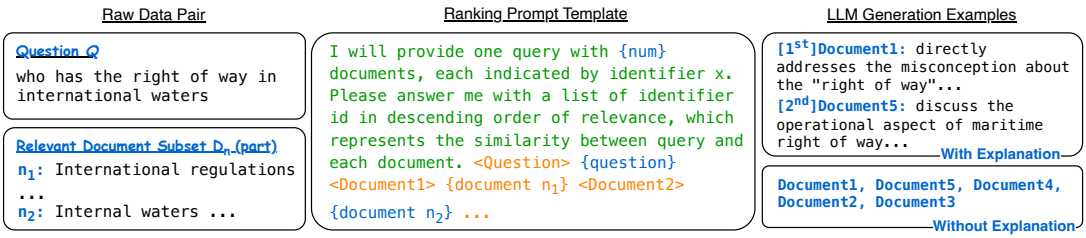

Figure 3: An example of our LLM teacher's re-ranking process.

each retrieved document, and θ is the temperature hyper-parameter.

We then leverage the similarity likelihood P_{RANK} from the previously trained ranker model to enhance the retriever model's performance under an unsupervised learning process by minimizing the KL-divergence between P_{RANK} and P_{RETR}, which is defined as $D_{KL}(P_{RANK}||P_{RETR})$.

This process ensures the retriever model aligns more closely with the text similarity knowledge from the LLMs. Through this two-stage distillation scheme, we enhance the retrieval accuracy and effectiveness of the retriever model, which can be further applied to the RAG framework to improve its performance on knowledge-intensive NLP tasks.

4 Experiment

4.1 Experiment Setup

Datasets We conduct experiments on four open-domain question-answering benchmark datasets: NaturalQuestions (NQ) (Kwiatkowski et al., 2019), TriviaQA (Joshi et al., 2017), PopQA (Mallen et al., 2022), and ASQA (Stelmakh et al., 2022). For the knowledge corpus base, we utilize data from Wikipedia as of December 20, 2018, adapting the passage embeddings provided by ATLAS (Izacard et al., 2023). We selectively sample 1,000 instances from each dataset's training set for retriever training, choosing the cases where the initial retriever can not reveal its effectiveness while keeping the validation and testing sets unchanged. This training set size is about 100 times smaller than those used in REPLUG (Shi et al., 2023).

Evaluation Metrics We evaluate the retrieval performance of our distilled retriever model through the *top-5* and *top-10* retrieval Hit Rates (HR@5 and HR@10), which is the percentage of questions where the relevant document subset D_n includes at least one correct answers with the top-5 and top-10 retrieved documents. Moreover, to comprehensively evaluate the retriever performance, we integrate the distilled retriever model into the RAG framework to conduct the question-answering (QA) downstream tasks, which allows us to measure the improvement in the quality of responses generated by the language model. We report the standard Exact Match (EM) value and F1-Score to show the accuracy and precision of the generated responses.

Models For the retriever training phase, we initialize our ranker and retriever models using the Contriever checkpoint (Izacard et al., 2022) with a dual-encoder structure. We test our Intermediate Distillation on different cutting-edge and representative LLMs, including the open-source Llama3 model (70B-Instruct) (AI@Meta, 2024), and the close-source GPT (3.5 Turbo, 4o, 4 Turbo) (OpenAI, 2024) and Claude (Sonnet and Opus) (Anthropic, 2024) models. For the QA downstream tasks, we use the off-the-shelf Llama3 (8B-Instruct) (AI@Meta, 2024) as the reader model within RAG to generate the answers.

Baseline We evaluate our Intermediate Distillation scheme against the following established text similarity methods to serve as the baseline distillation signals in ranker training: (1) BM25, a popular information retrieval algorithm; (2) ROUGE-2. The idea of employing such NLP evaluation metrics for knowledge distillation in training retriever models is first proposed by (He et al., 2022), which also uses a multi-step distillation approach to solve the Commonsense Reasoning tasks [2]; (3) Greedy Ranking, a rule-based method that ranks the answer-containing relevant documents to the top of the re-ranking order. We want to use this baseline to demonstrate that effective re-ranking distillation signals should not only highlight answers. For BM25 and ROUGE-2, we use them to calculate the similarity likelihood between the query and its relevant documents separately in order to form the corresponding re-ranking order that serves as

[2] We choose ROUGE-2 as our compared baseline metric, as it outperforms other metrics in this prior study.

