Understand What LLM Needs: Dual Preference Alignment for Retrieval-Augmented Generation

Guanting Dong
Yutao Zhu
Chenghao Zhang
dongguantinge@ruc.edu.cn
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China

Zechen Wang
Beijing University of Posts and Telecommunication
Beijing, China
shenshui@bupt.edu.cn

Ji-Rong Wen
Zhicheng Dou*
jrwen@ruc.edu.cn
dou@ruc.edu.cn
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China

Abstract

Retrieval-augmented generation (RAG) has effectively mitigated the hallucination problem of large language models (LLMs). However, the difficulty of aligning the retriever with the LLMs' diverse knowledge preferences inevitably poses a challenge in developing a reliable RAG system. To address this issue, we propose DPA-RAG, a universal framework designed to align diverse knowledge preferences within RAG systems. Specifically, we initially introduce a preference knowledge construction pipeline and incorporate five novel query augmentation strategies to alleviate preference data scarcity. Based on preference data, DPA-RAG accomplishes both external and internal preference alignment: 1) It jointly integrates pairwise, pointwise, and contrastive preference alignment abilities into the reranker, achieving external preference alignment among RAG components. 2) It further introduces a pre-aligned stage before vanilla Supervised Fine-tuning (SFT), enabling LLMs to implicitly capture knowledge aligned with their reasoning preferences, achieving LLMs' internal alignment. Experimental results across four knowledge-intensive QA datasets demonstrate that DPA-RAG outperforms all baselines and seamlessly integrates both black-box and open-sourced LLM readers. Further qualitative analysis and discussions provide empirical guidance for achieving reliable RAG systems. Our code and example dataset are available at https://github.com/dongguanting/DPA-RAG.

CCS Concepts

• **Information systems** → **Retrieval models and ranking**.

Keywords

Retrieval-Augmented Generation, Large Language Model

*Corresponding Author.

WWW '25, April 28-May 2, 2025, Sydney, NSW, Australia

ACM ISBN 979-8-4007-1274-6/25/04
https://doi.org/10.1145/3696410.3714717

ACM Reference Format:
Guanting Dong, Yutao Zhu, Chenghao Zhang, Zechen Wang, Ji-Rong Wen, and Zhicheng Dou. 2025. Understand What LLM Needs: Dual Preference Alignment for Retrieval-Augmented Generation. In *Proceedings of the ACM Web Conference 2025 (WWW '25), April 28-May 2, 2025, Sydney, NSW, Australia.* ACM, New York, NY, USA, 20 pages. https://doi.org/10.1145/3696410.3714717

1 Introduction

The emergence of large language models (LLMs) [1, 64, 66, 99, 100] has profoundly revolutionized a variety of real-world tasks expressed in natural languages [6, 49, 52, 95, 106]. However, when faced with knowledge-intensive tasks, relying solely on internal knowledge for reasoning may easily expose LLMs to factual inconsistency and hallucination [111]. To alleviate these issues, researchers use retrieval-augmented technology [18, 36] to assist LLMs in integrating relevant knowledge from the web (such as Wikipedia [89]) or other external knowledge bases, providing a promising solution to improve the quality of generated answers [71].

In an ideal retrieval-augmented generation (RAG) system, the goal is to enhance LLMs by incorporating supporting documents that align with their intrinsic knowledge preferences, thus facilitating reasoning. However, in practical applications, the retriever and the LLM-based reader serve as separate components within the RAG system, each with distinct model architectures, training objectives, and task formats [39]. These differences often result in documents retrieved by vector similarity failing to meet the specific knowledge demands for LLM reasoning. Moreover, retrieved documents could even conflict with the self-knowledge of LLMs, potentially disrupting LLMs' original reasoning abilities [11, 69].

As depicted in Figure 1, we perform a preliminary analysis on GPT-3.5 on three QA benchmarks, which compares two setups: LLM answering questions directly and answering questions by referencing different types of retrieved documents. We could categorize results into four distinct conditions: (1) **Both Correct**: the question can be resolved directly by the LLM or through the retrieved documents. (2) **Aligned Knowledge**: LLM gives the wrong answer, but the retrieved documents guide LLM to provide the right solution. (3) **Unaligned Knowledge**: LLM gives the right answer, but the retrieved documents may mislead it. (4) **Both Incorrect**: neither the retrieved documents nor the LLM can provide an answer correctly. Then we have the following observations: in the scenario of aligned knowledge, it is notable that documents with low vector

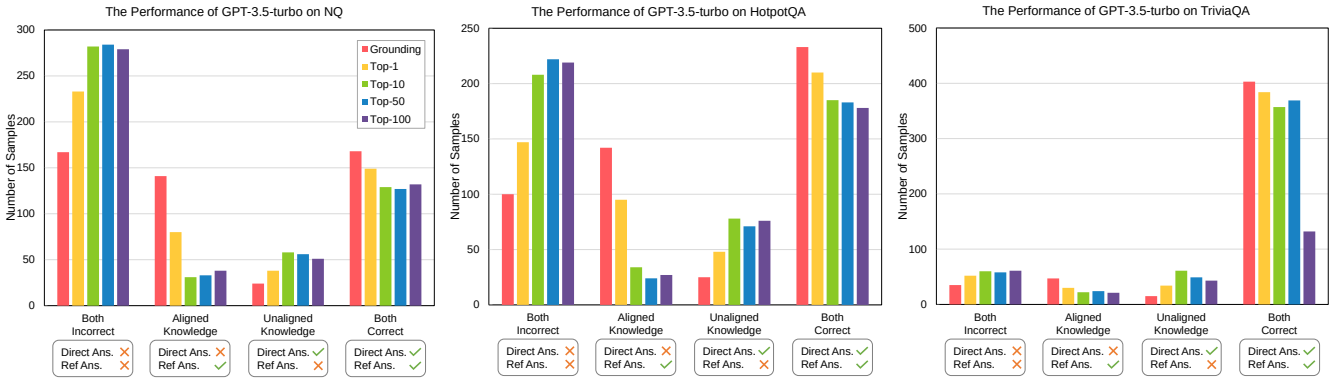

Figure 1: The results for GPT-3.5 comparing direct responses and answers referencing different retrieved documents (Grounding, 1st, 10th, 50th, 100th) on three QA benchmarks.

similarity (e.g., ranked 100th) still support the LLM in deducing correct answers. Conversely, within the unaligned knowledge scenario, several documents with high vector similarities tend to mislead LLM more than those with lower similarities (*e.g.*, 10th vs 100th). Surprisingly, even some documents that contain relevant grounding information struggle to align with the LLM's preferences [34]. These results highlight our statement that "The retrieved documents do not exactly match the knowledge required for LLM reasoning". Therefore, mitigating the preference gap between the LLM and the retriever emerges as a critical challenge in developing a reliable RAG system.

To address the above limitation, we propose a **D**ual **P**reference **A**lignment for **R**etrieval-**A**ugmented **G**eneration (**DPA-RAG**), a universal framework designed to align diverse preference knowledge within RAG systems. DPA-RAG consists of three key components: (1) **Preference Knowledge Construction**: motivated by our preliminary results in Figure 1, we first extract the specific knowledge that significantly affects LLMs' reasoning preferences. Then we introduce five query augmentation strategies and a quality filtering process to synthesize high-quality preference knowledge. (2) **Reranker-LLM Alignment**: To meet the diverse knowledge preferences of LLMs, we carefully design multi-grained alignment tasks for fine-tuning a preference-aligned reranker. Specifically, we jointly integrate pair-wise, point-wise, and contrastive preference alignment abilities into the reranker via multi-task optimization [79]. By this means, the reranker could provide the necessary knowledge for LLM's inference, achieving external alignment between the retriever and the LLM. (3) **LLM Self-Alignment**: To further enable LLMs to concentrate on knowledge aligned with their reasoning preferences, we introduce a pre-aligned phrase before the vanilla SFT stage. This stage allows the LLM to capture preference-aligned knowledge from multiple documents, completing the LLM's internal self-alignment.

To summarize, our contributions are as follows:

- Based on our quantitative analysis of GPT-3.5 across three QA benchmarks, we reveal the inherent preference gaps between the retriever and the LLM-based reader in RAG systems.
- We propose DPA-RAG, a universal framework designed to align the diverse knowledge preferences of LLMs within RAG systems. DPA-RAG achieves dual preference alignment in two aspects: (1) It jointly integrates multi-grained preference alignment abilities into the reranker, facilitating external alignment across RAG components. (2) It introduces a pre-aligned phrase prior to the standard SFT stage, guiding LLMs to concentrate on the aligned knowledge, thereby unlocking the internal alignment abilities of the LLMs.
- To overcome the scarcity and limited diversity of preference data, we devise five novel query augmentation strategies and a quality filtering process, aiming at automatically synthesizing high-quality preference data for effectively aligning downstream models.
- Experimental results on four knowledge-intensive QA datasets demonstrate the effectiveness of DPA-RAG. Further analysis across dimensions such as model parameters, preference alignment, data quality, and training strategies confirm DPA-RAG's role as a plug-and-play solution, providing practical insights for developing reliable RAG systems.

2 Related Work

2.1 Preference Alignment for LLMs

Traditional Preference alignment (PA) methodologies [17, 21, 23, 92] are designed to tailor pre-trained language models to reflect human preferences. Recently, a series of works have relied on reinforcement learning (RL) [78] to align LLMs with human preferences [66]. Owing to the sensitivity of RL's parameters and the complex process of reward modeling, research works [14, 44, 46, 47, 60, 82, 97, 105, 112] represented by DPO [73] further tried to optimize the loss function and reward scoring mechanism for pruning. However, depending on annotations from humans or expert models still increases the alignment cost. To construct reliable RAG systems, a branch of studies [4, 20, 81] aims to align the retriever with supervision signals generated by LLMs, showcasing remarkable alignment potential. Conversely, other studies attempt to improve the alignment abilities of RAG systems by implementing a multi-round retrieval paradigm [24, 75, 87, 90, 102, 115] and filtering out noise from the training corpus [26, 35, 40, 93, 94, 109, 110]. These approaches, however, often suffer from a lack of multi-level alignments, which

limits their ability to adapt to the diverse knowledge preferences of LLMs. In our paper, we introduce DPA-RAG, which bridges this gap without relying on external expert annotations.

2.2 Reranking Techniques for RAG

In the RAG system, the reranker is designed to rank a list of retrieved documents to accurately meet LLMs' demands. A series of sentence transformer models [25, 31, 63, 74, 98] have achieved excellent fine-grained ranking by better aligning the representations between queries and documents. With the rapid development of prompt learning [45], point-wise generative reranking frameworks [28, 62, 70, 116] have transformed traditional discriminative tasks into a Seq2seq paradigm, showcasing promising initial alignment abilities. The recent development and application of LLMs have introduced innovative pair-wise and list-wise rerankers, such as RankGPT [84], PRP [72], LRL [57] and RankLLaMA [56]. These models have brought multi-perspectives in addressing the fine-grained reranking problem. In real scenarios, users' expressions are often diverse and biased [7, 8, 15]. In response to the unique preferences of different users, various methods [41, 58, 68, 76] develope to achieve personalized user sorting, yielding significant results in aligning with industrial scenarios. These advancements inspire us to distill the preferences of LLMs into the reranker, facilitating the alignment between the RAG system's components.

3 Methodology

To address the misalignment between different components of retrieval-augmented generation (RAG) and improve overall generation performance, we propose the DPA-RAG framework, which is illustrated in Figure 2. In general, DPA-RAG improves traditional RAG architecture in two main aspects: (1) we fine-tune a preference-aligned reranker between the retriever and the LLM to selectively filter out knowledge that aligns with LLMs' knowledge preferences (§3.3), and (2) we design a self-alignment mechanism that fine-tunes the LLM to better recognize and utilize knowledge consistent with its reasoning preferences (§3.4). To acquire the LLM's preference knowledge, we devise a three-step data construction method, motivated by our preliminary analysis of how different types of retrieved documents affect RAG performance (§3.2). Below, we will first introduce the task definition (§3.1) and then delve into the specifics of our approach.

3.1 Task Definition

Compared to standard text generation, RAG often follows a *retrieve-then-read* paradigm [36], where an additional retriever is introduced to collect external knowledge and enhance the generation process. This architecture involves constructing a *query* q to reflect the information needs of the generation. For example, in question-answering systems, the input question is often used as the query. Given the query q, the retriever R returns relevant documents from a corpus $D_q = \{d_i\}_{i=1}^{N}$ with N documents. The relevance between document d and the query q can be measured by various methods. In this work, we employ a dense retriever that utilizes dual encoders to obtain hidden representations for both the query and the documents. The relevance score is then calculated by computing the dot-product similarity between these representations, enabling the retrieval of the top-k documents D_{retrieve}:

$$D_{\text{retrieve}} = \text{argtop-}k \left[E_{\text{d}}(d_i)^{\top} \cdot E_{\text{q}}(q) \mid i = \{1 \dots N\} \right]. \quad (1)$$

While the retrieved documents are relevant to the query, they may not necessarily contain the knowledge required by the LLMs. Therefore, in this study, we introduce a reranker E_r to rerank D_{retrieve} and filter out the documents D_{rerank}, which include only those documents aligned with the LLMs' preferences, *i.e.*, $D_{\text{rerank}} = E_r(q, D_{\text{retrieve}})$. Finally, the LLMs read from the reranked documents and generate the target text based on the query:

$$y = \text{LLM}(q, D_{\text{rerank}}) = \log P_\theta\ (q, D_{\text{rerank}}), \quad (2)$$

where P_θ represents the LLM's generation probability distribution.

Recognizing that LLMs might struggle to effectively utilize retrieved knowledge, we also design a self-alignment mechanism to optimize θ for RAG tasks.

3.2 Preference Knowledge Construction

To mitigate the misalignment between different RAG components, a critical step is to collect data that reflects LLMs' knowledge preferences. Therefore, we design a three-step method to gradually mine, augment, and filter out high-quality preference knowledge of LLMs, which are shown in the Figure 2.

3.2.1 Preference Knowledge Extraction To align with LLMs' knowledge preferences, it is essential to identify the specific knowledge that can bring performance gains or harms during the model's inference process.

