Unsupervised Large Language Model Alignment for Information Retrieval via Contrastive Feedback

Qian Dong
dq22@mails.tsinghua.edu.cn
DCST, Tsinghua University
Beijing, China

Yiding Liu
liuyiding.tanh@gmail.com
Baidu Inc.
Beijing, China

Qingyao Ai*
aiqy@tsinghua.edu.cn
Quan Cheng Laboratory&
DCST, Tsinghua University
Beijing, China

Zhijing Wu
zhijingwu@bit.edu.cn
School of Computer Science and
Technology
Beijing Institute of Technology
Beijing, China

Haitao Li
liht22@mails.tsinghua.edu.cn
DCST, Tsinghua University
Beijing, China

Yiqun Liu
yiqunliu@tsinghua.edu.cn
DCST, Tsinghua University
Beijing, China

Shuaiqiang Wang
shqiang.wang@gmail.com
Baidu Inc.
Beijing, China

Dawei Yin
yindawei@acm.org
Baidu Inc.
Beijing, China

Shaoping Ma
msp@tsinghua.edu.cn
DCST, Tsinghua University
Beijing, China

ABSTRACT

Large language models (LLMs) have demonstrated remarkable capabilities across various research domains, including the field of Information Retrieval (IR). However, the responses generated by off-the-shelf LLMs tend to be generic, i.e., cannot capture the **distinctiveness** of each document with similar content. This limits the performance of LLMs in IR because finding and distinguishing relevant documents from substantial similar documents is a typical problem in many IR tasks. To address this issue, we propose an unsupervised alignment method, namely *Reinforcement Learning* from *Contrastive Feedback* (**RLCF**), empowering LLMs to generate both high-quality and context-specific responses. Our approach constructs unsupervised contrastive feedback signals based on similar document groups, and adopts a reward function, named group-wise reciprocal rank, to optimize LLMs. We conduct extensive experiments to evaluate the effectiveness of RLCF.

CCS CONCEPTS

• **Information systems** → **Retrieval tasks and goals**.

KEYWORDS

large language models, information retrieval, alignment

ACM Reference Format:
Qian Dong, Yiding Liu, Qingyao Ai, Zhijing Wu, Haitao Li, Yiqun Liu, Shuaiqiang Wang, Dawei Yin, and Shaoping Ma. 2024. Unsupervised Large Language Model Alignment for Information Retrieval via Contrastive Feedback. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '24), July 14–18, 2024, Washington, DC, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3626772.3657689

*Corresponding author.

1 INTRODUCTION

Large Language Models (LLMs) have demonstrated promising performances across a wide range of research fields, including information retrieval (IR). IR aims to fulfil information needs of individuals though locating relevant documents from large-scale corpus [10], which plays a fundamental role in the digital era [8, 21, 62]. Previous studies have utilized LLMs in IR tasks such as generating search snippets [25, 29, 34, 46], conducting query or document expansion [23, 37, 55], creating training data for retrieval models [3, 15, 54], and etc. These studies have demonstrated the significant potential of LLMs in IR.

However, while widely adopted, applying an off-the-shelf LLM directly to IR tasks could be suboptimal. One of key reasons is the misalignment between the capabilities of off-the-shelf LLMs and the needs of IR tasks, particularly the capability to provide responses that capture the **distinctiveness** of each document among similar documents [15, 23]. For instance, Figure 1 show the results of FLAN-T5 [11] when used for document summarization, a representative IR task. As we can see, off-the-shelf FLAN-T5 generates the same summary for three different documents, making them indistinguishable to human. Using such summaries as document representations or search snippets is not acceptable in search engines, as they fail to assist users in quickly and accurately locating their desired documents.