Distillation Methods	NQ				TriviaQA				PopQA				ASQA			
	HR@5↑	HR@10↑	EM↑	F1↑	HR@5↑	HR@10↑	EM↑	F1↑	HR@5↑	HR@10↑	EM↑	F1↑	HR@5↑	HR@10↑	EM↑	F1↑
w/o Distillation	0.478	0.583	26.09	36.75	0.595	0.678	54.99	63.55	0.358	0.420	22.52	26.34	0.388	0.486	18.75	29.45
Rule Based Distillation Baseline																
BM25	0.186	0.262	18.17	27.88	0.120	0.175	46.90	55.38	0.043	0.071	17.99	22.60	0.359	0.459	26.57	37.67
Greedy Ranking	0.223	0.303	19.75	29.64	0.277	0.356	50.08	58.45	0.201	0.252	24.36	28.23	0.324	0.408	24.91	35.29
ROUGE-2	0.534	0.643	27.76	38.16	0.641	0.716	56.17	64.92	0.378	0.448	31.57	35.79	0.437	0.530	30.43	41.23
LLM distillation Baseline																
REPLUG	0.017	0.030	14.74	24.76	0.009	0.018	46.27	54.53	0.009	0.013	18.60	23.18	0.016	0.027	15.30	25.16
Intermediate Distillation (ours)																
Llama3-70b	0.548	0.649	27.31	37.59	0.649	0.720	54.25	63.28	0.407	0.478	32.51	36.87	0.486	0.577	30.65	41.47
GPT-3.5 Turbo	0.505	0.606	25.84	36.13	0.587	0.664	53.72	62.16	0.375	0.447	30.86	35.31	0.406	0.505	28.55	39.19
GPT-4o	0.553	0.652	27.01	37.38	0.664	0.734	56.27	64.98	0.446	0.512	35.41	**40.05**	0.489	**0.584**	31.25	41.81
GPT-4 Turbo	0.545	0.656	28.31	38.68	0.662	0.727	56.15	65.07	**0.464**	**0.541**	**35.49**	**40.05**	0.482	0.576	31.44	42.07
Claude Sonnet	0.554	0.658	**28.56**	38.58	**0.675**	**0.738**	55.04	63.77	0.398	0.459	31.24	35.63	0.486	0.577	31.19	42.04
Claude Opus	**0.562**	**0.665**	28.45	**38.83**	0.669	0.733	**56.68**	**65.36**	0.409	0.480	33.23	37.56	**0.490**	**0.584**	**32.44**	**43.28**

Table 1: The performance comparison of our proposed Intermediate Distillation scheme with other baseline distillation methods on question-answering tasks.

the supervision signals. For Greedy Ranking, we do not change the ordering of the remaining documents. In addition, we use the previous related work REPLUG (Shi et al., 2023) as the baseline retriever training method[3]. Specifically, we use GPT-3.5 Turbo as the LLM distillation teacher. To better show the results change after each baseline training, we also evaluate a baseline version of the RAG framework without additional retriever training (i.e., *w/o Distillation*). For all the baseline training, we maintain the consistency of our training sets, including their size and type.

Implantation Details We set both the ranker and retriever models with a hidden layer size of 768, thus having approximately 10 million training parameters for each model. The learning rates are set as 5e-5 for the ranker and 2e-5 for the retriever. Both models are trained for five epochs on four QA datasets, using a batch size of 20, and optimized with the Adam optimizer. Additionally, we restrict the size of the relevant document subset D_n to 5, each retrieved document with a maximum length of 128.

4.2 Main Results

We present our experimental results, including all the baseline methods and settings evaluated on the testing set of NQ, TriviaQA, PopQA, and ASQA in Table 1[4]. The results show that the retriever model, under our proposed Intermediate Distillation scheme and supervised by LLMs, outperforms all of the baseline distillation signals, confirming the effectiveness of our proposed training scheme. At the same time, the results show that REPLUG is unsuitable for small-scale data training (i.e., thousands level), leading to a sharp decline in retriever performance. The comparison of our results with those of the REPLUG reveals the data efficiency of our proposed scheme.

Additionally, different supervision signals lead to different training qualities. Supervising the retriever with BM25 or Greedy Ranking significantly impairs its performance, whereas metric-based (i.e., ROUGE-2) and LLM-based methods enhance its retrieval capabilities.