Motivated by the preliminary analysis in Figure 1, given the training set $\widetilde{D}_{\text{train}} = \{q_i, D_{q_i}, y_{q_i}\}_{i=1}^{N_{\text{train}}}$, where each sample includes a query q_i, top-k retrieved documents $D_{q_i} = \{d_i\}_{i=1}^{k}$ and an answer y_{q_i}. We guide LLMs to directly answer questions or response by referencing different types of documents, aiming to filter out samples from $\widetilde{D}_{\text{train}}$ that reflects LLMs' knowledge preferences.

To ensure the distinctiveness among these documents, we hierarchically sample four documents from D_{q_i} to construct the document subset $D_{q_i}^{\text{sub}} = \{d_i | i = 1, 25, 50, 100\}$ for each query, as shown in the upper part of Figure 2. Consequently, we also categorize the results of LLMs into *"Both Correct"*, *"Both Incorrect"*, *"Aligned Knowledge"*, and *"Unaligned Knowledge"*. From $\widetilde{D}_{\text{train}}$, we selectively extract samples whose document subsets $D_{q_n}^{\text{sub}}$ contain at least one document labeled as *"Aligned Knowledge"* or *"Unaligned Knowledge"*. This allows us to obtain the preference dataset $\widetilde{D}_{\text{pref}} = \{qi, D_{q_i}^{\text{sub}}, Y_i^{\text{sub}}\}_{i=1}^{N}$, where $Y_i^{\text{sub}} = \{y_i | i = 1, 25, 50, 100\}$ denotes the preference labels of $D_{q_i}^{\text{sub}}$, corresponding to the four distinct categories.

The motivation behind this selection process is that documents labeled as *"Aligned Knowledge"* or *"Unaligned Knowledge"* provide the LLM with a clear positive or negative impact during reasoning. Due to the difficulty in distinguishing the role of retrieved documents labeled as *"Both Correct"* or *"Both Incorrect"*, we choose to discard them.

3.2.2 Diverse Query Augmentation After obtaining the dataset $\widetilde{D}$pref that reflects the preferences of the LLM, we encountered an issue: data scarcity — $\widetilde{D}$pref contains only 20% of the data from $\widetilde{D}_{\text{train}}$. This scarcity hinders subsequent fine-tuning and alignment of the LLM. Furthermore, data sparsity leads to limited patterns,

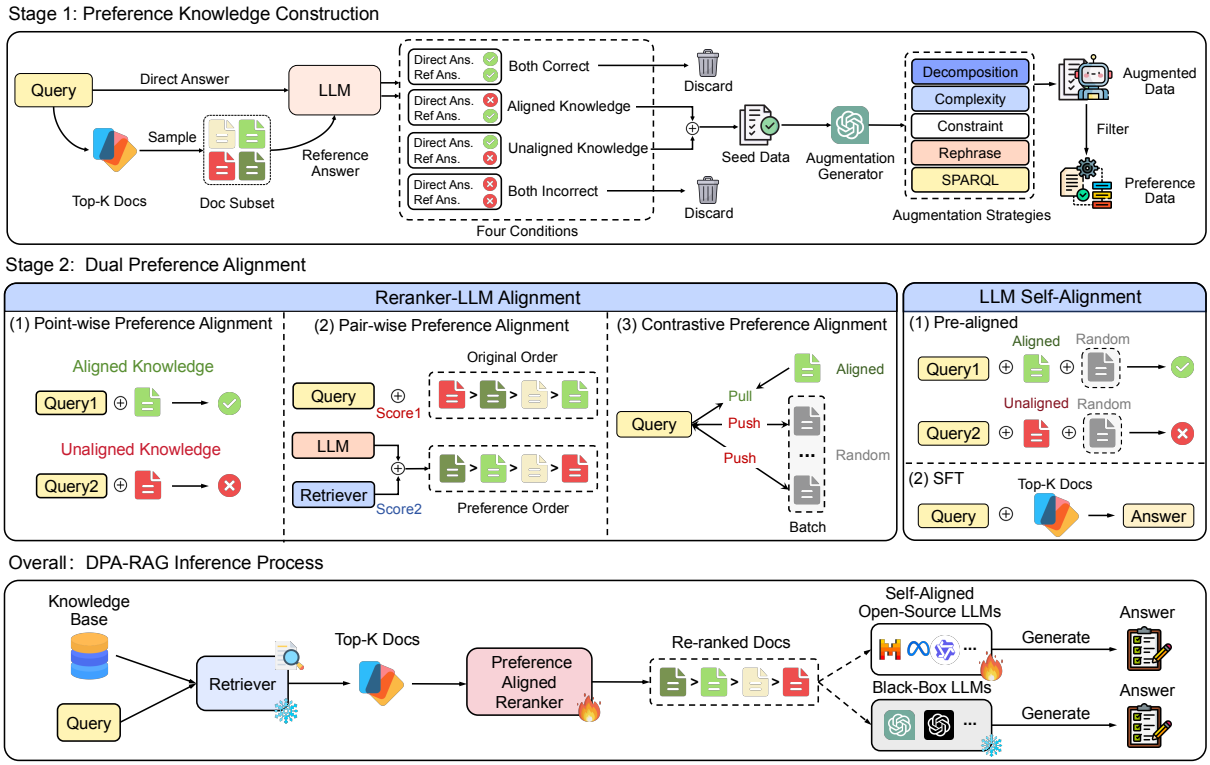

Figure 2: The overall framework of DPA-RAG. The upper part shows the pipeline of preference knowledge construction. The middle part displays the task format of dual preference alignment. The bottom part illustrates our inference process.

which in turn results in insufficient diversity and complexity in the data [48, 108]. To address these limitations, we draw inspiration from various augmentation techniques [38, 52, 54, 104, 106] and propose five query augmentation strategies specifically designed for the RAG system:[1]

• **Rephrasing.** Rephrase the original query with the same intention.
• **Complexity.** Increase the semantic complexity of the original query.
• **Decomposition.** Decompose the original query into several sub-problems.
• **Constraint.** Add more conditional and constrained statements to the original query.
• **SPARQL.** Rewrite the original query based on the SPARQL syntax and generate it directly.

We utilize GPT-3.5-turbo generate different augmneted datasets $\{\widetilde{D}_{r_i}\}_{i=1}^{n}$, and then merge them with original dataset $\widetilde{D}_{\text{ori}}$, which can be formulated as $\widetilde{D}_{\text{pref}}^{\text{ori}} = \widetilde{D}_{\text{pref}}^{\text{ori}} \cup (\cup_{i=1}^{n} \widetilde{D}_{r_i})$.

To control the augmented data's quality, we introduce a quality filtering procedure by a natural language inference (NLI) model. Given the original query q as the *"premise"* and the augmented query q_{aug} as the *"hypothesis"*, the NLI model seeks to determine the semantic relationship between the two queries. The relation can be categorized as *entailment*, *contradiction*, or *neutral*, as follows:

$$p_\theta(\cdot \mid q, q_{\text{aug}}) = \text{softmax}\left(\text{score}_\theta(q, q_{\text{aug}})\right), \tag{3}$$

where $\text{score}_\theta : \mathbb{R}^{k\times\ell_q} \times \mathbb{R}^{k\times\ell_{qaug}} \to \mathbb{R}^3$ is a scoring function dependent on the model's parameters θ. To maintain intent consistency between the original and augmented datasets, we exclude any augmented data labeled as "contradiction" (approximately 20%).

3.3 Reranker-LLM Alignment

After obtaining D_{pref}, we introduce multi-grained preference alignment tasks to jointly fine-tune a reranker, aiming to filter retrieved knowledge that aligns with LLM preferences.

3.3.1 Point-wise Preference Alignment Distinguishing beneficial or harmful knowledge of LLMs is essential for aligning their p references. Hence, from each sample $\{q_i, D_{q_i}^{\text{sub}}, Y_i^{\text{sub}}\} \sim \widetilde{D}_{\text{pref}}$, we can further extract one sub-sample $\{q_i, d_i, y_i\}$ where y_i is labeled as *"Aligned Knowledge"* or *"Unaligned Knowledge"*. As shown in Figure 2, we use $\{q_i, d_i, y_i\}_{i=1}^{N}$ to fine-tune the Reranker model $E_r(\theta)$ with binary cross-entropy loss [80], achieving a point-wise preference alignment:

$$\mathcal{L}_{\text{point}} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log\left(p_\theta(q_i, d_i)\right) + (1-y_i)\log(1-p_\theta(q_i, d_i))\right],$$

where y_i is label (Postive / Negative) for judging the d_i is aligned or unaligned knowledge.

[1]Detailed information on the different augmentation strategies can be found in Appendix C.2

3.3.2 Pair-wise Preference Alignment Since point-wise alignment empowers the reranker to identify LLM's favored knowledge, enhancing the reranker to prioritize this preferred knowledge presents a new challenge. Therefore, we propose a pair-wise preference ranking task for fine-grained alignment. In detail, given $\{q_i, D_{q_i}^{\text{sub}}, y_i^{\text{sub}}\} \sim \widetilde{D}_{\text{pref}}$, we derive an order $\{o_i\}_{i=1}^{K}$ of the documents subset $D_{q_i}^{\text{sub}} = \{d_i\}_{i=1}^{K}$ based on the initial similarity scores from the retriever.

Our idea is elegantly simple: we leverage the LLM within the RAG system as a preference reward model r_θ to score documents, eliminating the need for external experts. To mitigate bias from relying solely on LLM-generated preference scores [117], we calculate the preference score s_i for each query by weighting both the LLM preference score r_θ and the original similarity score $s_R(\cdot)$ from the retriever:

$$s_i = a \cdot r_\theta(q, d_i) + (1 - a) \cdot s_R(q, d_i), \tag{4}$$

where s_i denotes the preference score of the i-th retrieved document. We then sort the documents according to these preference scores to obtain the LLM's knowledge preference order $\{\hat{o_i}\}_{i=1}^{K}$. Subsequently, we integrate the preference order into the reranker using RLHF loss [66, 83]:

$$\mathcal{L}_{\text{pair}} = -\frac{1}{C_k^2}\,\mathbb{E}_{(q,d_w,d_l,y_w,y_l)\sim\widetilde{D}_{\text{pref}}} \left[\log\left(\sigma\left(p_\theta\left(q, d_w, y_w\right) - p_\theta\left(q, d_l, y_l\right)\right)\right)\right], \tag{5}$$

where y_w and y_l represent the labels for documents d_w and d_l, corresponding to *"winner"* or *"loser"* in the preference order $\{\hat{o_i}\}_{i=1}^{K}$. $p\theta$ denotes the logits of the output.[2]

3.3.3 Contrastive Preference Alignment To align query representations with the LLM's preferred knowledge, we employ contrastive learning [88] to fine-tune our reranker, thereby preventing the LLM from being misled by highly similar but unaligned knowledge. Unlike previous pairwise approaches [73], our $\widetilde{D}$pref dataset associates each query with multiple documents, rather than a single positive or negative example. Considering this one-to-N scenario, we employ Supervised Contrastive Learning (SCL) [32] to fully leverage $\widetilde{D}_{\text{pref}}$. In our task, the query serves as an anchor point h_q. Aligned documents are treated as positive samples h_p, while documents randomly sampled from other instances in the batch act as negative samples h_n. As shown in Figure 2, SCL seeks to reduce the distance of queries and positive samples h_p, while increasing the distance from negative samples h_n in the semantic space. The loss $\mathcal{L}_{\text{CPA}}$ is formulated as follows:

$$\mathcal{L}_{\text{CPA}} = -\sum_{i=1}^{N_t} \frac{1}{N_{y_i} - 1} \sum_{j=1}^{N_t} \mathbf{1}_{i\neq j}\mathbf{1}_{y_i=y_j} \log \frac{\exp(h_q \cdot h_p/\tau)}{\sum_{k=1}^{N_t} \mathbf{1}_{i\neq k} \exp(h_q \cdot h_n/\tau)},$$

where N_t is the nums of samples in each batch. N_{y_i} denotes samples in the batch with same label as y_i. τ is a temperature parameter. $\mathbf{1}$ is an indicator.

3.3.4 Multi-task Optimization Optimizing multi-grained preference tasks via Multi-task Learning (MTL) [5, 12] offers an efficient way for fine-tuning the reranker. However, learning tasks jointly may further introduce potential bias and conflicts [55]. To tackle this challenge, we employ the MGDA-UB [79], aiming to dynamically find a pareto optimal [42] solution for balancing multi-task optimization.

By utilizing MGDA-UB to optimize the MTL weights $\{c^t\}_{t=1}^{T}$ for T tasks. We finally obtain the multi-grained alignment loss function:

$$\mathcal{L}_{\text{total}} = c^1\mathcal{L}_{\text{point}} + c^2\mathcal{L}_{\text{pair}} + c^3\mathcal{L}_{\text{CPA}}. \tag{6}$$

3.4 LLM Self-Alignment

After initially aligning the preferences between external RAG components, in this section, we focus on guiding LLMs to emphasize aligned knowledge during the reasoning process to achieve internal alignment. Inspired by several pre-alignment works [43, 91], we introduce a pre-aligned stage to assist LLMs in implicitly identifying the knowledge crucial for reasoning [26].

Pre-aligned Stage. As illustrated in Figure 2, for each sample $\{q_i, D_{q_i}^{\text{sub}}, Y_i^{\text{sub}}\} \sim \widetilde{D}_{\text{pref}}$, we randomly select one document d_q labeled *"Aligned Knowledge"* or *"Unaligned Knowledge"* from $D_{q_i}^{\text{sub}}$, along with $k-1$ random documents from the retrieved corpus $D = \{d_i\}_{i=1}^{N}$. This selection process constructs a top-k document set $D_{\text{align}} = \{d_q, d_{\text{rand}_1}, \ldots, d_{\text{rand}_{k-1}}\}$ for each query q. Then we perform the following training objective with task specific template:[3]

$$\mathcal{L}\left(\theta\right) = \sum_{(q_n, D_q, y_n)\in D_{\text{pref}}} \log P_\theta\left(y_n | \,\text{prompt}(q_n, D_{\text{align}})\right), \tag{7}$$

Prompt: Given the documents $\{D_{\text{align}} = (d_q, d_{\text{rand}_1}, \ldots, d_{\text{rand}_{k-1}})\}$. Answer the following question based on the given information or your internal knowledge without the source with a few words. Query: $\{q\}$.
[Judgement]: document-$\{i_{d_q}\}$ is Positive or Negative knowledge for answering question.

where $\log P(\cdot)$ denote probability distribution of LLM's output. θ denotes model parameters. $\{i_{d_q}\}$ represents the position of the preference document. LLMs will implicitly learn the ability to capture self-preferred knowledge from top-k documents by distinguishing $y \in \{positive, negative\}$ during the pre-aligned task.