To empower LLMs with desired response attributes, one of the most widely adopted methods is to conduct LLM alignment. The idea of LLM alignment is to construct feedback to LLM's responses
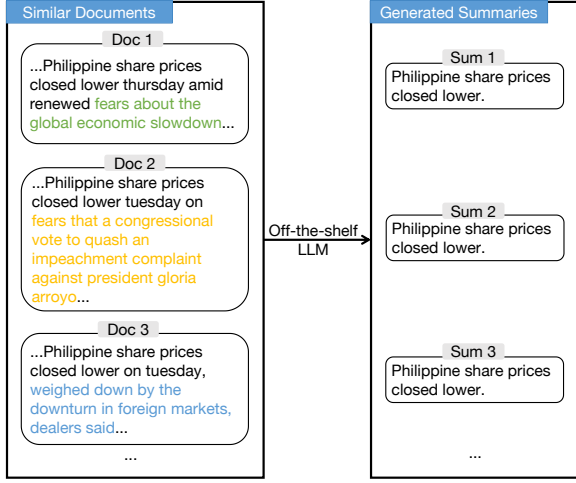
Figure 1: Illustrations of LLMs application in document summarization for similar documents. The distinctive parts of each document are highlighted in different colors.
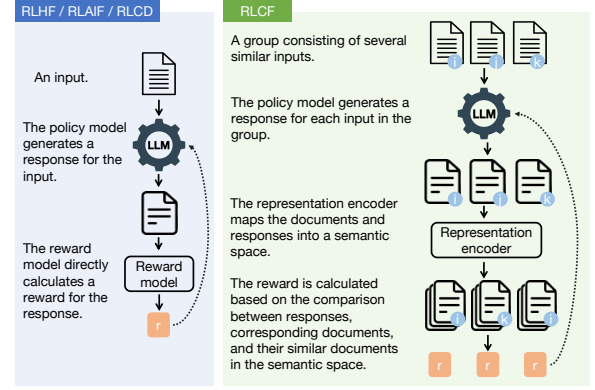

Figure 2: The comparison between existing alignment methods and RLCF. The dotted line represents that the reward score is returned to LLM for PPO optimization.

based on the preferred attributes (e.g., helpfulness [42], harmlessness [14, 31, 61], etc.) and use them to optimize LLMs. Depending on how the feedback is collected, existing LLM alignments can be broadly categorized into human-based methods and model-based methods. Unfortunately, none of the existing alignment methods can be used to improve distinctiveness of response from LLMs.

- While human-based methods such as RLHF [42] are flexible and generic, the cost of collecting large-scale human feedback is expensive and usually prohibitive in IR scenarios.
- While existing model-based methods such as RLAIF [31] and RLCD [61] do not require human in the loop, they are inherently incapable of capturing distinctiveness, as their feedback signals (i.e., reward scores) are computed in a point-wise manner. Specifically, the distinctive information of an LLMs response must be specified in the context of other similar responses, and thus the feedback signals of distinctiveness can only be derived from differentiating multiple responses in a group-wise manner, where existing methods are clearly infeasible.

To address these limitations and improve the distinctiveness of LLMs responses, we propose a novel optimization framework called Reinforcement Learning from Contrastive Feedback (**RLCF**). RLCF is a **group-wise** alignment method that constructs feedback of distinctiveness for a group of similar inputs, and it is fully **unsupervised**. The workflow of RLCF is depicted in Figure 2. Specifically, we first identify a group of similar documents for each document in the corpus. Then, each document in the group is fed to an LLM to obtain a corresponding response. For example, the responses could be summaries of long documents for a summarization task. Next, we construct contrastive feedback from the group of LLM-generated responses, using a reward function named group-wise reciprocal rank. Finally, the LLM is optimized based on the contrastive feedback with the standard Proximal Policy Optimization (PPO) algorithm [49].

To demonstrate the effectiveness of RLCF in optimizing the distinctiveness of LLMs responses, we conduct comprehensive experiments, which include three representative IR tasks, various scales of LLMs (ranging from 770M to 11B) with both encoder-decoder and decoder-only architectures, and both Chinese and English corpora. In particular, we evaluate RLCF on the tasks of document summarization, document expansion and data augmentation. Note that the performances on these tasks are essentially sensitive to the distinctiveness of the LLM-generated responses (e.g., summaries). For document summarization, we directly evaluate the quality of the LLM-generated summaries, measured by human evaluation and an automatic metric called Rouge-diff. To assess the effectiveness of RLCF-optimized LLMs in document expansion and data augmentation, we evaluate the LLMs responses through downstream document retrieval tasks, i.e., comparing the performances of retrieval models enhanced by LLMs with different alignment methods. The results indicate that, compared to existing alignment methods, RLCF can significantly improve the distinctiveness of LLMs responses, which can largely benefit various IR applications [1].

We summarize our main contributions as follows:

- To the best of our knowledge, we are the first to study the distinctiveness of LLMs response and demonstrate its significance in powering IR applications.
- We propose a novel framework namely RLCF which utilizes the contrastive feedback to unsupervised align the capability of LLMs with the needs of IR.
- The experimental results demonstrate the effectiveness of RLCF on both English and Chinese LLMs as well as various parameter scales and architectures.