Moreover, the quality of supervision signals from LLMs greatly influences the performance of the distilled retriever models. With the rapid development of LLMs, the improvement in supervision quality has also evolved: from being less effective than the ROUGE-2 metric (i.e., as seen in the distilled retriever under supervision by GPT-3.5 Turbo) to significantly surpassing it (i.e., the distilled retriever under supervised by GPT-4 Turbo).

For the QA tasks within the RAG framework, the results show that a more robust retriever model is more likely to enhance the accuracy and precision of generated answers, demonstrating the effectiveness and adaptability of our proposed scheme for NLP downstream tasks. However, while Intermediate Distillation using GPT-4o typically outperforms Supervised Distillation using ROUGE-2 in retriever performance, the latter can still produce slightly higher quality generations within RAG. This divergence may be due to the different objectives of the retriever and the reader within the RAG framework: the retriever focuses on accurately identifying the ground truth, whereas the reader wants the retriever to provide information that more effectively helps the reader generate accurate responses.

[3]Since REPLUG's original code can not run successfully, we reimplement it at REPLUG-reimplementation.

[4]The highest values in the table are highlighted in bold

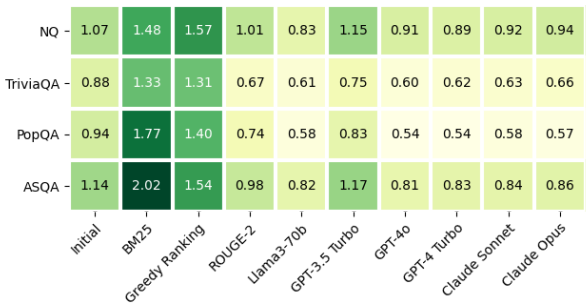

Figure 4: Average answer position of different intermediate ranker models, the first place starts from zero.

5 Analysis

5.1 Performance comparison of the intermediate ranker model

To further validate the ranking hypothesis (i.e., the intermediate ranker model can effectively convey supervision signals from LLMs to the retriever model), We analyze and report the initial ranking and the re-ranking of the same documents by the ranker models trained with different supervision signals, as shown in the Figure 4. Here, we use the average answer position value as the evaluation metric[5] to reveal the changes of the intermediate ranker, as the following shows

$$Avg.AnsPosition = \frac{min(p_i, p_j, ..., p_m)}{n} \quad (4)$$

where $p_i, p_j, \cdots, p_n$ represent the ranking position of the retrieved document that contains the correct answer, and n represents the total number of retrieved documents.

According to Figure 4, we can see that compared to the *initial* ranker's ranking results, different training supervision signals have varying impacts on the ranker: BM25 and Greedy Ranking degrade the ranker's performance, causing the ranker to re-rank target documents (i.e., documents contain the correct answer) to more backward positions, whereas metric-based and LLM-based methods help the ranker better understand the similarity between queries and retrieved documents. These results show a strong correlation with the training results of the retriever: the stronger the re-ranking capability of the intermediate ranker models, the more effective the supervision signals it provides for the next stage of retriever training. Conversely, if the ranker's re-ranking ability is impaired, it will negatively impact the training of the retriever, which underscores the importance of the intermediate ranker in the training scheme.

[5]It should be noted that this metric is not the most comprehensive way to measure the ranker's performance, but it is the most intuitive way.

Method	Dataset	Evaluation Metrics		
		EM↑	F1↑	HR@5↑
Direct Distillation	NQ	26.23	36.47	0.505
	TriviaQA	55.39	63.96	0.623
	PopQA	33.70	37.89	0.414
	ASQA	29.02	39.84	0.441
Intermediate Distillation	NQ	27.01	37.38	0.553
	TriviaQA	56.27	64.98	0.664
	PopQA	35.41	40.05	0.446
	ASQA	31.25	41.81	0.489

Table 2: Ablation studies on the effectiveness of two-stage distillation scheme design.

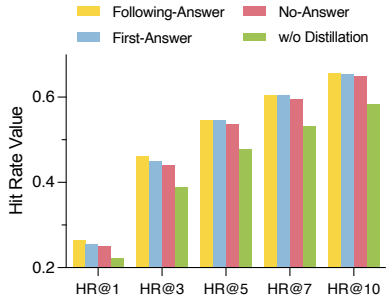

Figure 5: The performance of retriever models varies across three types of training sets, which differ based on the initial appearance and placement of ground truth in the retrieved subsets.