Supervied Fine-tuning Stage. Following the pre-aligned task, we load pre-trained parameters and perform subsequent Supervised Fine-tuning (SFT) for QA tasks using the same objective described in Equation (7). We utilize the traditional QA format training set $\widetilde{D}_{\text{train}} = \{q_i, D_{q_i}, y_{q_i}\}_{i=1}^{N_{\text{train}}}$. Moreover, we merge five augmented datasets $\{\widetilde{D}_{r_i}\}_{i=1}^{5}$ with $\widetilde{D}_{\text{train}}$. Using the preference-aligned reranker E_r, we reorder the documents and filter out the top-k documents as described in Equation (8), forming the final training set $\widetilde{D}_{\text{train}}^{\text{rank}} = \{q_i, D_{q_i}^{\text{rank}}, y_{q_i}\}_{i=1}^{N_{\text{train}}}$ of SFT stage.

$$D_{q_i}^{\text{rank}} = \text{argtop-}k\left[E_r(q_i, D_{q_i})\right]. \tag{8}$$

The preference knowledge identification capability developed during the pre-aligned stage enables LLMs to focus more effectively on aligned knowledge during the SFT stage, thereby enhancing their internal alignment potential. The prompt template for the SFT stage is as follows:

[2] An in-depth discussion on scoring mechanisms for different LLMs can be found in Appendix A.2.

[3] The document d_q is placed at a random position among k documents.

Table 1: Results of DPA-RAG and different kinds of baselines on four QA benchmarks.

Method	Reader	NQ		Trivia-QA		Hotpot-QA		WebQSP	
		Hit@1	F1	Hit@1	F1	Hit@1	F1	Hit@1	F1
Traditional RAG with DPR									
RAG [67]	GPT-3.5	47.47	47.99	75.04	74.13	26.28	32.84	67.97	63.33
RAG [65]	GPT-4	54.04	51.19	79.98	76.85	28.46	33.87	71.30	67.20
RAG [86]	LLaMA2-7B	50.94	54.76	63.90	63.80	31.40	38.90	68.52	64.22
RAG [86]	LLaMA2-13B	56.60	60.60	70.43	71.32	36.31	45.23	76.39	78.63
RAG [59]	LLaMA3-8B	54.81	58.33	69.54	71.21	34.28	42.29	72.82	73.94
RAG [2]	Qwen2-7B	52.01	56.13	63.88	66.52	31.39	39.70	75.98	77.82
RAG with DPR & Reranker									
RAG+RankGPT [84]	LLaMA2-7B	47.81	52.37	59.05	56.39	28.32	37.06	66.32	62.22
RAG+LRL [57]	LLaMA2-7B	48.09	53.06	60.33	56.86	29.13	37.81	67.43	63.44
RAG+PRP [72]	LLaMA2-7B	51.91	56.17	62.28	57.98	31.90	40.87	68.54	64.08
RAG+RankLLaMA [56]	LLaMA2-7B	52.18	56.62	62.34	58.05	32.31	41.39	69.11	65.70
RAG+BGE [98]	LLaMA2-7B	52.43	56.92	62.70	57.58	32.53	41.73	70.20	68.80
RAG+BCEmbedding [61]	LLaMA2-7B	49.91	53.19	61.93	57.67	31.52	40.59	68.20	65.40
RAG+ColBERTv2 [77]	LLaMA2-7B	51.49	56.02	62.34	58.16	31.72	40.79	69.70	66.90
Preference-aligned Methods for RAG									
REPLUG [81]	GPT-3.5	49.67	50.58	75.67	75.34	27.30	34.30	69.59	66.22
RA-Judgement [75]	GPT-3.5	48.52	50.18	76.21	76.58	26.50	32.81	66.07	68.32
KnowPAT [110]	LLaMA2-7B	51.42	54.82	63.20	65.20	29.00	37.40	68.73	65.31
RRHF [107]	LLaMA2-7B	50.11	52.01	62.50	60.20	28.16	35.40	66.90	63.10
RAFT [109]	LLaMA2-7B	50.24	53.86	60.10	57.40	30.20	35.80	-	-
FILCO [93]	LLaMA2-7B	52.71	55.32	67.30	67.80	32.70	40.80	69.96	68.34
Our Method: DPA-RAG									
DPA-RAG	GPT-3.5	51.60 (+4.13)	52.80 (+4.81)	78.65 (+3.61)	77.05 (+2.92)	28.42 (+2.14)	36.12 (+3.28)	71.80 (+3.83)	69.20 (+5.87)
DPA-RAG	GPT-4	56.45 (+2.41)	53.28 (+2.09)	84.41 (+4.43)	80.08 (+3.23)	33.79 (+5.33)	37.67 (+3.80)	73.12 (+1.82)	74.83 (+7.63)
DPA-RAG	LLaMA2-7B	56.03 (+5.09)	60.19 (+5.43)	70.16 (+6.26)	70.29 (+6.49)	35.23 (+3.83)	43.34 (+4.44)	72.40 (+3.88)	71.80 (+7.58)
DPA-RAG	LLaMA2-13B	59.19 (+2.59)	62.97 (+2.37)	74.18 (+3.75)	75.53 (+4.31)	41.07 (+4.76)	49.60 (+4.37)	80.28 (+3.89)	81.74 (+3.11)
DPA-RAG	LLaMA3-8B	57.43(+2.62)	61.02 (+2.69)	72.04(+2.50)	73.58 (+2.37)	36.01 (+1.73)	44.32 (+2.03)	74.26 (+1.44)	76.11 (+2.17)
DPA-RAG	Qwen2-7B	54.66(+2.65)	58.84 (+2.71)	68.58(+4.70)	70.26 (+3.74)	34.56 (+2.87)	42.47 (+2.77)	78.66 (+2.68)	80.53 (+2.71)

Prompt: Given the documents {Top-K Docs: D_q^{rank}}. Answer the following question based on the given information or your internal knowledge without the source with a few words. Query:{q}.

4 Experiments

4.1 Datasets and Metrics

We select four question answering (QA) datasets covering three types, including **(1) Open-Domain QA**, represented by NaturalQuestions (NQ) [33] and TriviaQA (TQA) [27]; **(2) Multi-Hop QA**, represented by HotpotQA (HQA) [101]; and **(3) Knowledge Base QA**, represented by WebQuestionsSP (WebQSP) [103]. For evaluation metrics, we use Hit@1 for the accuracy of the top-ranked response and F1 score to assess the quality and similarity to the ground-truth. We also provide a detailed **estimation of the training and inference FLOPs** of DPA-RAG compared to baselines in Appendix A.3 and Table 4, validating the efficiency of DPA-RAG. More details of the experimental setup are listed in Appendix B.

4.2 Overall Results

The experimental results are shown in Table 1. In general, our DPA-RAG significantly outperforms all baselines across four datasets in different setups. This highlights the superiority of our approach. We further have the following observations:

(1) Compared to traditional RAG baselines, DPA-RAG (LLaMA2-7B) shows a remarkable performance improvement (over 5%) across all four datasets. More importantly, this improvement is consistent across various models, including LLaMA2-13B, Qwen2-7B, LLaMA3-8B, GPT-3.5, and GPT-4. This indicates the broad applicability and generalizability of our method.

(2) For reranker-based methods, we find that smaller rerankers such as BGE and ColBERTv2 can achieve comparable or even better performance than LLM-based rerankers. This result validates our motivation for using BGE as the alignment backbone, as it combines efficiency with effectiveness.

(3) Among preference-aligned methods, DPA-RAG outperforms direct alignment methods (*i.e.*, REPLUG and RA-Judgement), which rely on logits. This emphasizes the value of implementing multi-grained alignments within our framework. Surprisingly, Filco, which

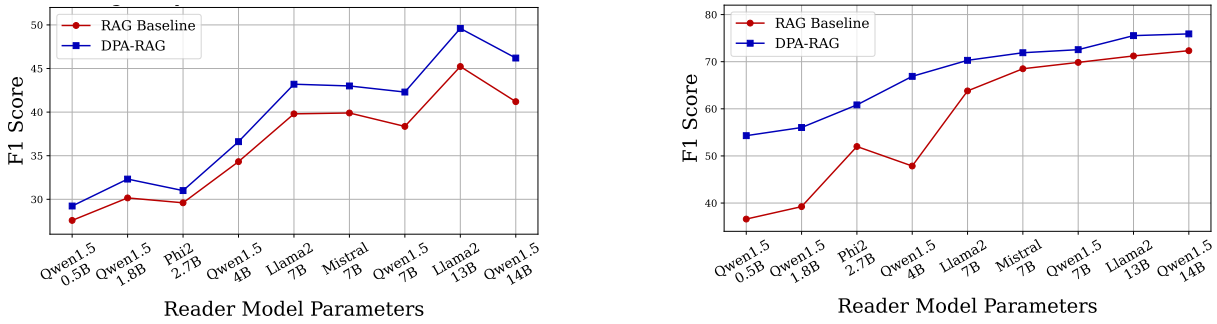

Figure 3: The scaling analysis of different parameter scales for HQA (left) and TQA (right).

Table 2: Ablation studies on NQ and TQA.

	NQ		TQA	
Method	**Hits@1**	**F1**	**Hits@1**	**F1**
Standard RAG	50.94	54.76	63.90	63.80
DPA-RAG	56.03	60.19	70.16	70.29
w/o PA-Rerank	52.80	60.19	66.26	66.39
w/o Pre-Align	54.31	58.95	61.69	61.35
w/o Pre-Align + PA-Rerank	51.91	55.98	58.24	59.30
w/o Query Augmentation	54.81	57.45	61.28	60.93

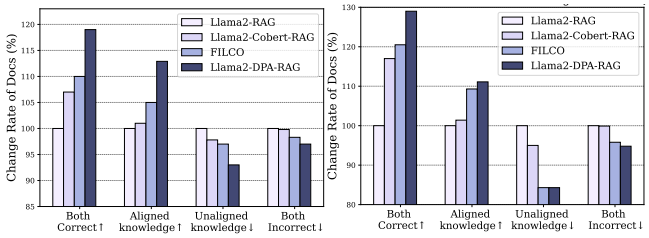

Figure 4: The comparison experiment of preference alignment on NQ (left) and TQA (right).

employs data filtering, shows robust alignment capabilities, confirming that unaligned knowledge exists in training corpora. This observation highlights again the importance of our preference optimization at the data level, ensuring that the retrieved and used knowledge is highly relevant and aligned with the LLM's needs.

Ablation Study. To explore the roles of different modules in DPA-RAG. We perform an ablation study and Table 2 shows the results. We use *w/o* to indicate the version *without* a particular module. We can see: (1) The performance of DPA-RAG declines when any component is removed, which suggests that all the components are very effective. (2) Removing the preference aligned reranker (PA-Rerank.) leads to the largest performance drop, indicating a clear knowledge preference gap between RAG components and LLMs. This confirms the benefit of using a preference-aligned reranker for external alignment. (3) The combined performance gains of preference aligned reranker and pre-aligned task are lower than the complete DPA-RAG framework, which implies that integrating both alignment methods yields a mutually reinforcing effect, demonstrating the superiority of our dual alignment strategies. More detailed results can be found in Appendix C.1.

4.3 Quantitative Analysis

4.3.1 Scaling Analysis for Different Model Parameters To investigate the impact of parameter scale and RAG performance, we gradually increase the parameters of LLM readers (ranging from 500M to 13B) and evaluate their performance. According to the results in Figure 3, we have following observations:

(1) **Emergence of RAG Capabilities at Lower Parameter Scales (<7B)**: We notice a significant improvement in RAG baseline performance, which sharply rises from 500M to 7B parameters (40% F1 score increase), then stabilizes for parameters beyond 7B. A similar pattern is observed in HQA, indicating a strong correlation between the emergence of RAG capabilities and model parameters. This finding presents an interesting parallel to those reported in LIMA [114], where parameter increases below a certain threshold significantly boost model capabilities.

(2) **Stable Performance Gains with DPA-RAG as Parameters Increase**: Compared to the baseline, DPA-RAG delivers stable improvements as parameter size expands across both datasets, displaying a smoother performance curve.

(3) **Greater Benefits from DPA-RAG in Datasets with More Unalignment**: The performance gains from DPA-RAG exhibit interesting variations between TQA and HQA as parameters increase. In TQA, where the average F1 score is over 60, the model quickly reaches a high-performance threshold as parameters increase, leaving limited room for further improvements through preference alignment. Conversely, HQA, characterized by more extensive unaligned knowledge and a lower average F1 score (below 50), shows that the alignment gains provided by DPA-RAG exceed those from increasing foundational RAG capabilities alone, leading to more improvement in alignment for RAG.

4.3.2 Effectiveness on Preference Alignment To delve deeper into the impact of preference alignment, in line with the setup in Section 3.2, we conduct a comparative experiment on direct query answering versus referencing top-3 documents. As shown in Figure 4, DPA-RAG consistently achieve the highest scores in the *"Aligned Knowledge"* category across all three datasets, while significantly reducing the *"Unaligned Knowledge"* category. This demonstrates that DPA-RAG effectively aligns retrieved knowledge with the LLM's inherent preferences. Interestingly, the improvement of DPA-RAG

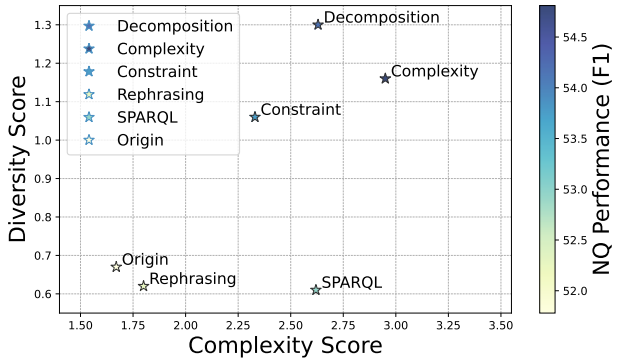

Figure 5: The visualization of different data complexity and diversity.