2 RELATED WORK

2.1 Large Language Models

Recently, LLMs are emerged and boost many natural language processing tasks. The architecture of LLMs, particularly the Transformer [53], leads to significant improvements in capturing textual semantics. This advancement empowers many influential models

---

[1]The codes are released at https://github.com/Deriq-Qian-Dong/RLCF

such as BERT [16] and GPT [43]. These models pave the way for subsequent advancements like GPT-2 [44] and GPT-3 [6], with increasing model sizes and capabilities. The training pipeline of LLMs also earned significant attention in recent years due to its pivotal role in enabling models like GPT to exhibit remarkable language understanding and generation capabilities. Pre-training is a cornerstone of training LLMs and involves training the model on a massive corpus to learn linguistic patterns and structures, leveraging the tasks such as masked language modeling [16], next token prediction [43] and etc. By utilizing large-scale pre-training, LLMs acquire a general understanding of language, making them available for various downstream tasks. Supervised Fine-Tuning (SFT) involves training LLMs on task-specific datasets with labeled examples. This stage adapts the generic linguistic knowledge acquired during pre-training to specific tasks, such as sentiment analysis [19], text classification [18, 20], and dialogues [42]. Alignment technique facilitates LLMs in learning from the generated responses and environmental feedback, thereby aligning the capability with the desired attribute. The environment feedback could be from human [42] or other models [1]. This approach has shown promise in improving the helpfulness and harmlessness of LLMs.

2.2 Alignment for LLM

Alignment techniques, which aim to ensure that language models act in accordance with human values or desired attributes, have garnered significant research attention. This surge of interest is primarily attributed to the widespread proliferation and increasing impact of language models in recent years. In recent years, researchers have increasingly turned their focus to leveraging human feedback as a valuable resource for optimizing language models [2, 51, 69]. Reinforcement learning is typically employed for such optimization, leading to the development of a class of methods referred to as RLHF (Reinforcement Learning from Human Feedback). RLHF [42] leverages human-provided reward signals to guide the training process of language models, enhancing their performance in various natural language generation tasks. Despite the superior performance, a major drawback of RLHF is the requirement for extensive manual labor to provide feedback, making it expensive and time-consuming. Recently, RLAIF [31] utilizes the feedback from the LLM itself to train a reward model, and aligns the capability of LLM in a same manner with RLHF. RLCD [61] utilizes the positive and negative prompts to output preferred responses, thereby training the reward model.

2.3 LLM Applications in IR

Document Summarization. Document summarization is a vital research area in information retrieval. Here, we provide an overview of document summarization. Extractive summarization methods [28, 35, 39, 65] select sentences or phrases directly from the input document to form a summary. Abstractive summarization [4, 48, 50, 60, 66] approach imitates human that comprehends a source document and writes a summary based on the salient concepts of the document Multi-document summarization [5, 7, 64] concentrates on generating concise summaries from a cluster of topic-related documents. Besides, PLMs, such as BART [32], GPT-2 [44], and T5 [45], are also be used for multi-document summarization task.

Language Model for Document Expansion. LLMs are widely used to supplement missing information, thereby mitigating issues associated with data sparsity [57] or information gaps [67]. The vocabulary mismatch between query and document could be effectively alleviate by document expansion using language models [23, 40, 55]. Doc2Query [41] predicts which queries will be issued for a given document and then expands it with those predictions with a vanilla sequence-to-sequence model, trained using datasets consisting of pairs of query and relevant documents. DocT5Query [40], employing T5 [11] as its backbone in the Doc2Query framework, achieves remarkable performance, illustrating that an enhanced backbone results in superior improvements. Doc2Query- [23] illustrates the significance of query quality in document expansion. The research suggests that eliminating low-quality queries can enhance the effectiveness of Doc2Query.

Language Model for Data Augmentation. Data augmentation is an effective strategy to address the challenge of limited training sample sizes. This challenge is especially prominent in the zero-shot learning scenario, which can be viewed as a cold-start problem. Owing to LLMs' superior language comprehension capabilities, they are extensively employed for data augmentation in numerous research fields, like text classification [13], multilingual commonsense reasoning [56], dense retrieval [3, 15, 26, 27, 54] and etc. The distinctiveness of responses from LLMs plays a pivotal role in data augmentation, particularly in dense retrieval. Dense retrieval is trained using query and document pairs, which draws extensive attention from both academia and industry due to its superior performance when applied to the documents it has already been trained on [17, 33, 36]. However, in practical search engines, a large number of new web documents are emerged daily, which often leads to a collapse in the performance of dense retrieval methods with respect to these new documents [15, 52]. Therefore, boosting the zero-shot performance of dense retrieval on new documents is a crucial challenge, in which LLMs play a pivotal role.