5.2 Performance comparison between the direct and intermediate distillation

To further verify the rationality and effectiveness of our proposed two-stage training scheme, we conduct the *Direct Distillation* experiment, where we train the retriever model *directly* using the relevance likelihood generated by LLM. More details of this experimental setting can be found in Appendix A. We compare the results of this approach with our proposed two-stage distillation scheme under the same LLM teacher model GPT-4o, and the experimental results are shown in Table 2. These results indicate that the Direct Distillation method is less effective than our proposed Intermediate Distillation scheme.

5.3 Impact of the training data type

The ineffective of the Greedy Ranking baseline indicates that simply re-ranking documents based solely on the position of the correct answer does not provide the high-quality text similarity insights required for effective distillation. To delve deeper into the influence of the appearance and placement of ground truth in the re-ranking process, we categorize the initial retrieved document subsets D_n

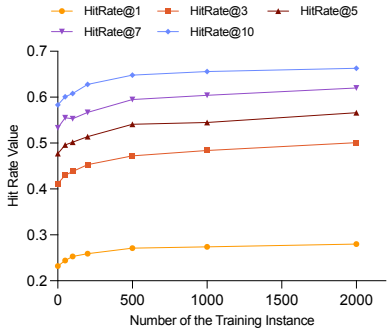

Figure 6: The performance of retriever models under different training set sizes, following a scaling law.

based on *NQ*'s queries into three categories: (1) *Following-Answer*: contains at least one document with the correct answer, which does not rank first in the subset. We use this data type as the training data in Section 4. (2) *First-Answer*: contains at least one document with the correct answer, which ranks first in the subset. (3) *No-Answer*: no documents in the subset contain the correct answer.

We follow the same training setting used in our main experiments and use GPT-4 Turbo as the LLM teacher model. The experiment results shown in Figure 5 indicate that considering the semantic similarity of the given text is very important, as even the retriever under the *No-Answer* data has notable improvements. Moreover, the *Following-Answer* data yields better training results than using the *First-Answer* data in training, which indicates that optimizing the ground-truth position in re-ranking also has a positive effect on the experimental results after the consideration of text similarity.

5.4 Impact of the training data size

To further explore how different training set sizes affect distillation effectiveness, we use training sets of 50, 100, 200, 500, 1000, and 2000 data instances from the *Following-Answer* data type and remain other settings unchanged. The experiment results are shown in Figure 6. According to the results, the performance of the retriever model improves significantly with thousands of or even only hundreds of training instances, which highlights the data efficiency of our proposed distillation scheme. In addition, although initial performance increases are notable with small training sets, the rate of improvement decreases as more training data is used. This pattern indicates a *scaling law* in distillation training, where further enhancements become increasingly difficult as the model's performance improves. For models that already perform well, even marginal improvements require much more data, demanding greater training resources.

5.5 Impact of the re-ranking list size

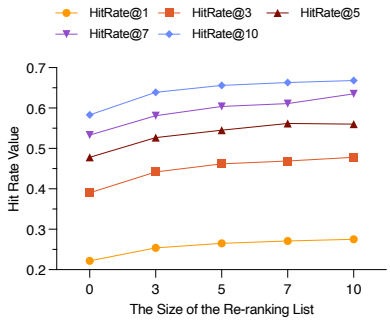

Figure 7: The performance of retriever models under different size of the re-ranking list. The performance corresponding to 0 re-ranking list size represents the baseline retriever model performance.

In previous experiments, we set the re-ranking list length to five. Generally, larger re-ranking lists offer more supervision signals from LLMs, thus potentially enhancing the effectiveness of retriever training. To explore the impact of re-ranking list size on our proposed scheme, we vary the re-ranking list sizes, using the top-3, top-5, top-7, and top-10 documents from each relevant retrieved subset to conduct the training and keep other settings unchanged.

The experimental results shown in Figure 7 reveal that increasing the re-ranking list size progressively improves the effectiveness of the distillation training: as the list expands from re-ranking three documents to ten documents, the performance of the distilled retriever model consistently improves. Moreover, compared with the baseline performance of the retriever, setting the size of the re-ranking list to three still significantly enhances the model's performance, not only in HitRate@3 but also across broader metrics from HitRate@5 to HitRate@10.