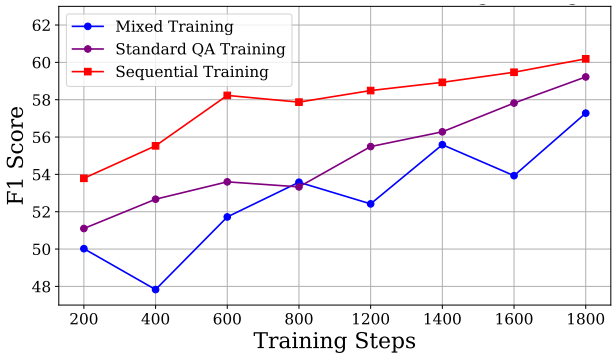

Figure 6: The performance of different training strategies on NQ.

Table 3: The performance result correlates with complexity and diversity on NQ.

Aug-Type	Complexity	Diversity	Total	NQ
Origin	1.61	0.35	1.96	51.78
Rephras.	1.64	0.39	2.03	52.27
SPARQL	1.77	0.39	2.16	52.95
Constraint	1.72	0.47	2.19	53.75
Decompos.	1.77	0.51	2.28	54.16
Complexity	1.85	0.48	2.33	54.81

in the "Both Correct" category even outperforms that observed in *"Aligned Knowledge"*. Given the significant decrease in *"Unaligned Knowledge"*, this suggests that DPA-RAG prioritizes addressing the conflicts present in retrieved documents. This behavior is in line with our pipeline's core principle: the preference-aligned reranker first externally eliminates misaligned knowledge, and the subsequent self-alignment stage allows the LLM to more effectively and implicitly capture information that is aligned with its preferences.

4.3.3 Discussion on Query Augmentations Liu et al. [48] and Lu et al. [51] highlight the significant impact of dataset complexity and diversity on model alignment. To investigate how the complexity and diversity of our augmented queries affect RAG performance, we randomly select 1,000 samples from each dataset and employ Intag technology [51] for automated intent annotation. For each dataset, we measure diversity by calculating $\frac{\#\text{ unique tags}}{\#\text{ all samples}}$ and complexity by $\frac{\#\text{ all tags}}{\#\text{ all samples}}$. Figure 5 visualizes the quality of the augmented data, showing that our five methods consistently enhance data complexity. Specifically, *Complexity* and *Decomposition* markedly boost both complexity and diversity scores, which also align with the case studies presented in Table 5. Moreover, we mix the augmented data with the original training set in actual proportions and calculate the data quality. Table 3 (left) shows that all five augmentation strategies enhance the LLM's performance to different degrees. Surprisingly, when we sum up the two metrics, the overall trend of performance on NQ increases along with the growth of the total quality score. This insight further validates that in RAG tasks, the effectiveness of query augmentations is highly correlated with their complexity and diversity.

4.3.4 Sequential Training vs. Mixed Training In Section 3.4, we design a knowledge self-alignment task during the pre-aligned phase and further perform sequential SFT on the QA dataset. An alternative approach is directly mixing preference data with QA task data for joint training. Figure 6 illustrates the performance of these two training strategies across training steps. Compared to standard QA fine-tuning, we notice that mixing training data from both tasks leads to a noticeable performance decline and fluctuations. This result may stem from optimization conflicts in multi-task training [13]. However, the sequential training after the pre-aligned phase yields stable performance gains, validating its efficacy. Similar conclusions have been reported in studies on reasoning [16, 85, 96].

5 Conclusion

In this paper, we reveal the inherent preference gap among RAG components and first propose DPA-RAG to align diverse knowledge preferences. Specifically, we gradually extract and filter out the LLM preferred knowledge from training set, and propose five high-quality query augmentation strategies to alleviate data sparsity issues. Based on preference data, we jointly integrate pair-wise, point-wise, and contrastive preference alignment abilities into the reranker, achieving external preference alignment among RAG components. Further, we introduce LLM Self-Alignment task to remove knowledge biases and achieve internal alignment. Experimental results demonstrate that DPA-RAG outperforms all strong baselines across four knowledge-intensive QA datasets. Further analysis provides practical insights for developing reliable RAG systems.

Acknowledgments

Zhicheng Dou is the corresponding author. This work was supported by Beijing Natural Science Foundation No. L233008, National Natural Science Foundation of China No. 62272467, Beijing Municipal Science and Technology Project No. Z231100010323009, the fund for building world-class universities (disciplines) of Renmin University of China. The work was partially done at the Engineering Research Center of Next-Generation Intelligent Search and Recommendation, MOE.

References

[1] Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernández Ábrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan A. Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Díaz, Nan Du, Ethan Dyer, Vladimir Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, and et al. 2023. PaLM 2 Technical Report. *CoRR* abs/2305.10403 (2023). https://doi.org/10.48550/ARXIV.2305.10403 arXiv:2305.10403

[2] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen Technical Report. *CoRR* abs/2309.16609 (2023). https://doi.org/10.48550/ARXIV.2309.16609 arXiv:2309.16609

[3] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen Technical Report. *CoRR* abs/2309.16609 (2023). https://doi.org/10.48550/ARXIV.2309.16609 arXiv:2309.16609

[4] Luiz Henrique Bonifacio, Hugo Queiroz Abonizio, Marzieh Fadaee, and Rodrigo Frassetto Nogueira. 2022. InPars: Data Augmentation for Information Retrieval using Large Language Models. *CoRR* abs/2202.05144 (2022). arXiv:2202.05144 https://arxiv.org/abs/2202.05144

[5] Rich Caruana. 1997. Multitask Learning. *Mach. Learn.* 28, 1 (1997), 41–75. https://doi.org/10.1023/A:1007379606734

[6] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé de Oliveira Pinto, Jared Kaplan, Harrison Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Joshua Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. Evaluating Large Language Models Trained on Code. *CoRR* abs/2107.03374 (2021). arXiv:2107.03374 https://arxiv.org/abs/2107.03374

[7] Guanting Dong, Daichi Guo, Liwen Wang, Xuefeng Li, Zechen Wang, Chen Zeng, Keqing He, Jinzheng Zhao, Hao Lei, Xinyue Cui, Yi Huang, Junlan Feng, and Weiran Xu. 2022. PSSAT: A Perturbed Semantic Structure Awareness Transferring Method for Perturbation-Robust Slot Filling. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12-17, 2022*, Nicoletta Calzolari, Chu-Ren Huang, Hansaem Kim, James Pustejovsky, Leo Wanner, Key-Sun Choi, Pum-Mo Ryu, Hsin-Hsi Chen, Lucia Donatelli, Heng Ji, Sadao Kurohashi, Patrizia Paggio, Nianwen Xue, Seokhwan Kim, Younggyun Hahm, Zhong He, Tony Kyungil Lee, Enrico Santus, Francis Bond, and Seung-Hoon Na (Eds.). International Committee on Computational Linguistics, 5327–5334. https://aclanthology.org/2022.coling-1.473

[8] Guanting Dong, Tingfeng Hui, Zhuoma Gongque, Jinxu Zhao, Daichi Guo, Gang Zhao, Keqing He, and Weiran Xu. 2023. DemoNSF: A Multi-task Demonstration-based Generative Framework for Noisy Slot Filling Task. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, 10506–10518. https://doi.org/10.18653/V1/2023.FINDINGS-EMNLP.705

[9] Guanting Dong, Rumei Li, Sirui Wang, Yupeng Zhang, Yunsen Xian, and Weiran Xu. 2023. Bridging the kb-text gap: Leveraging structured knowledge-aware pre-training for kbqa. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 3854–3859.

[10] Guanting Dong, Keming Lu, Chengpeng Li, Tingyu Xia, Bowen Yu, Chang Zhou, and Jingren Zhou. 2024. Self-play with Execution Feedback: Improving Instruction-following Capabilities of Large Language Models. *CoRR* abs/2406.13542 (2024). https://doi.org/10.48550/ARXIV.2406.13542 arXiv:2406.13542

[11] Guanting Dong, Xiaoshuai Song, Yutao Zhu, Runqi Qiao, Zhicheng Dou, and Ji-Rong Wen. 2024. Toward General Instruction-Following Alignment for Retrieval-Augmented Generation. *CoRR* abs/2410.09584 (2024). https://doi.org/10.48550/ARXIV.2410.09584 arXiv:2410.09584

[12] Guanting Dong, Zechen Wang, Jinxu Zhao, Gang Zhao, Daichi Guo, Dayuan Fu, Tingfeng Hui, Chen Zeng, Keqing He, Xuefeng Li, Liwen Wang, Xinyue Cui, and Weiran Xu. 2023. A Multi-Task Semantic Decomposition Framework with Task-specific Pre-training for Few-Shot NER. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM 2023, Birmingham, United Kingdom, October 21-25, 2023*, Ingo Frommholz, Frank Hopfgartner, Mark Lee, Michael Oakes, Mounia Lalmas, Min Zhang, and Rodrygo L. T. Santos (Eds.). ACM, 430–440. https://doi.org/10.1145/3583780.3614766

[13] Guanting Dong, Hongyi Yuan, Keming Lu, Chengpeng Li, Mingfeng Xue, Dayiheng Liu, Wei Wang, Zheng Yuan, Chang Zhou, and Jingren Zhou. 2023. How abilities in large language models are affected by supervised fine-tuning data composition. *arXiv preprint arXiv:2310.05492* (2023).

[14] Guanting Dong, Chenghao Zhang, Mengjie Deng, Yutao Zhu, Zhicheng Dou, and Ji-Rong Wen. 2024. Progressive multimodal reasoning via active retrieval. *arXiv preprint arXiv:2412.14835* (2024).

[15] Guanting Dong, Jinxu Zhao, Tingfeng Hui, Daichi Guo, Wenlong Wang, Boqi Feng, Yueyan Qiu, Zhuoma Gongque, Keqing He, Zechen Wang, and Weiran Xu. 2023. Revisit Input Perturbation Problems for LLMs: A Unified Robustness Evaluation Framework for Noisy Slot Filling Task. In *Natural Language Processing and Chinese Computing - 12th National CCF Conference, NLPCC 2023, Foshan, China, October 12-15, 2023, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 14302)*, Fei Liu, Nan Duan, Qingting Xu, and Yu Hong (Eds.). Springer, 682–694. https://doi.org/10.1007/978-3-031-44693-1_53

[16] Shihan Dou, Enyu Zhou, Yan Liu, Songyang Gao, Jun Zhao, Wei Shen, Yuhao Zhou, Zhiheng Xi, Xiao Wang, Xiaoran Fan, et al. 2023. The Art of Balancing: Revolutionizing Mixture of Experts for Maintaining World Knowledge in Language Model Alignment. *arXiv preprint arXiv:2312.09979* (2023).

[17] Yin Fang, Ningyu Zhang, Zhuo Chen, Lingbing Guo, Xiaohui Fan, and Huajun Chen. 2024. Domain-Agnostic Molecular Generation with Chemical Feedback. arXiv:2301.11259 [cs.LG]

[18] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. REALM: Retrieval-Augmented Language Model Pre-Training. *CoRR* abs/2002.08909 (2020). arXiv:2002.08909 https://arxiv.org/abs/2002.08909

[19] Mojan Javaheripi, Sébastien Bubeck, Marah Abdin, Jyoti Aneja, Sebastien Bubeck, Caio César Teodoro Mendes, Weizhu Chen, Allie Del Giorno, Ronen Eldan, Sivakanth Gopi, et al. 2023. Phi-2: The surprising power of small language models. *Microsoft Research Blog* (2023).