3 REINFORCEMENT LEARNING FROM CONTRASTIVE FEEDBACK

In this section, we present the details of our proposed framework, Reinforcement Learning from Contrastive Feedback (RLCF). It facilitate LLMs to capture fine-grained distinctions in similar input documents and output responses that are more distinctive.

3.1 Motivation

Existing alignment methods, such as RLHF [42], RLAIF [31], RLCD [61], etc., have demonstrated effectiveness in adjusting response attributes (e.g., helpfulness). However, these methods can hardly be utilized to enhance the distinctiveness of LLMs response due to two limitations. First, the feedback signals of existing alignment methods are not applicable for our task at hand. In particular, human feedback can be carefully-designed to imply distinctiveness, yet is clearly too expensive to scale. On the other hand, model-based feedback has more manageable cost since it relies on a model but a human to generate accurate feedback. Unfortunately, as shown in our experiments, existing LLMs struggle to provide accurate feedback w.r.t. distinctiveness. Second, the feedback computation in existing alignment methods follows a point-wise input manner.

Table 1: The notations used in this paper.

Notations	Descriptions
q	The query used to search a document.
d	A document from corpus.
$\mathcal{D}$	The corpus of documents.
$\mathbb{G}_d$	The similar documents of document d.
o_d	The response of LLM for document d.
π	The original parameters of LLM.
π_ϕ^{RL}	The optimized RL policy.
$\mathcal{R}$	The reward used to optimize LLM, including the penalty term.
GRR	The reward function, referred to as **g**roup-wise **r**eciprocal **r**ank (GRR).
Inst	An instruction used for response generation.

As shown in the left part of Figure 2, a reward score is computed for a single input in the existing methods, while overlooking the relationships between inputs. Consequently, the subtle distinctions among similar inputs are neglected, which could result in trivial or less informative outputs. Therefore, a natural question is: **How to construct a group-wise feedback with high-quality and low-cost to enhance the distinctiveness of LLMs' responses?**

To answer the question, we propose an unsupervised alignment method, namely *Reinforcement Learning* from *Contrastive Feedback* (**RLCF**). As shown in Figure 3, we first construct data for formulating contrastive feedback, including similar documents identification and response generation. After that, the rest part in this figure outlines the process of optimizing an LLM with the contrastive feedback, which teaches the LLM to identify more distinctive information from a document. The commonly used notations are summarized in Table 1.

3.2 Data Construction

Similar Documents Identification. To facilitate the capacity of LLMs for capturing subtle distinctions among documents, we first need to gather groups of similar documents for computing contrastive feedback. To avoid the high cost of data labeling, we leverage an unsupervised dual-encoder to construct each group of similar documents.

In particular, we randomly select a document d in the corpus $\mathcal{D}$, and retrieve its top-K most similar documents to form the similar documents $\mathbb{G}_d$ of document d, which can be formally defined as

$$\mathbb{G}_{d_i} = \{d_j \mid \underset{\text{top-K}}{\text{argmax}} \, S\left(d_i, d_j\right), \forall d_j \in \mathcal{D}, i \neq j\}, \quad (1)$$

where $S(d_i, d_j)$ denotes the similarity score between d_i and d_j. Taking a standard dual-encoder based dense retriever as an example, the similarity between documents are computed as

$$S(d_i, d_j) = E_{d_i} \otimes E_{d_j}. \quad (2)$$

Here, the $\otimes$ means the inner production operation, and

$$E_d = \text{Avg\_Pooling}\left(\text{M}\left(d\right)\right), \quad (3)$$

which is the average pooling of the last layer's token representations produced by the encoder M.
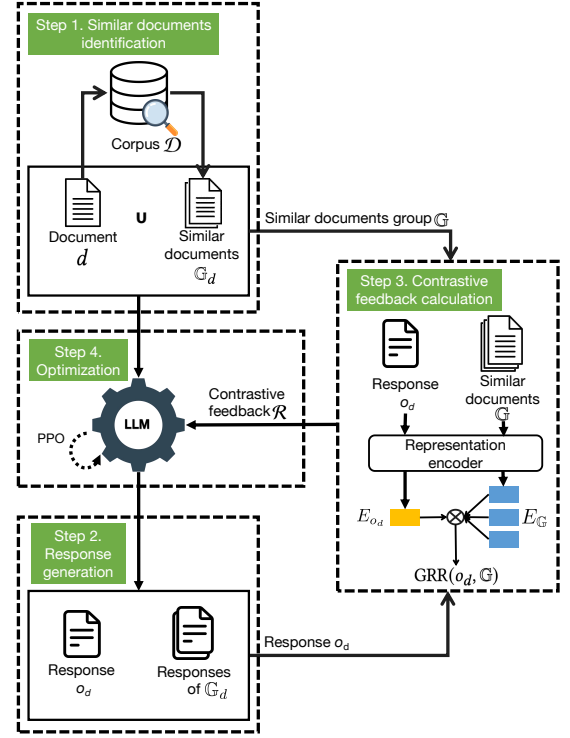