6 Conclusion

We propose Intermediate Distillation, a two-stage data-efficient knowledge distillation scheme that uses the remarkable capabilities of black-box LLMs to train an information retrieval model through an intermediate ranker model. We conduct extensive experiments with advanced LLMs, demonstrating that our method enhances effectiveness and efficiency compared to other comparative supervision baselines.

7 Limitation

This paper proposes a data-efficient distillation scheme using black-box LLMs to train smaller information retrieval models, which prove its effectiveness with training data on the scale of thousands. However, we do not evaluate our proposed distillation scheme with larger scales of training data, such as tens of thousands or millions of instances, due to budget limitations on accessing responses from closed-source LLMs. Future work could focus on extending this study to larger-scale training data, using either closed-source or advanced open-source LLMs to further analysis the effectiveness of such multi-stage distillation scheme.

References

Mofetoluwa Adeyemi, Akintunde Oladipo, Ronak Pradeep, and Jimmy Lin. 2023. Zero-shot cross-lingual reranking with large language models for low-resource languages. *arXiv preprint arXiv:2312.16159*.

Rishabh Agarwal, Nino Vieillard, Piotr Stanczyk, Sabela Ramos, Matthieu Geist, and Olivier Bachem. 2023. Gkd: Generalized knowledge distillation for auto-regressive sequence models. *arXiv preprint arXiv:2306.13649*.

AI@Meta. 2024. Llama. https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

Anthropic. 2024. Claude. https://www.anthropic.com/.

Guangji Bai, Zheng Chai, Chen Ling, Shiyu Wang, Jiaying Lu, Nan Zhang, Tingwei Shi, Ziyang Yu, Mengdan Zhu, Yifei Zhang, et al. 2024. Beyond efficiency: A systematic survey of resource-efficient large language models. *arXiv preprint arXiv:2401.00625*.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. 2022. Improving language models by retrieving from trillions of tokens. In *International conference on machine learning*, pages 2206–2240. PMLR.

Nathan Brown, Ashton Williamson, Tahj Anderson, and Logan Lawrence. 2023. Efficient transformer knowledge distillation: A performance review. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 54–65, Singapore. Association for Computational Linguistics.

Zhuyun Dai, Vincent Y Zhao, Ji Ma, Yi Luan, Jianmo Ni, Jing Lu, Anton Bakalov, Kelvin Guu, Keith B Hall, and Ming-Wei Chang. 2022. Promptagator: Few-shot dense retrieval from 8 examples. *arXiv preprint arXiv:2209.11755*.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. 2023. Minillm: Knowledge distillation of large language models. In *The Twelfth International Conference on Learning Representations*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.

Ching Nam Hang, Pei-Duo Yu, and Chee Wei Tan. 2024. Trumorgpt: Query optimization and semantic reasoning over networks for automated fact-checking. In *2024 58th Annual Conference on Information Sciences and Systems (CISS)*, pages 1–6. IEEE.

Xingwei He, Yeyun Gong, A Jin, Weizhen Qi, Hang Zhang, Jian Jiao, Bartuer Zhou, Biao Cheng, Siu Ming Yiu, Nan Duan, et al. 2022. Metric-guided distillation: Distilling knowledge from the metric to ranker and retriever for generative commonsense reasoning. *arXiv preprint arXiv:2210.11708*.

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

Namgyu Ho, Laura Schmid, and Se-Young Yun. 2022. Large language models are reasoning teachers. *arXiv preprint arXiv:2212.10071*.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. *arXiv preprint arXiv:2305.02301*.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised dense information retrieval with contrastive learning. *Preprint*, arXiv:2112.09118.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2023. Atlas: Few-shot learning with retrieval augmented language models. *Journal of Machine Learning Research*, 24(251):1–43.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

M Abdul Khaliq, P Chang, M Ma, Bernhard Pflugfelder, and F Miletić. 2024. Ragar, your falsehood radar: Rag-augmented reasoning for political fact-checking using multimodal large language models. *arXiv preprint arXiv:2404.12065*.

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.

Jinhyuk Lee, Zhuyun Dai, Xiaoqi Ren, Blair Chen, Daniel Cer, Jeremy R Cole, Kai Hui, Michael Boratko, Rajvi Kapadia, Wen Ding, et al. 2024. Gecko: Versatile text embeddings distilled from large language models. *arXiv preprint arXiv:2403.20327*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Shiyang Li, Jianshu Chen, Yelong Shen, Zhiyu Chen, Xinlu Zhang, Zekun Li, Hong Wang, Jing Qian, Baolin Peng, Yi Mao, et al. 2022. Explanations from large language models make small reasoners better. *arXiv preprint arXiv:2210.06726*.