[20] Vitor Jeronymo, Luiz Henrique Bonifacio, Hugo Queiroz Abonizio, Marzieh Fadaee, Roberto de Alencar Lotufo, Jakub Zavrel, and Rodrigo Frassetto Nogueira. 2023. InPars-v2: Large Language Models as Efficient Dataset Generators for Information Retrieval. *CoRR* abs/2301.01820 (2023). https://doi.org/10.48550/ARXIV.2301.01820 arXiv:2301.01820

[21] Jiaming Ji, Tianyi Qiu, Boyuan Chen, Borong Zhang, Hantao Lou, Kaile Wang, Yawen Duan, Zhonghao He, Jiayi Zhou, Zhaowei Zhang, Fanzhi Zeng, Kwan Yee Ng, Juntao Dai, Xuehai Pan, Aidan O'Gara, Yingshan Lei, Hua Xu, Brian Tse, Jie Fu, Stephen McAleer, Yaodong Yang, Yizhou Wang, Song-Chun Zhu, Yike Guo, and Wen Gao. 2023. AI Alignment: A Comprehensive Survey. *CoRR* abs/2310.19852 (2023). https://doi.org/10.48550/ARXIV.2310.19852 arXiv:2310.19852

[22] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7B. *CoRR* abs/2310.06825 (2023). https://doi.org/10.48550/ARXIV.2310.06825 arXiv:2310.06825

[23] Jinhao Jiang, Kun Zhou, Zican Dong, Keming Ye, Xin Zhao, and Ji-Rong Wen. 2023. StructGPT: A General Framework for Large Language Model to Reason over Structured Data. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, 9237–9251. https://doi.org/10.18653/V1/2023.EMNLP-MAIN.574

[24] Zhengbao Jiang, Frank F. Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Active Retrieval Augmented Generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, 7969–7992. https://doi.org/10.18653/V1/2023.EMNLP-MAIN.495

[25] Jiajie Jin, Yutao Zhu, Xinyu Yang, Chenghao Zhang, and Zhicheng Dou. 2024. FlashRAG: A Modular Toolkit for Efficient Retrieval-Augmented Generation Research. *CoRR* abs/2405.13576 (2024). https://doi.org/10.48550/ARXIV.2405.13576 arXiv:2405.13576

[26] Jiajie Jin, Yutao Zhu, Yujia Zhou, and Zhicheng Dou. 2024. BIDER: Bridging Knowledge Inconsistency for Efficient Retrieval-Augmented LLMs via Key Supporting Evidence. *CoRR* abs/2402.12174 (2024). https://doi.org/10.48550/ARXIV.2402.12174 arXiv:2402.12174

[27] Mandar Joshi, Eunsol Choi, Daniel S. 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, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, Regina Barzilay and Min-Yen Kan (Eds.). Association for Computational Linguistics, 1601–1611. https://doi.org/10.18653/V1/P17-1147

[28] Jia-Huei Ju, Jheng-Hong Yang, and Chuan-Ju Wang. 2021. Text-to-Text Multi-view Learning for Passage Re-ranking. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, Fernando Diaz, Chirag Shah, Torsten Suel, Pablo Castells, Rosie Jones, and Tetsuya Sakai (Eds.). ACM, 1803–1807. https://doi.org/10.1145/3404835.3463048

[29] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling Laws for Neural Language Models. *CoRR* abs/2001.08361 (2020). arXiv:2001.08361 https://arxiv.org/abs/2001.08361

[30] Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. arXiv:2004.04906 [cs.CL]

[31] Omar Khattab and Matei Zaharia. 2020. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, Jimmy X. Huang, Yi Chang, Xueqi Cheng, Jaap Kamps, Vanessa Murdock, Ji-Rong Wen, and Yiqun Liu (Eds.). ACM, 39–48. https://doi.org/10.1145/3397271.3401075

[32] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised Contrastive Learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (Eds.). https://proceedings.neurips.cc/paper/2020/hash/d89a66c7c80a29b1bdbab0f2a1a94af8-Abstract.html

[33] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. 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. *Trans. Assoc. Comput. Linguistics* 7 (2019), 452–466. https://doi.org/10.1162/TACL_A_00276

[34] Sarah Lebovitz, Natalia Levina, and Hila Lifshitz-Assaf. 2021. Is AI Ground Truth Really True? The Dangers of Training and Evaluating AI Tools Based on Experts' Know-What. *MIS Q.* 45, 3 (2021). https://doi.org/10.25300/MISQ/2021/16564

[35] Shanglin Lei, Guanting Dong, Xiaoping Wang, Keheng Wang, and Sirui Wang. 2023. InstructERC: Reforming Emotion Recognition in Conversation with a Retrieval Multi-task LLMs Framework. *CoRR* abs/2309.11911 (2023). https://doi.org/10.48550/ARXIV.2309.11911 arXiv:2309.11911

[36] Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (Eds.). https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html

[37] Chengpeng Li, Guanting Dong, Mingfeng Xue, Ru Peng, Xiang Wang, and Dayiheng Liu. 2024. DotaMath: Decomposition of Thought with Code Assistance and Self-correction for Mathematical Reasoning. *CoRR* abs/2407.04078 (2024). https://doi.org/10.48550/ARXIV.2407.04078 arXiv:2407.04078

[38] Chengpeng Li, Zheng Yuan, Guanting Dong, Keming Lu, Jiancan Wu, Chuanqi Tan, Xiang Wang, and Chang Zhou. 2023. Query and response augmentation cannot help out-of-domain math reasoning generalization. *arXiv preprint arXiv:2310.05506* (2023).

[39] Huayang Li, Yixuan Su, Deng Cai, Yan Wang, and Lemao Liu. 2022. A Survey on Retrieval-Augmented Text Generation. *CoRR* abs/2202.01110 (2022). arXiv:2202.01110 https://arxiv.org/abs/2202.01110

[40] Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. 2025. Search-o1: Agentic Search-Enhanced Large Reasoning Models. *arXiv preprint arXiv:2501.05366* (2025).

[41] Yi Li, Jieming Zhu, Weiwen Liu, Liangcai Su, Guohao Cai, Qi Zhang, Ruiming Tang, Xi Xiao, and Xiuqiang He. 2022. PEAR: Personalized Re-ranking with Contextualized Transformer for Recommendation. In *Companion of The Web Conference 2022, Virtual Event / Lyon, France, April 25 - 29, 2022*, Frédérique Laforest, Raphaël Troncy, Elena Simperl, Deepak Agarwal, Aristides Gionis, Ivan Herman, and Lionel Médini (Eds.). ACM, 62–66. https://doi.org/10.1145/3487553.3524208

[42] Xi Lin, Hui-Ling Zhen, Zhenhua Li, Qingfu Zhang, and Sam Kwong. 2019. Pareto Multi-Task Learning. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (Eds.). 12037–12047. https://proceedings.neurips.cc/paper/2019/hash/685bfde03eb646c27ed565881917c71c-Abstract.html

[43] Fangyu Liu, Ehsan Shareghi, Zaiqiao Meng, Marco Basaldella, and Nigel Collier. 2021. Self-Alignment Pretraining for Biomedical Entity Representations. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tür, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (Eds.). Association for Computational Linguistics, 4228–4238. https://doi.org/10.18653/V1/2021.NAACL-MAIN.334

[44] Hao Liu, Carmelo Sferrazza, and Pieter Abbeel. 2023. Chain of Hindsight Aligns Language Models with Feedback. *CoRR* abs/2302.02676 (2023). https://doi.org/10.48550/ARXIV.2302.02676 arXiv:2302.02676

[45] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *ACM Comput. Surv.* 55, 9 (2023), 195:1–195:35. https://doi.org/10.1145/3560815

[46] Ruibo Liu, Ruixin Yang, Chenyan Jia, Ge Zhang, Denny Zhou, Andrew M. Dai, Diyi Yang, and Soroush Vosoughi. 2023. Training Socially Aligned Language Models in Simulated Human Society. *CoRR* abs/2305.16960 (2023).

[47] Tianqi Liu, Yao Zhao, Rishabh Joshi, Misha Khalman, Mohammad Saleh, Peter J. Liu, and Jialu Liu. 2023. Statistical Rejection Sampling Improves Preference Optimization. *CoRR* abs/2309.06657 (2023).

[48] Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. 2023. What Makes Good Data for Alignment? A Comprehensive Study of Automatic Data Selection in Instruction Tuning. *CoRR* abs/2312.15685 (2023). https://doi.org/10.48550/ARXIV.2312.15685 arXiv:2312.15685

[49] Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, and Adam Roberts. 2023. The Flan Collection: Designing Data and Methods for Effective Instruction Tuning. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA (Proceedings of Machine Learning Research, Vol. 202)*, Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (Eds.). PMLR, 22631–22648. https://proceedings.mlr.press/v202/longpre23a.html

[50] Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101* (2017).

[51] Keming Lu, Hongyi Yuan, Zheng Yuan, Runji Lin, Junyang Lin, Chuanqi Tan, Chang Zhou, and Jingren Zhou. 2023. #InsTag: Instruction Tagging for Analyzing Supervised Fine-tuning of Large Language Models. *CoRR* abs/2308.07074 (2023). https://doi.org/10.48550/ARXIV.2308.07074 arXiv:2308.07074

[52] Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, and Dongmei Zhang. 2023. WizardMath: Empowering Mathematical Reasoning for Large Language Models via Reinforced Evol-Instruct. *CoRR* abs/2308.09583 (2023). https://doi.org/10.48550/ARXIV.2308.09583 arXiv:2308.09583

[53] Haoran Luo, Zichen Tang, Shiyao Peng, Yikai Guo, Wentai Zhang, Chenghao Ma, Guanting Dong, Meina Song, Wei Lin, et al. 2023. Chatkbqa: A generate-then-retrieve framework for knowledge base question answering with fine-tuned large language models. *arXiv preprint arXiv:2310.08975* (2023).

[54] Ziyang Luo, Can Xu, Pu Zhao, Qingfeng Sun, Xiubo Geng, Wenxiang Hu, Chongyang Tao, Jing Ma, Qingwei Lin, and Daxin Jiang. 2023. WizardCoder: Empowering Code Large Language Models with Evol-Instruct. *CoRR* abs/2306.08568 (2023). https://doi.org/10.48550/ARXIV.2306.08568 arXiv:2306.08568

[55] Minh-Thang Luong, Quoc V. Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2016. Multi-task Sequence to Sequence Learning. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). http://arxiv.org/abs/1511.06114

[56] Xueguang Ma, Liang Wang, Nan Yang, Furu Wei, and Jimmy Lin. 2023. Fine-Tuning LLaMA for Multi-Stage Text Retrieval. *CoRR* abs/2310.08319 (2023). https://doi.org/10.48550/ARXIV.2310.08319 arXiv:2310.08319

[57] Xueguang Ma, Xinyu Zhang, Ronak Pradeep, and Jimmy Lin. 2023. Zero-Shot Listwise Document Reranking with a Large Language Model. *CoRR* abs/2305.02156 (2023). https://doi.org/10.48550/ARXIV.2305.02156 arXiv:2305.02156

[58] Yubo Ma, Yixin Cao, Yong Hong, and Aixin Sun. 2023. Large Language Model Is Not a Good Few-shot Information Extractor, but a Good Reranker for Hard

Samples!. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, 10572–10601. https://doi.org/10.18653/V1/2023.FINDINGS-EMNLP.710

[59] Meta. 2024. Introducing Meta Llama 3: The most capable openly available LLM to date. https://ai.meta.com/blog/meta-llama-3/

[60] Deepak Nathani, David Wang, Liangming Pan, and William Yang Wang. 2023. MAF: Multi-Aspect Feedback for Improving Reasoning in Large Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, 6591–6616. https://doi.org/10.18653/V1/2023.EMNLP-MAIN.407

[61] Inc. NetEase Youdao. 2023. BCEmbedding: Bilingual and Crosslingual Embedding for RAG. https://github.com/netease-youdao/BCEmbedding.

[62] Rodrigo Frassetto Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document Ranking with a Pretrained Sequence-to-Sequence Model. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020 (Findings of ACL, Vol. EMNLP 2020)*, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association for Computational Linguistics, 708–718. https://doi.org/10.18653/V1/2020.FINDINGS-EMNLP.63

[63] Rodrigo Frassetto Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. 2019. Multi-Stage Document Ranking with BERT. *CoRR* abs/1910.14424 (2019). arXiv:1910.14424 http://arxiv.org/abs/1910.14424

[64] OpenAI. 2023. GPT-4 Technical Report. *CoRR* abs/2303.08774 (2023). https://doi.org/10.48550/ARXIV.2303.08774 arXiv:2303.08774

[65] OpenAI. 2023. GPT-4 Technical Report. *CoRR* abs/2303.08774 (2023). https://doi.org/10.48550/ARXIV.2303.08774 arXiv:2303.08774

[66] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (Eds.). http://papers.nips.cc/paper_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html

[67] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *NeurIPS*.

[68] Changhua Pei, Yi Zhang, Yongfeng Zhang, Fei Sun, Xiao Lin, Hanxiao Sun, Jian Wu, Peng Jiang, Junfeng Ge, Wenwu Ou, and Dan Pei. 2019. Personalized re-ranking for recommendation. In *Proceedings of the 13th ACM Conference on Recommender Systems, RecSys 2019, Copenhagen, Denmark, September 16-20, 2019*, Toine Bogers, Alan Said, Peter Brusilovsky, and Domonkos Tikk (Eds.). ACM, 3–11. https://doi.org/10.1145/3298689.3347000

[69] Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick S. H. Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander H. Miller. 2019. Language Models as Knowledge Bases?. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (Eds.). Association for Computational Linguistics, 2463–2473. https://doi.org/10.18653/V1/D19-1250

[70] Ronak Pradeep, Rodrigo Frassetto Nogueira, and Jimmy Lin. 2021. The Expando-Mono-Duo Design Pattern for Text Ranking with Pretrained Sequence-to-Sequence Models. *CoRR* abs/2101.05667 (2021). arXiv:2101.05667 https://arxiv.org/abs/2101.05667

[71] Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. 2023. Measuring and Narrowing the Compositionality Gap in Language Models. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, 5687–5711. https://doi.org/10.18653/V1/2023.FINDINGS-EMNLP.378

[72] Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, and Michael Bendersky. 2023. Large Language Models are Effective Text Rankers with Pairwise Ranking Prompting. *CoRR* abs/2306.17563 (2023). https://doi.org/10.48550/ARXIV.2306.17563 arXiv:2306.17563

[73] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (Eds.). http://papers.nips.cc/paper_files/paper/2023/hash/a85b405ed65c6477a4fe8302b5e06ce7-Abstract-Conference.html

[74] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (Eds.). Association for Computational Linguistics, 3980–3990. https://doi.org/10.18653/V1/D19-1410

[75] Ruiyang Ren, Yuhao Wang, Yingqi Qu, Wayne Xin Zhao, Jing Liu, Hao Tian, Hua Wu, Ji-Rong Wen, and Haifeng Wang. 2023. Investigating the Factual Knowledge Boundary of Large Language Models with Retrieval Augmentation. *CoRR* abs/2307.11019 (2023). https://doi.org/10.48550/ARXIV.2307.11019 arXiv:2307.11019

[76] Jon Saad-Falcon, Omar Khattab, Keshav Santhanam, Radu Florian, Martin Franz, Salim Roukos, Avirup Sil, Md. Arafat Sultan, and Christopher Potts. 2023. UDAPDR: Unsupervised Domain Adaptation via LLM Prompting and Distillation of Rerankers. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, 11265–11279. https://doi.org/10.18653/V1/2023.EMNLP-MAIN.693

[77] 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, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, Marine Carpuat, Marie-Catherine de Marneffe, and Iván Vladimir Meza Ruíz (Eds.). Association for Computational Linguistics, 3715–3734. https://doi.org/10.18653/V1/2022.NAACL-MAIN.272

[78] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal Policy Optimization Algorithms. *CoRR* abs/1707.06347 (2017).