Figure 3: The framework of RLCF. We take the response o_d as an example for illustration of group-wise contrastive feedback calculation. The green and blue rectangles represent the embedding of response and documents, respectively. The $\otimes$ represents the inner production operation between the embedding E_{o_d} and $E_\mathbb{G}$.
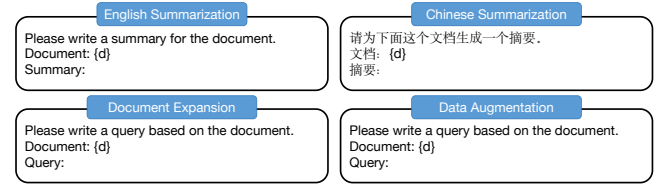

Figure 4: The templates used in our RLCF framework.

Response Generation. Next, for each group of similar documents $\mathbb{G} = \{d \cup \mathbb{G}_d\}$, we use an LLM to generate a response for each document within the group. These response could be a query or the summary w.r.t. the document, or any other desired output.

More specifically, for each document $d \in \mathbb{G}$, we concatenate the pre-defined instruction (denoted as Inst) as the input prefix for d, which can be defined as

$$o_d = \text{LLM}\left(\text{Inst} \oplus d\right), \quad (4)$$

where $\oplus$ represents the concatenation operation. The instruction templates are presented in Figure 4.

3.3 Model Optimization

Contrastive Feedback. To construct contrastive feedback with a group of similar documents, we first use an unsupervised dual encoder, as the representation encoder, to maps the responses and documents into a sematic space, and then compute the similarity between o_{d_i} and each document $d_j \in \mathbb{G} = \{d_i \cup \mathbb{G}_{d_i}\}$. Intuitively, the similarity score between d_i and o_{d_i} would be the highest if LLM captures the key information of d_i that distinguish it from other documents in $\mathbb{G}_{d_i}$. Based on this intuition, we leverage a group-wise reciprocal rank (GRR) to implement contrastive feedback, which can be defined as

$$\text{GRR}(o_{d_i}, \mathbb{G}) = \frac{\lambda}{\sum_{j=1}^{|\mathbb{G}|} \mathbb{I}(S(o_{d_i}, d_i) \leq S(o_{d_i}, d_j))}, \quad (5)$$

where $\mathbb{I}$ is an indicator function and λ is a hyper-parameter. $S(o_{d_i}, d_j)$ represents the similarity score of response o_{d_i} and document d_j. The similarity score is defined in Equation 2.

The computation of contrastive feedback is efficient. Although it necessitates $|\mathbb{G}|^2$ comparisons for the construction of a group-wise feedback, each document in a similar document group $\mathbb{G}$ requires encoding only once. These encoded document representations can then be reused to conduct inner product operations with multiple responses from $\mathbb{G}$. For RLHF [42] and RLAIF [31], with the increase in the size of $\mathbb{G}$, their computational overhead of constructing a group-wise contrastive feedback increase dramatically. Each comparison involves the evaluation of a document-response pair, leading to quadratic computational complexity. Moreover, the expense associated with utilizing either an LLM annotator or a human annotator is considerably higher than that of dense retrieval.

Notably, $\text{GRR}(o_{d_i}, \mathbb{G})$ is computed in a group-wise manner, specifically relying on the group $\mathbb{G}$. Previous alignment techniques utilize a point-wise manner for computing reward scores, relying solely on o_{d_i} and d_i. This is problematic for adapting LLMs for IR because the distinctiveness of responses can only be measured in a group-wise manner. In other words, we cannot determine the distinctiveness of a response for a document unless it is compared with responses generated for other documents.