Yiheng Liu, Tianle Han, Siyuan Ma, Jiayue Zhang, Yuanyuan Yang, Jiaming Tian, Hao He, Antong Li, Mengshen He, Zhengliang Liu, et al. 2023. Summary of chatgpt-related research and perspective towards the future of large language models. *Meta-Radiology*, page 100017.

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*.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Hannaneh Hajishirzi, and Daniel Khashabi. 2022. When not to trust language models: Investigating effectiveness and limitations of parametric and non-parametric memories. *arXiv preprint arXiv:2212.10511*, 7.

Jacob Menick, Maja Trebacz, Vladimir Mikulik, John Aslanides, Francis Song, Martin Chadwick, Mia Glaese, Susannah Young, Lucy Campbell-Gillingham, Geoffrey Irving, and Nat McAleese. 2022. Teaching language models to support answers with verified quotes. *Preprint*, arXiv:2203.11147.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *Preprint*, arXiv:2305.14251.

Sewon Min, Weijia Shi, Mike Lewis, Xilun Chen, Wen-tau Yih, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Nonparametric masked language modeling. *arXiv preprint arXiv:2212.01349*.

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

OpenAI. 2023. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

OpenAI. 2024. Gpt. https://chat.openai.com/.

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

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *Transactions of the Association for Computational Linguistics*, 11:1316–1331.

Ohad Rubin and Jonathan Berant. 2023. Long-range language modeling with self-retrieval. *arXiv preprint arXiv:2306.13421*.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2023. Replug: Retrieval-augmented black-box language models. *arXiv preprint arXiv:2301.12652*.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. *arXiv preprint arXiv:2104.07567*.

Shamane Siriwardhana, Rivindu Weerasekera, Elliott Wen, Tharindu Kaluarachchi, Rajib Rana, and Suranga Nanayakkara. 2023. Improving the domain adaptation of retrieval augmented generation (rag) models for open domain question answering. *Transactions of the Association for Computational Linguistics*, 11:1–17.

Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. 2022. Asqa: Factoid questions meet long-form answers. *arXiv preprint arXiv:2204.06092*.

Weiwei Sun, Lingyong Yan, Xinyu Ma, Pengjie Ren, Dawei Yin, and Zhaochun Ren. 2023. Is chatgpt good at search? investigating large language models as re-ranking agent. *arXiv preprint arXiv:2304.09542*.

Takuma Udagawa, Aashka Trivedi, Michele Merler, and Bishwaranjan Bhattacharjee. 2023. A comparative analysis of task-agnostic distillation methods for compressing transformer language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 20–31, Singapore. Association for Computational Linguistics.

Hongru Wang, Wenyu Huang, Yang Deng, Rui Wang, Zezhong Wang, Yufei Wang, Fei Mi, Jeff Z Pan, and Kam-Fai Wong. 2024a. Unims-rag: A unified multi-source retrieval-augmented generation for personalized dialogue systems. *arXiv preprint arXiv:2401.13256*.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, et al. 2024b. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):1–26.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. 2023. Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*.

Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwen Ding, Boyang Hong, Ming Zhang, Junzhe Wang, Senjie Jin, Enyu Zhou, et al. 2023. The rise and potential of large language model based agents: A survey. *arXiv preprint arXiv:2309.07864*.

Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. 2008. Listwise approach to learning to rank: theory and algorithm. In *Proceedings of the 25th international conference on Machine learning*, pages 1192–1199.

Ran Xu, Wenqi Shi, Yue Yu, Yuchen Zhuang, Yanqiao Zhu, May D Wang, Joyce C Ho, Chao Zhang, and Carl Yang. 2024. Bmretriever: Tuning large language models as better biomedical text retrievers. *arXiv preprint arXiv:2404.18443*.

Haopeng Zhang, Xiao Liu, and Jiawei Zhang. 2023a. Extractive summarization via chatgpt for faithful summary generation. *Preprint*, arXiv:2304.04193.

Haopeng Zhang, Xiao Liu, and Jiawei Zhang. 2023b. Summit: Iterative text summarization via chatgpt. *Preprint*, arXiv:2305.14835.