[79] Ozan Sener and Vladlen Koltun. 2018. Multi-Task Learning as Multi-Objective Optimization. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (Eds.). 525–536. https://proceedings.neurips.cc/paper/2018/hash/432aca3a1e345e339f35a30c8f65edce-Abstract.html

[80] Claude E. Shannon. 1948. A mathematical theory of communication. *Bell Syst. Tech. J.* 27, 3 (1948), 379–423. https://doi.org/10.1002/J.1538-7305.1948.TB01338.X

[81] 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. *CoRR* abs/2301.12652 (2023). https://doi.org/10.48550/ARXIV.2301.12652 arXiv:2301.12652

[82] Feifan Song, Bowen Yu, Minghao Li, Haiyang Yu, Fei Huang, Yongbin Li, and Houfeng Wang. 2024. Preference Ranking Optimization for Human Alignment. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, Michael J. Wooldridge, Jennifer G. Dy, and Sriraam Natarajan (Eds.). AAAI Press, 18990–18998. https://doi.org/10.1609/AAAI.V38I17.29865

[83] Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. 2020. Learning to summarize from human feedback. *CoRR* abs/2009.01325 (2020). arXiv:2009.01325 https://arxiv.org/abs/2009.01325

[84] Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. 2023. Is ChatGPT Good at Search? Investigating Large Language Models as Re-Ranking Agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, 14918–14937. https://doi.org/10.18653/V1/2023.EMNLP-MAIN.923

[85] Zhengyang Tang, Xingxing Zhang, Benyou Wang, and Furu Wei. 2024. MathScale: Scaling Instruction Tuning for Mathematical Reasoning. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net. https://openreview.net/forum?id=Kjww7ZN47M

[86] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang,

Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. *CoRR* abs/2307.09288 (2023). https://doi.org/10.48550/ARXIV.2307.09288 arXiv:2307.09288

[87] Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. Interleaving Retrieval with Chain-of-Thought Reasoning for Knowledge-Intensive Multi-Step Questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki (Eds.). Association for Computational Linguistics, 10014–10037. https://doi.org/10.18653/V1/2023.ACL-LONG.557

[88] Aäron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation Learning with Contrastive Predictive Coding. *CoRR* abs/1807.03748 (2018). arXiv:1807.03748 http://arxiv.org/abs/1807.03748

[89] Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A Free Collaborative Knowledgebase. *Commun. ACM* 57, 10 (sep 2014), 78–85. https://doi.org/10.1145/2629489

[90] Keheng Wang, Feiyu Duan, Peiguang Li, Sirui Wang, and Xunliang Cai. 2024. LLMs Know What They Need: Leveraging a Missing Information Guided Framework to Empower Retrieval-Augmented Generation. arXiv:2404.14043 [cs.CL]

[91] Yejie Wang, Keqing He, Guanting Dong, Pei Wang, Weihao Zeng, Muxi Diao, Yutao Mou, Mengdi Zhang, Jingang Wang, Xunliang Cai, et al. 2024. DolphCoder: Echo-Locating Code Large Language Models with Diverse and Multi-Objective Instruction Tuning. *arXiv preprint arXiv:2402.09136* (2024).

[92] Yufei Wang, Wanjun Zhong, Liangyou Li, Fei Mi, Xingshan Zeng, Wenyong Huang, Lifeng Shang, Xin Jiang, and Qun Liu. 2023. Aligning Large Language Models with Human: A Survey. *CoRR* abs/2307.12966 (2023). https://doi.org/10.48550/ARXIV.2307.12966 arXiv:2307.12966

[93] Zhiruo Wang, Jun Araki, Zhengbao Jiang, Md. Rizwan Parvez, and Graham Neubig. 2023. Learning to Filter Context for Retrieval-Augmented Generation. *CoRR* abs/2311.08377 (2023). https://doi.org/10.48550/ARXIV.2311.08377 arXiv:2311.08377

[94] Zihao Wang, Anji Liu, Haowei Lin, Jiaqi Li, Xiaojian Ma, and Yitao Liang. 2024. RAT: Retrieval Augmented Thoughts Elicit Context-Aware Reasoning in Long-Horizon Generation. *CoRR* abs/2403.05313 (2024). https://doi.org/10.48550/ARXIV.2403.05313 arXiv:2403.05313

[95] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (Eds.). http://papers.nips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html

[96] Chengyue Wu, Yukang Gan, Yixiao Ge, Zeyu Lu, Jiahao Wang, Ye Feng, Ping Luo, and Ying Shan. 2024. Llama pro: Progressive llama with block expansion. *arXiv preprint arXiv:2401.02415* (2024).

[97] Zeqiu Wu, Yushi Hu, Weijia Shi, Nouha Dziri, Alane Suhr, Prithviraj Ammanabrolu, Noah A. Smith, Mari Ostendorf, and Hannaneh Hajishirzi. 2023. Fine-Grained Human Feedback Gives Better Rewards for Language Model Training. *CoRR* abs/2306.01693 (2023).

[98] Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighof. 2023. C-Pack: Packaged Resources To Advance General Chinese Embedding. *CoRR* abs/2309.07597 (2023). https://doi.org/10.48550/ARXIV.2309.07597 arXiv:2309.07597

[99] A Yang, B Yang, B Hui, B Zheng, B Yu, C Zhou, C Li, C Li, D Liu, F Huang, et al. [n. d.]. Qwen2 technical report. arXiv 2024. *arXiv preprint arXiv:2407.10671* ([n. d.]).

[100] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. 2024. Qwen2 Technical Report. *arXiv preprint arXiv:2407.10671* (2024).

[101] 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. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (Eds.). Association for Computational Linguistics, 2369–2380. https://doi.org/10.18653/V1/D18-1259

[102] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. 2023. ReAct: Synergizing Reasoning and Acting in Language Models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net. https://openreview.net/pdf?id=WE_vluYUL-X

[103] Wen-tau Yih, Matthew Richardson, Christopher Meek, Ming-Wei Chang, and Jina Suh. 2016. The Value of Semantic Parse Labeling for Knowledge Base Question Answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers*. The Association for Computer Linguistics. https://doi.org/10.18653/V1/P16-2033

[104] Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T. Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2023. MetaMath: Bootstrap Your Own Mathematical Questions for Large Language Models. *CoRR* abs/2309.12284 (2023). https://doi.org/10.48550/ARXIV.2309.12284 arXiv:2309.12284

[105] Weizhe Yuan, Kyunghyun Cho, and Jason Weston. 2023. System-Level Natural Language Feedback. *CoRR* abs/2306.13588 (2023).

[106] Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Chuanqi Tan, and Chang Zhou. 2023. Scaling relationship on learning mathematical reasoning with large language models. *arXiv preprint arXiv:2308.01825* (2023).

[107] Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. 2023. RRHF: Rank Responses to Align Language Models with Human Feedback without tears. *CoRR* abs/2304.05302 (2023). https://doi.org/10.48550/ARXIV.2304.05302 arXiv:2304.05302

[108] Weihao Zeng, Can Xu, Yingxiu Zhao, Jian-Guang Lou, and Weizhu Chen. 2024. Automatic Instruction Evolving for Large Language Models. *CoRR* abs/2406.00770 (2024). https://doi.org/10.48550/ARXIV.2406.00770 arXiv:2406.00770

[109] 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. *CoRR* abs/2403.10131 (2024). https://doi.org/10.48550/ARXIV.2403.10131 arXiv:2403.10131

[110] Yichi Zhang, Zhuo Chen, Yin Fang, Yanxi Lu, Fangming Li, Wen Zhang, and Huajun Chen. 2024. Knowledgeable Preference Alignment for LLMs in Domain-specific Question Answering. arXiv:2311.06503 [cs.CL]

[111] Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023. Siren's Song in the AI Ocean: A Survey on Hallucination in Large Language Models. *CoRR* abs/2309.01219 (2023). https://doi.org/10.48550/ARXIV.2309.01219 arXiv:2309.01219

[112] Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J. Liu. 2023. SLiC-HF: Sequence Likelihood Calibration with Human Feedback. *CoRR* abs/2305.10425 (2023).

[113] Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, and Yongqiang Ma. 2024. LlamaFactory: Unified Efficient Fine-Tuning of 100+ Language Models. *arXiv preprint arXiv:2403.13372* (2024). http://arxiv.org/abs/2403.13372

[114] Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. LIMA: Less Is More for Alignment. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (Eds.). http://papers.nips.cc/paper_files/paper/2023/hash/ac662d74829e4407ce1d126477f4a03a-Abstract-Conference.html

[115] Yujia Zhou, Zheng Liu, Jiajie Jin, Jian-Yun Nie, and Zhicheng Dou. 2024. Metacognitive Retrieval-Augmented Large Language Models. *CoRR* abs/2402.11626 (2024). https://doi.org/10.48550/ARXIV.2402.11626 arXiv:2402.11626

[116] 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, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, Hsin-Hsi Chen, Wei-Jou (Edward) Duh, Hen-Hsen Huang, Makoto P. Kato, Josiane Mothe, and Barbara Poblete (Eds.). ACM, 2308–2313. https://doi.org/10.1145/3539618.3592047

[117] Shengyao Zhuang, Bing Liu, Bevan Koopman, and Guido Zuccon. 2023. Open-source Large Language Models are Strong Zero-shot Query Likelihood Models for Document Ranking. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, 8807–8817. https://doi.org/10.18653/V1/2023.FINDINGS-EMNLP.590

Appendix

Contents

A More Details about DPA-RAG

A.1 The Overall Algorithm Workflow of DPA-RAG

In this section, we delve into the overall workflow of the DPA-RAG algorithm, which can be divided into **Reranker Training Algorithm** and **LLM-based Generator Training**.

Reranker Training Algorithm: Given the train set $\widetilde{D}_{\text{train}} = \{q_i, D_{q_i}, y_{q_i}\}_{i=1}^{N_{\text{train}}}$, we initially perform preference knowledge mining techniques to select, augment and filter the data to construct a preference-aligned dataset $\widetilde{D}_{\text{pref}}$. Subsequently, relying on the $\widetilde{D}_{\text{pref}}$, we perform multi-grained distillation alignments with MGDA-UB stategy to better fine-tune a preference-aligned reranker. The detailed process is listed in algorithm diagram 1.

LLM-based Reader Training Algorithm: As shown in algorithm diagram 2, for open-source LLM-based reader, we directly utilize the preference-aligned reranker to perform preference-based reranking on retrieved documents in $\widetilde{D}_{\text{train}}$[4] and $\widetilde{D}_{\text{test}}$, resulting in sorted datasets $\widetilde{D}^{\text{rank}}_{\text{train}}$ and $\widetilde{D}^{\text{rank}}_{\text{test}}$. In addition, we also construct a dataset $\widetilde{D}^{\text{PA}}_{\text{train}}$ for the knowledge self-alignment task based on $\widetilde{D}_{\text{pref}}$. Initially, we use $\widetilde{D}^{\text{PA}}_{\text{train}}$ for the pre-aligned task, then we load the pre-trained model parameters and then conduct vanilla QA supervised fine-tuning based on $\widetilde{D}^{\text{rank}}_{\text{train}}$. During the inference phase, we input the preference-sorted test set $\widetilde{D}^{rank}_{\text{test}}$ into the LLM to complete the prediction.

For close-source LLM-based reader, the process is more simple: the preference-aligned reranker is used to sort documents in the test set $\widetilde{D}_{\text{test}} \rightarrow \widetilde{D}^{\text{rank}}_{\text{test}}$, then we use LLMs for the prediction process.

A.2 Preference Scoring Mechanism for Different LLMs

In practice, we find that models with fewer than 7B parameters struggle with instruction-following capabilities, making it difficult for them to perform the scoring task. To address this, we follow the RankLLaMA [56] and RePLUG [81], utilizing the output's logit as the basis for scoring as follow:

$$r_\theta(q, d_i) = \log \mathbf{P}_\theta\ (\ \text{prompt}\ (q, d_i)) \tag{9}$$

$$s_i = a \cdot r_\theta(q, p_i) + (1 - a) \cdot s_R(q, p_i) \tag{10}$$

where q, d_i denotes the query and top i-th document. $\log \mathbf{P}(\cdot)$ represents the model's probability distribution. Prompt denotes the prompt template. s_i is the final preference score of i-th retrieved document. For the hyper-parameter a, we follow QLM Reranker [117] and set it to 0.8 without performing any grid search. Next, we rank them to obtain the preference order $\{o_1, o_2, .., o_n \mid r_\theta, s_R\}$ according to $\{s_i\}_{i=1}^{K}$.

For the 7B and 13B models, we observe that these models fundamentally possess the capability to follow instructions in our preliminary experiments. Therefore, we prompt them to perform preference scoring from 1 to 5. Then we normalize the preference score $r_\theta(q, d_i)$ and sum it with the retriever's similarity score $s_R(q, d_i)$ as equation 10. Finally, we rank them to obtain the preference order.

As the result in Table 1, for powerful LLMs (such as GPT-3.5 and GPT-4), we find that pair-wise comparative ranking can achieve a more precise preference ordering compared to ranking by scoring each paragraph individually. Therefore, we perform C_k^2 pair-wise comparisons of knowledge documents as PRP [72] through LLMs to obtain the preference ordering results.

A.3 Estimating FLOP of Training and Inference

Training Budget. We mainly follow the notations of Scaling Laws here [29]. For each input sample of length in SFT dataset (NQ, TQ, HQ, WebQSP), we can split it into 3 parts:

$$n_{ctx} = n_Q + n_{Docs} + n_R \tag{11}$$

$$C_{\text{train}} \approx 6 N n_{ctx} N_s \tag{12}$$

where n_Q, n_{Docs}, n_R denotes the length of query, TopK documents and answers respectively. N,N_s denotes the non-embedding parameters and the numbers of samples, which we refer to Chinchilla for calculations. In NQ dataset, $n_Q \approx 15$, $n_{Docs} \approx 478$ and $n_R \approx 2$.

[4]The training set $\widetilde{D}_{\text{train}}$ consists of the original training set $\widetilde{D}^{ori}_{\text{train}}$ and $\widetilde{D}_{\text{aug}} \in \widetilde{D}_{\text{pref}}$ with five query augmentations.