Tianjun Zhang, Shishir G Patil, Naman Jain, Sheng Shen, Matei Zaharia, Ion Stoica, and Joseph E Gonzalez. 2024. Raft: Adapting language model to domain specific rag. *arXiv preprint arXiv:2403.10131*.

Zexuan Zhong, Tao Lei, and Danqi Chen. 2022. Training language models with memory augmentation. *arXiv preprint arXiv:2205.12674*.

A Analysis of Different Distillation Signal Generated by LLMs

Here we provide more details about the prompt design of the *Direct Distillation* experiment. We also analyze the corresponding generation quality of LLMs, including the stability and interpretability, and compare it with the re-ranking generation.

A.1 Direct Distillation Prompt Design

We follow the re-ranking prompt format design and replace the re-ranking task with quantifying the similarity scores of the retrieved documents for LLM's supervised signals generation. An example input prompt and the generated responses without explanations is as follows:

Input Prompt:

```
I will provide one query with {num}
documents, each indicated by number
identifier x.
Please answer me with a list of the
similarity score between the provided
query and documents based on your judgment.
The score should be between 0-1.
Please don't use interoperator and only
output the score list.
<Question> {question}
<Document1> {document n1}
...
```

Generated Response:

```
[0.1, 0.1, ...., 0.8, 0.2]
```

A.2 Comparison with Re-ranking Response

We compare the responses generated from re-ranking prompts and those derived from similarity score prompts used in the Direct Distillation experiment. Specifically, we randomly select 1,000 data instances from the *NQ* dataset, using the queries and their retrieved documents to prompt the LLM to generate both list-wise re-ranking orders and similarity scores. Examples of responses generated by the LLM, specifically using GPT-4 Turbo, are shown in Table 3.

Table 3 shows that responses generated from re-ranking prompts are more interpretable than those from similarity score prompts, which often use scaled values that are ambiguous. Additionally, responses from the similarity score-based prompt frequently yield extreme values, such as 0.0 (completely dissimilar) or 1.0 (highly similar), in some cases, which means that that supervision signals based on similarity scores are less informative and act more like binary signals in certain data instances.

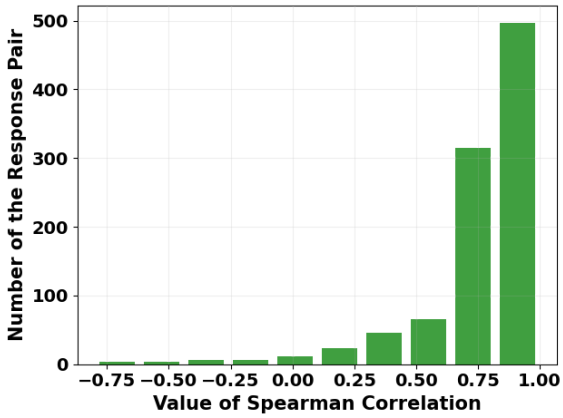

Figure 8: Spearman Correlation between the responses from the re-ranking prompt and the responses from the similarity score prompt.

Moreover, we use the *Spearman Correlation* to assess the consistency between responses generated from re-ranking prompts and those from similarity score prompts, and the analysis result is visualized in Figure 8. A higher Spearman correlation value suggests a stronger positive correlation between the two types of responses. From Figure 8, we can see that many response pairs are closely related, indicating the stability and reliability of LLMs in generating responses for similar tasks. In addition, based on our previous analysis, we can see that the responses from re-ranking prompts are not only reliable but also possess a higher information density compared to those from similarity score prompts, showing that responses from re-ranking prompts have higher-quality supervision capabilities.

A.3 Implantation Details of Direct Distillation Experiment

We use the same training settings as those used in Intermediate Distillation experiments. Superficially, we set the retriever model with a hidden layer size to 768 and initialize it using a dual-encoder Contriever checkpoint. The model is trained over five epochs using the same dataset as described in Section 4, with a learning rate of 5e-5 for NQ and TriviaQA datasets and 1e-5 for PopQA and ASQA datasets, a batch size of 20, and optimization via the Adam optimizer. In addition, we also limit the size of the relevant document subset D_n to five, and each retrieved document with a maximum length of 128.

Question & Retrieved Documents	Type	LLM's Generation
[Question] membership in the european union requires countries to have which type of government **[Document1]** the european continent. the eu's example is being followed by the african union, the union of south American nations... **[Document2]** powers of states, the considerable level of translation and the choice of a lesser city as the capital... **[Document3]** principal characteristic of classical democracy. the concept of a mixed government was studied during the renaissance and the age of reason by niccolo machiavelli, giambattista vico... **[Document4]** thus essentially preventing it from participating in government. the european union (eu) is an association of twenty - eight sovereign member states... **[Document5]** civiletti opinions, not all funding gaps led to shutdowns. of the nine funding gaps between 1980 and 1990, only four led to furloughs...	*Re-ranking*	[*Re-ranking order*] Document5, Document3, Document2, Document1, Document4. *[Explanation]* Document5 directly discusses the governmental systems used by most European nations, particularly the parliamentary system, which is relevant since the EU generally comprises nations with parliamentary democracies. This makes it the most relevant to the query about the type of government required for EU membership. Document3 talks about mixed constitutions...
	Similarity Score	[*Similarity scores from Document1 to Document5*] Doc1:0.0, Doc2:0.1, Doc3:0.3, Doc4:0.1, Doc5:0.7. *[Explanation]* Document1 talks about the EU and various other international organizations, focusing on aspects like the single market and policy integration but does not mention government types required for EU membership. Hence, the score is 0.0. Document2 discusses the structure of EU governance...
[Question] when was the immigration reform and control act passed **[Document1]** united states temporarily for money or a better life. an increase in legal immigrants migrating to the united states also led to an increase of illegal aliens... **[Document2]** there are noted communists, socialists, and anti - american sympathizers. parts of the act remain in place today... **[Document3]** ...the immigration reform and control act (irca), also known as the simpson 2013 mazzoli act or the reagan amnesty, signed into law by ronald reagan on november 6, 1986... **[Document4]** five percent of the u. s. workforce. it was also estimated that about 70 percent of those illegal workers... **[Document5]** to reduce them as restrictionists. "fry traced" restrictions roots " to a surprising surge in illegal and legal immigration...	*Re-ranking*	[*Re-ranking order*] Document3, Document1, Document4, Document5, Document2. *[Explanation]* Document3 directly answers the query by explicitly mentioning the "Immigration Reform and Control Act of 1986" and stating it was signed into law by Ronald Reagan on November 6, 1986. It provides the most specific information related to the query...
	Similarity Score	[*Similarity scores from Document1 to Document5*] Doc1:0.0, Doc2:0.0, Doc3:1.0, Doc4:0.0, Doc5:0.0. *[Explanation]* Document1 discusses general immigration trends and issues but does not provide the specific date of the passage of the Immigration Reform and Control Act, which is directly asked in the query. Therefore, it receives a score of 0.0. Although Document2 mentions other immigration-related legislation and acts, it does not specify...
[Question] who sings in there's something about mary **[Document1]** melody. mary described the completed song as "whitney - fied". meron maintains that... **[Document2]** ...he film concludes with the two engaging in a kiss while singer / songwriter jonathan richman, who has periodically narrated the story in song throughout the film... **[Document3]** day but her. minutes later bing notices his tackle box has been disturbed, and an instant later the kids see the door to the tent is open... **[Document4]** true opinions of other contestants and her stance on the judges. the book achieves in combining fifty years of a'salt of the earth'existence with a new - found life of fame... **[Document5]** and breezy number and the narrator opens up the story about bing and his family. and while bing is cleaning the fish, he tells the story of " goldilocks"...	*Re-ranking*	[*Re-ranking order*] Document2, Document5, Document4, Document1, Document3. *[Explanation]* Document2 is the most relevant sentence as it directly answers the query. It mentions that Jonathan Richman narrated parts of the story through song throughout the film "There's Something About Mary," which clearly identifies the singer related to your query...
	Similarity Score	[*Similarity scores from Document1 to Document5*] Doc1:0.0, Doc2:1.0, Doc3:0.0, Doc4:0.0, Doc5:0.0. *[Explanation]* Document1 discusses a song in a completely different context, related to "Cinderella" and Whitney Houston's music adjustments, not related to the film "There's Something About Mary." Thus, it doesn't answer the query about who sings in the specified film, receives a score of 0.0. Document2 directly answers the query. It mentions that Jonathan Richman narrated parts of the story...

Table 3: Examples of two different LLM-generated responses. We also let LLM generate the corresponding explanations in these examples. The red color indicates the corresponding answer to the question.