Algorithm 1 Reranker Training

procedure ConstructPreferenceDataset($\widetilde{D}_{\text{train}}$).
  $\widetilde{D}_{\text{pref}} \leftarrow \emptyset$
  From $(q_i, D_{q_i}, y_{q_i}) \in \widetilde{D}_{\text{train}}$, we select the $\widetilde{D}_{\text{sub}} = \{q_i, D_{q_i}^{\text{sub}}, Y_i^{\text{sub}}\}_{i=1}^{N}$.
  for all $\{q_i, D_{q_i}^{\text{sub}}, Y_i^{\text{sub}}\} \in \widetilde{D}_{\text{sub}}$ **do** ▷ Mine Preference Knowledge
    for all $\{d_i | i = 1, 25, 50, 100\} \in D_{q_i}^{\text{sub}}$ **do**
      $a_{\text{LLM}} \leftarrow$ LLM answer to query q_i
      $a_{\text{docs}} \leftarrow$ Correct answer from d_i
      if $a_{\text{LLM}} \neq y_n$ and $a_{\text{docs}} = y_n$ **then**
        $\widetilde{D}_{\text{pref}} \leftarrow \widetilde{D}_{\text{pref}} \cup \{(q_i, D_{q_i}^{\text{sub}}, Y_i^{\text{sub}})\}$ ▷ Aligned Knowledge
        Continue
      else if $a_{\text{LLM}} = y_n$ and $a_{\text{docs}} \neq y_n$ **then**
        $\widetilde{D}_{\text{pref}} \leftarrow \widetilde{D}_{\text{pref}} \cup \{(q_i, D_{q_i}^{\text{sub}}, Y_i^{\text{sub}})\}$ ▷ Unaligned Knowledge
        Continue
      end if
    end for
  end for
  $G_\theta \leftarrow$ Augmented query generator
  $R \leftarrow$ {Complexity, Constraint, SPARQL, Decomposition, Rephrasing}
  for all R_i in R **do**
    for all $(q_i, D_{q_i}) \in \widetilde{D}_{\text{pref}}$ **do**
      $q_{\text{aug},i} \leftarrow G_\theta(R_i, q_i, D_{q_i})$
      $D_{r_i} \leftarrow D_{r_i} \cup \{(q_{\text{aug},i}, D_{q_i}, y_{q_i})\}$
    end for
    $\widetilde{D}_{\text{pref}} \leftarrow \widetilde{D}_{\text{pref}} \cup \left(\cup_{i=1}^{n} D_{r_i}\right)$
  end for
  $p_\Theta \leftarrow$ NLI model for quality filtering
  for all augmented query q_{aug} in $\widetilde{D}_{\text{pref}}$ **do**
    $score_\theta \leftarrow p_\Theta(q, q_{\text{aug}})$
    if $score_\theta$ is not "entailment" **then**
      $\widetilde{D}_{\text{pref}} \leftarrow \widetilde{D}_{\text{pref}} \setminus \{(q_{\text{aug}}, D_{q_i}, y_{q_i})\}$
    end if
  end for
  return $\widetilde{D}_{\text{pref}}$
end procedure
procedure MultiGrainedDistillationAlignment($\widetilde{D}_{\text{pref}}$)
  Initialize model parameters $\theta^{sh}, \theta^1, \ldots, \theta^T$
  repeat
    Compute losses $\mathcal{L}_{\text{CPD}}$, $\mathcal{L}_{\text{FPR}}$, $\mathcal{L}_{\text{SCA}}$
    procedure MGDA-UB($\theta^{sh}, \theta^1, \ldots, \theta^T, c^t$)
      $\mathbf{Z} \leftarrow \sum_{t=1}^{T} c^t \nabla_{\theta^{sh}} \hat{\mathcal{L}}^t(\theta^{sh}, \theta^t)$
      Optimize MTL weights α^t for Pareto optimal solution
      $\mathbf{L} \leftarrow \sum_{t=1}^{T} c^t \hat{\mathcal{L}}^t(\theta^{sh}, \theta^t)$
      return $\mathbf{L}$
    end procedure
    Update model parameters $\theta^{sh}, \theta^1, \ldots, \theta^T$ to minimize $\mathbf{L}$
  until convergence
  return Optimized parameters $\theta^{sh}, \theta^1, \ldots, \theta^T$
end procedure

Algorithm 2 LLM-based Reader Training

procedure PRE-ALIGN($\widetilde{D}_{\text{pref}}$, k)
  for all $\{q_i, D_{\text{pref}}, y_{q_i}\} \in \widetilde{D}_{\text{pref}}$ **do**
    Select one document from D_{pref}
    Randomly select $k-1$ documents from $D = \{d_i\}_{i=1}^{N}$
    Construct Top-k document set $D_{\text{align}} = \{d_{\text{pref}}, d_{\text{rand}_1}, \ldots, d_{\text{rand}_{k-1}}\}$
    Initialize prompt with the selected documents and query
  end for
  Fine-tune the LLMs with the objective $\mathcal{L}(\theta) = \sum_{(q_i, D_{\text{align}}, y_{q_i}) \in \mathcal{D}} \log \mathbf{P}_\theta \left(y_{q_i} | prompt(q_i, D_{\text{align}}) \right)$
end procedure
procedure SUPERVISED FINE-TUNING($\mathcal{D}$, Pre-Aligned Parameters)
  Load pre-warmed parameters from PreAligned stage
  Merge augmented dataset as $\widetilde{D}_{\text{train}} = \widetilde{D}_{\text{train}} \cup (\cup_{i=1}^{n} \widetilde{D}_{r_i})$
  for all $\{q_i, D_{q_i}, y_{q_i}\} \in \widetilde{D}_{\text{train}}$ **do**
    $D_{q_i}^{\text{rank}} \leftarrow$ Top-K [Reranker(q_i, D_{q_i})]
    $\widetilde{D}_{\text{train}}^{\text{rank}} \leftarrow \{(q_i, D_{q_i}^{\text{rank}}, y_{q_i})\}$
  end for
  Perform supervised fine-tuning
end procedure

Methods	Reranker/Filter Model	Reader	Total Training FLOPs	Training Hours	Inference Process
Standard RAG	-	Llama2 (7B)	2.49×10^{17} (1)	0.8 (1)	LLM (1)
DPA-RAG	BGE (125M)	Llama2 (7B)	3.48×10^{17} (2)	1.1 (2)	Reranker (BGE) + LLM (2)
RAG+BGE	BGE (125M)	Llama2 (7B)	2.49×10^{17} (1)	0.8 (1)	Reranker (BGE) + LLM (2)
RAG+RankLlama	Llama2 (7B)	Llama2 (7B)	1.9×10^{18} (5)	7.6 (5)	Reranker (Llama2) + LLM (4)
KnowPAT	-	Llama2 (7B)	5.0×10^{17} (4)	2 (4)	LLM (1)
FILCO	FlanT5 (3B)	Llama2 (7B)	3.55×10^{17} (3)	1.2 (3)	Filter model (FlanT5) + LLM (3)

Table 4: The statistics of Model Types, FLOPs, GPU hours and Inference Process. The numbers in parentheses represent the ranking of resource consumption from lowest to highest.

Therefore, We estimate the FLOPs and GPU times on NQ dataset in Table 1 of Rebuttal PDF.

Inference Budget. Due to the presence of KV cache computations, it is quite difficult to accurately derive the inference FLOPs of different models. Therefore, we quantified the inference steps required by various baselines, which allowed us to roughly rank their inference costs.

Analysis. Following the steps outlined above, we carefully calculate the data size, training FLOPs, training time, and inference costs of different methods in Table 4.

(1) In terms of training budgets, we outperform the KnowPAT, FILCO, and RAG+RankLlama methods, particularly when compared to reranker-based and preference alignment baselines.

(2) For inference, our performance is comparable to the classic RAG+bge reranker-based baseline and significantly exceeds that of other baselines.

These results indicate that the resource expenditure of our dual alignment method is reasonable and does not lead to significant additional resource consumption.

B More Details on Experiment Setup

B.1 Datasets

In this section, we report the detailed information of our 4 datasets, including NaturalQuestions (NQ), TriviaQA (TQA), HotpotQA (HQA), WebQuestionsSP (WebQSP). Table **??** illustrates the statistics of them.

Natural Questions (NQ) [33] dataset, with its approximately 300,000 real Google searches and corresponding answers from Wikipedia, annotated for detailed context and brief replies, is crucial for developing question-answering systems, enhancing AI's comprehension of natural language.

TriviaQA (TQA) [27] serves as a benchmark for QA models, with its extensive set of over 650,000 question-answer pairs sourced from quizzes and trivia competitions. Each question is linked to supporting documents, presenting a challenge for systems to extract correct information from various subjects, which in turn evaluates their information gathering and language comprehension capabilities.

HotpotQA (HQA) [101] dataset comprises 113,000 questions necessitating answers through multi-step logic. It pushes the envelope in AI development by demanding linkage of several documents for inferencing comprehensive answers, aiming to improve AI abilities in complex understanding far exceeding simple fact extraction.

WebQuestionsSP (WebQSP) [103] dataset consists of more than 4,700 Google Suggest-derived questions, each associated with a query in SPARQL format that retrieves answers from the Freebase. It is specifically crafted for refining QA systems' semantic parsing skills and their ability to transform natural language into formal database queries, thereby pushing the boundaries of AI in processing and understanding intricate queries from real-life scenarios.

B.2 Prompt Templates

In the vanilla SFT stage, we follow the template of the RA-Judgement as follow [75]:

Prompt Template of SFT Stage

Given the documents {Top-K Documents}. Answer the following question based on the given information or your internal knowledge with one or few words without the source. Query: {Query}.

For the pre-aligned stage, our prompt template is almost aligned with the SFT stage's template. The only difference is that we add an additional judgment statement that allows the LLMs to distinguish whether the influence of the preference document d_q on answering questions is positive or negative, thereby implicitly learning the ability to distinguish between aligned knowledge and unaligned knowledge. The prompt template is displayed as follow:

Prompt Template of Pre-aligned Stage

Given the documents $\{D_{\text{align}} = (d_q, d_{\text{rand}_1}, \ldots, d_{\text{rand}_{k-1}})\}$. Answer the following question based on the given information or your internal knowledge with few words without the source. Query: $\{q\}$.
[Judgement] document-$\{i_{d_q}\}$ is Positive or Negative knowledge for answering question.

where d_q denotes the preference document that influences the LLM's reasoning results for query q. $\{d_{\text{rand}_1}, \ldots, d_{\text{rand}_{k-1}}\}$ denotes $k-1$ random documents from the retrieved corpus D_{align}. Moreover, i_{d_q} denotes the order of d_q in D_{align}.

For data augmentation process, motivated by the data augmentation process of several works [10, 37, 38, 52, 54, 104, 106], we employ `gpt-3.5-turbo-0613` APIs with a temperature of 1.0. Then we specially design a augmentation prompt for RAG as follow:

Query Augmentation Prompt

You are an AI assistant helping me rewrite the query. I will give you the original query, reference document, title and rewriting requirements. Please rewrite the query based on the following information:

Original Query: {Query}
Reference Documents: {Top-K Documents}
Title: {Title}
Augmentation Requirements: {Augmneted Requirements}
New Queries:

B.3 Implementation Details

Here, we report our detailed information of DPA-RAG, as a retriever-reranker-reader architecture:

For retriever, following the previous works [9, 53], we utilize Dense Document Retriever (DPR) [30] for encoding documents and questions respectively. After that, we use it retrieves the top 100 relevant Wikipedia documents [89] according to the dot-product similarity.

For reranker, we use the BGE [98] as our backbone model. Specifically, we adjust our batch size to 16. We fine-tune our reranker for 10 epochs and set the learning rate to 1e-5. We utilize the BGE reranker to order the top 100 retrieved documents to obtain the top-3 results.[5].

For the QA fine-tuning setting, we employ the AdamW optimizer [50] to train our LLMs for 3 epochs. Moreover, we set our training batch size to 128. We use eight A100 80g GPUs to fine-tune all models with top-3 documents. Our learning rate is set as 7e-5 with a 3% warmup process. For all experiments, we conduct them using the LLaMA Factory framework [113] with model's default system prompts. We use the version 0.6.3[6] for training LLaMA2, Mistral, Qwen1.5 and Phi2. In addition, we use the version 0.8.1 [7] for Qwen2 and LLaMA3. We report the average performance from five experiments, each with a different random seed.

To facilitate the reproduction of our results, all datasets and evaluation benchmarks used in our experiments have been open-sourced and their detailed sources are indicated. We promise to open-source our code after the blind review process.

B.4 Baselines

We mainly compare DPA-RAG with multiple strong baselines by using reranker-based methods and preference aligned methods for RAG as follow:

Reranker-based Baselines:

- **RankGPT** [84] leverages listwise prompting and utilizes specific distillation method to replicate the document reranking abilities of GPT-3.5 within a smaller ranking model.
- **LRL** [57] is a model that utilizes GPT-3.5 as a zero-shot reranker for listwise ranking, which directly generates a ranking list of candidate documents.
- **PRP** [72], Pairwise Ranking Prompting, which involves submitting a query alongside a pair of documents into the prompt, enabling large language models to perform ranking tasks.

[5] we use mDeberta as our filtering model, which can be downloaded at https://huggingface.co/MoritzLaurer/mDeBERTa-v3-base-xnli-multilingual-nli-2mil7
[6] https://github.com/hiyouga/LLaMA-Factory/releases/tag/v0.6.3
[7] https://github.com/hiyouga/LLaMA-Factory/releases/tag/v0.8.1

- **RankLLaMA** [56], based on LLaMA, is trained as a pointwise reranker. This approach involves passing both query and document together to the model. RankLLaMA generates a similarity score reflecting the document's relevance to the query.
- **BGE** [98] is a general Embedding Model developed by BAAI. The reranker use the cross-encoder structure to do full-attention on the input pair.
- **BCEmbedding** [61], Bilingual and Crosslingual Embedding in English and Chinese, developed by NetEase Youdao. Their Reranker is particularly proficient at refining search results and improving ranking tasks.
- **ColBERTv2** [77], a model employs a combination of denoised supervision and residual compression techniques, utilizing token-level decomposition during late interaction.

Preference-aligned Baselines:

- **KnowPAT** [110] is a framework that constructs a knowledgeable preference set to align model preferences with knowledge. This framework effectively guides language models to select relevant knowledge for specific inquiries, enhancing their ability to provide pertinent information.
- **REPLUG** [81] It is a retrieval-enhanced language modeling framework that dynamically optimizes the retriever through the output probability of a black box large language model.
- **RA-Judgement** [75], which is known as Retrieval-augmented judgement. In this work, authors explores the knowledge boundary problem of RAG and proposes two experimental settings, Priori Judgment and Posteriori Judgment. RA-judgment is a dynamic improvement method based on Priori Judgment, which can better capture factual information.
- **RRHF** [107] is a training paradigm, which aims to align probabilities of model responses with human preferences by a ranking loss, which can retain the performance of Proximal Policy Optimization (PPO) and is much simpler.
- **RAFT** [109] boosts a language model's proficiency in answering questions within a specific domain by teaching it to disregard irrelevant documents and reference pertinent segments from retrieved texts. It enhances the model's reasoning capabilities and effectiveness in domain-related tasks while maintaining resilience against incorrect retrievals.
- **FILCO** [93] It is a data selection method based on vocabulary and information theory to improve the quality of generated answers provided to the generative model by filtering useful context in the training data.

Furthermore, We also provide a detailed introduction to the **LLM reader model** used by DPA-RAG:

- **LLaMA2** [86] is an upgraded version of LLaMA developed by MetaAI. It utilizes more robust data cleaning and mixing techniques, and up-samples sources closest to factual information, which can enhance knowledge and reduce hallucinations. Additionally, it employs Grouped-Query Attention technology to lessen reliance on memory.
- **LLaMA3** [59], created by MetaAI, the newest version of the LLaMA series, LLaMA3, includes major enhancements. In contrast to LLaMA2, LLaMA3 incorporates a larger training dataset, extended context length, and an enriched vocabulary, leading to better performance on a range of tasks. Additionally, LLaMA3 offers notable improvements in contextual comprehension and language generation, setting it apart from its predecessor.
- **Qwen1.5** [3] series, created by Alibaba, comprises language models with advanced features like SwiGLU activation, attention QKV bias, group query attention, and a combination of sliding window and full attention mechanisms. These models boast robust fundamental abilities, particularly in language comprehension.
- **Qwen2** [3], developed by Alibaba, is available in several sizes: Qwen2-0.5B /1.5B /7B and 72B. This model is trained on data sources spanning 29 kinds of languages, enabling it to perform exceptionally well in multilingual tasks. Additionally, Qwen2 exhibits strong capabilities in coding and mathematics. Qwen2-72B-Instruct is notable for its ability to handle input windows of up to 128K tokens in length, making it exceptionally well-suited for processing long texts and tackling complex tasks.
- **Mistral** [22], a language model boasting 7 billion parameters, is engineered by Mistral AI for exceptional performance and efficiency. Mistral 7B utilizes Packet Query Attention to accelerate inference and integrates Sliding Window Attention to efficiently manage sequences of varying lengths, all while minimizing inference costs.
  Phi2 [19], proposed by Microsoft, is a powerful small language model with 2.7 billion parameters. Despite its relatively modest size, Phi-2 demonstrates exceptional reasoning and language comprehension capabilities. At its release, it showcased great performance among small foundational LLMs. In different benchmark tests, model's performance was comparable to, or even surpassed, models that are 25 times larger.
- **GPT-3.5 and GPT-4** [65], proposed by OpenAI, which are part of the GPT families that incorporate a multi-step reinforcement learning from human feedback (RLHF) techniques. the algorithm not only enhances the models' instruction-following ability but also significantly reduces the likelihood of producing harmful or toxic content. Moreover, GPT-4 introduces support for image inputs and attains human-like performance on a range of benchmarks.

C More Details about Experimental Results

C.1 Detailed Results for Ablation Studies

Table 6 presents the detailed ablation results of our DPA-RAG across three key phases, with "w/o" indicating the model's version without a particular module. Our findings are as follows:

- DPA-RAG's result declines when any of its components are removed, further validating the necessity of each part we designed.
- Focusing on the Preference Knowledge Construction stage, we notice that the Query Augmentation methods lead to a substantial improvement in performance, which is in line with our expectations. These strategies introduce additional supervision signals during the training stages of both the Reranker and the Reader, yielding a joint boost to the DPA-RAG framework. Moreover, the quality filtering process also brings slight performance gains, underscoring the importance of maintaining intent consistency between original and augmented data.

Table 5: Examples of different methods for generating new queries.

Method	Requirement	Query
Origin	-	What screenwriter with credits for "Evolution" co-wrote a film starring Nicolas Cage and Téa Leoni?
Rephrasing	Rephrase the original query with the same intention.	Who is the screenwriter credited for "Evolution" who also co-authored a movie featuring Nicolas Cage and Téa Leoni?
Decomposition	Decompose the original query into several sub-problems.	Sub-problem 1: Identify the screenwriter who has credits for the film "Evolution". Sub-problem 2: Determine if the screenwriter from sub-problem 1 has also co-written a film where Nicolas Cage and Téa Leoni were cast.
SPARQL	Rewrite the original query based on the SPARQL syntax and generate it directly.	`SELECT ?screenwriter WHERE { ?film rdf:type dbo:Film . ?film dbo:writer ?screenwriter . ?film dbo:starring dbr:Nicolas_Cage . ?film dbo:starring dbr:Tea_Leoni . ?screenwriter dbo:film dbr:Evolution . ?screenwriter rdfs:label ``David Weissman'' . }`
Constraint	Add more conditional and constrained statements to the original query.	Which screenwriter, known for working on the movie "Evolution", also co-authored a screenplay for a feature film that includes Nicolas Cage and Téa Leoni in the cast, and has a history of collaboration with David Diamond?
Complexity	Increase the semantic complexity of the original query.	Which scriptwriter, known for his partnership with David Diamond and shared film credits on "Evolution", also co-authored a screenplay that featured Nicolas Cage and Téa Leoni in leading roles, after initially meeting his writing colleague at Akiba Hebrew Academy and making their screenwriting sale debut with "The Whiz Kid" to 20th Century Fox?

Table 6: Detailed Ablations of LLaMA2-7B on NQ and TQA. Point-wise., Pair-wise., CPA denotes Point-wise, Pair-wise and Contrastive Preference Alignment respectively.

Method	NQ		TQA	
	Hits@1	F1	Hits@1	F1
LLaMA2-7B DPA-RAG	56.03	60.19	70.16	70.29
Preference Knowledge Construction				
w/o Query Aug.	-2.13	-2.31	-2.62	-2.87
w/o Filtering.	-0.92	-0.71	-1.39	-1.45
Multi-Grained Distillation Alignment				
w/o point-wise.	-1.95	-2.12	-2.43	-2.43
w/o pair-wise.	-0.98	-0.92	-1.51	-1.74
w/o CPA	-1.54	-1.12	-1.84	-2.13
w/o MGDA-UB.	-0.52	-0.77	-0.84	-1.10
Knowledge Self-Alignment				
w/o Pre-Align.	-1.72	-1.76	-2.21	-2.45
LLaMA2-7B RAG	50.94	54.76	63.90	63.80

- In the multi-grained distillation alignment stage, each task independently provides stable gains in both NQ and TQA. Point-wise preference alignment, as a fundmental capability for distinguishing knowledge preferences, brings the largest gains in aligning LLMs' preferences. Notably, the MGDA-UB strategy further yields stable gains on top of the joint optimization of three tasks, proving the necessity of introducing multi-task balance optimization.
- The pre-aligned phase also shows steady performance gains, especially evident in TQA. In practice, we find that the potential for internal alignment in TQA is even greater than external, differing from NQ and HQA. Therefore, this insight also highlights the necessity of dual alignment to align datasets from different domains.

C.2 Details about Diverse Query Augmentations

Case Study of Augmented Queries. Table 5 shows some samples which are generated by `gpt-3.5-turbo-0613` APIs in the way of different augmneted requierment, respectively. We can observe that the complexity level of the augmented data showcased in the case is generally consistent with the trend of complexity and diversity scores presented in Table 3.

Tag Review of Training Data. In section "Discussion on Query Augmentations", we initially explore how the performance outcome is linked to complexity and diversity within the Natural Questions (NQ) dataset. Following the Instag [51], we also carry out an review of the intent tags within the training dataset. We randomly selected 10,000 samples from the final Supervised Fine-Tuning (SFT) data pool, which includes both the original data and 5 sets of augmented data. Figure 7 displays the most common tags, which predominantly

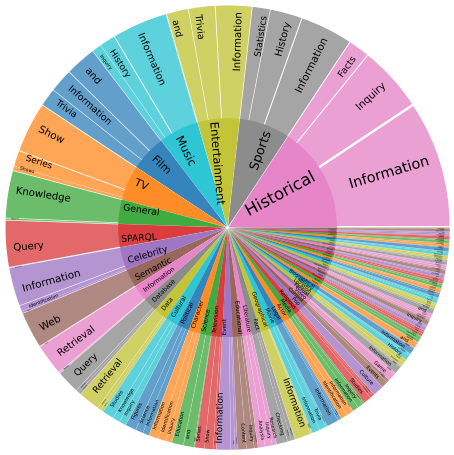

Figure 7: The sunburst chart displays all tags, with each segment representing the first two words of each tag. The size of each segment is proportional to the tag's frequency.

pertain to historical information, sports-related data, and entertainment queries. The tags are represented by the initial two words, and their size is directly proportional to their frequency. We limit our visualization to only those tags that appear more than 600 times within our dataset.

C.3 Case Studies for Preference Alignment

To gain a deeper understanding of how DPA-RAG aligns the preferences of large models to knowledge, we conducted a case study and manual analysis, marking key supporting knowledge and misleading knowledge in the TOP-3 documents with green and red, respectively.

NQ-Case1

Baseline

Query: Where was the diary of a wimpy kid filmed ?

Reference Documents:
document1:
Title: Diary of a Wimpy Kid: Rodrick Rules (film)
Content: The film was released on March 25, 2011. Talks of a sequel were announced after the release of the first, but was not officially announced until May 12, 2010, announcing that it would be released March 25, 2011. Filming took place in Vancouver, British Columbia and New Westminster, British Columbia from August 23 to October 27, 2010. The mall scene was filmed at Park Royal Mall in West Vancouver. Director Thor Freudenthal was replaced by director David Bowers ('Flushed Away" and "Astro Boy"). The film's trailer was shown with "Gulliver's Travels" on December 25, 2010. It was later online on

document2:
Title: Diary of a Wimpy Kid: Dog Days (film)
Content: The outdoor pool scenes at the beginning and the end of the movie were filmed at Steveston Outdoor pool in Richmond, BC. Filming occurred at the Steveston Outdoor pool during the beginning of September 2011. The Chinese Bunkhouse at the Steveston Shipyards in Richmond, BC was the location of the Wilderness Explorers cabin for Troop 133. During filming, stars Zachary Gordon and Robert Capron were seen riding the Corkscrew at Playland at Vancouver's fair, the PNE. A poster was leaked in March 2012. A teaser trailer was attached to "The Three Stooges". An advance screening for the film was held on
document3:
Title: Diary of a Wimpy Kid (film series)
Content: The "Diary of a Wimpy Kid" film series was released on March 25, 2011 and is based on the second book, "Rodrick Rules" with scenes from "The Last Straw". Principal photography began on August 23, 2010, and was completed on October 27, 2010, with filming taking place in Vancouver and New Westminster. "Rodrick Rules" was directed by David Bowers, with Zachary Gordon reprising his role as Greg Heffley. New main characters include Holly Hills (Peyton List), Grandpa (Terence Kelly), and Bill Walter (Fran Kranz). Edward Shearmur composed the original score for the film. "Diary of a Wimpy Kid: Dog Days" is the third film in the series.

Output: New Westminster, British Columbia ✗

****

DPA-RAG

Query: Where was the diary of a wimpy kid filmed ?

Reference Documents:
document1:
Title: Diary of a Wimpy Kid: Dog Days (film)
Content: Outdoor pool scenes at the beginning and the end of the movie were filmed at Steveston Outdoor pool in Richmond, B.C. Filming occurred at the Steveston Outdoor pool during the beginning of September 2011. The Chinese Bunkhouse at the Steveston Shipyards in Richmond, B.C. was the location of the Wilderness Explorers cabin for Troop 133. During filming, stars Zachary Gordon and Robert Capron were seen riding the Corkscrew at Playland at Vancouver's fair, the P.N.E. A poster was leaked in March 2012. A teaser trailer was attached to "The Three Stooges". An advance screening for the film was held on...

document1:
Title: Diary of a Wimpy Kid: Dog Days (film)
Content: All hang out together at the local pool. Greg

describes that his summer did not go the way he expected, but had positive circumstances. This film is the last movie in the "Diary of a Wimpy Kid" film series to feature the original cast, as they aged out of their roles as middle-schoolers. Principal photography began on August 8, 2011, in Vancouver and was completed on October 7, 2011. The location for the country club pool was Eagle Ridge Outdoor pool in Coquitlam, B.C. Filming at Eagle Ridge Outdoor pool took place during the end of August 2011. The municipal...

Title: Diary of a Wimpy Kid (film series)
Content: "Diary of a Wimpy Kid" film series. It was released on March 25, 2011, and is based on the second book, "Rodrick Rules" with scenes from "The Last Straw". Principal photography began on August 23, 2010, and was completed on October 27, 2010, with filming taking place in Vancouver and New Westminster. "Rodrick Rules" was directed by David Bowers, with Zachary Gordon reprising his role as Greg Heffley. New main characters include Holly Hills (Peyton List), Grandpa (Terence Kelly), and Bill Walter (Fran Kranz). Edward Shearmur composes the original score for the film. "Diary of a Wimpy Kid: Dog Days" is the third film...

Output: Vancouver ✓

****

Analysis: The retrieved documents of the baseline contain both aligned knowledge and unaligned knowledge, with the final reasoning being misled by the unaligned knowledge. DPA-RAG filters out the unaligned knowledge during the Reranker process, retaining only the aligned knowledge, leading to successful reasoning in the end.