Optimization. Our objective is to optimize the policy model, i.e., the LLM, using contrastive feedback to generate responses that are desired in the context of IR. We achieve this through reinforcement learning, specifically with the PPO [49] algorithm. We consider the GRR as the reward score for the entire response, and maximize it using the PPO algorithm. Following prior study [42], we also incorporate a term in the reward that penalizes the KL divergence between the optimized RL policy π_ϕ^{RL} with parameters ϕ and the original LLM π. The penalty term prevents the policy model from producing responses that diverge significantly from the vanilla LLM, thereby preserving the language capabilities of the policy model. The full reward $\mathcal{R}$ could be written as

$$\mathcal{R}(d, o_d, \mathbb{G}) = \text{GRR}(o_d, \mathbb{G}) - \beta \log\left[\pi_\phi^{RL}(o_d \mid d)/\pi(o_d \mid d)\right], \quad (6)$$

where β is a hyper parameter that balances the GRR and penalty term.

4 EXPERIMENTAL SETUP

4.1 LLM Applications in IR

There are three popular applications of LLMs in IR, including document summarization [25, 29, 34, 46], document expansion for sparse retrieval [23, 55] and data augmentation for dense retrieval [3, 15]. We evaluate RLCF on the optimizing LLMs for these tasks.

Document Summarization. Document summarization is a direct application of LLMs in many information systems, which is critical for reducing users' cognitive burden. Obviously, the summarization performance highly relies on the distinctive information captured by the summarization model (i.e., LLMs). However, due to a notable lake of distinctiveness in the responses of off-the-shelf LLMs, they often provide indistinguishable summaries for similar documents, such as the example shown in Figure 1.

Document Expansion for Sparse Retrieval. Document expansion is an effective technique for enhancing the performance of sparse retrieval [23, 40, 41], usually via mitigating the vocabulary mismatch between query and document. To facilitate the accuracy of retrieval, the expansion of a document is desirable to be distinctive compared with other similar documents, for which we find out that vanilla LLMs are not well-aligned.

Data Augmentation for Dense Retrieval. For dense retrieval, data augmentation using LLMs is an effective solution for handling out-of-distribution (OOD) documents (e.g., newly-created contents) and scenarios [15], where the labeled data is usually scarce. In particular, existing methods propose to generate synthetic queries for the OOD documents, and train a retrieval model on such synthetic query-document pairs with contrastive learning [9].

4.2 Datasets

Document Summarization. To compare the effectiveness for different alignment methods on document summarization, we perform experiments on two datasets: LCSTS for Chinese and Gigaword for English. **LCSTS** [24] is a widely used dataset employed for Chinese text summarization task. It was created to facilitate research and development in the field of short text summarization. **Gigaword** [50] is extensively utilized in English text summarization research, comprises substantial news articles and their associated headline summaries. This dataset is known for its extensive coverage of diverse topics and its massive size, which makes it a valuable resource for training and evaluating text summarization models. The corpus of LSCTS and Gigaword contain millions of documents. The corpus of other datasets, such as CNN/Daily Mail [38], WikiSum [22] and PubMed [12], are typically of smaller size. Documents in these datasets diverge significantly, which makes them easy to be differentiated and not appropriate to simulate real IR scenarios. Therefore, these datasets are not the primary focus of this paper.

Document Expansion and Data Augmentation for Document Retrieval. We use the **BEIR** [52] to evaluate the effectiveness of data augmentation. BEIR [52] is a widely-used benchmark for document retrieval. It comprises a variety of tasks, including passage retrieval, entity retrieval, fact checking, and others. The dataset includes a broad range of domains, such as medical, finance, and science. NQ [30] is not included in our experiments, because it is presented in the training stage of FLAN-T5 [15].

4.3 Baselines

We utilize two representative LLM alignment methods, namely Reinforcement Learning from AI Feedback(RLAIF) [31] and Reinforcement Learning from Contrast Distillation(RLCD) [61], to evaluate the effectiveness of RLCF. **RLAIF** leverages feedback signals from the LLM itself, thus eliminating the necessity for expensive human annotation. Specifically, the LLM generates two responses to the same input, which are then evaluated by the LLM based on a elaborated instruction for preference annotation. The reward model is subsequently trained using these annotated preference samples. **RLCD** employs both positive and negative prompts to directly generate positive and negative responses respectively, and subsequently trains the reward model using these pairwise samples.

Both RLAIF and RLCD are unsupervised alignment methods. Upon obtaining the reward model, their optimization pipeline is identical to Reinforcement Learning from Human Feedback (RLHF) [42], specifically employing Proximal Policy Optimization (PPO) [59].

4.4 Implementation Details

We only utilize the corpus of summarization datasets (i.e., LCSTS and Gigaword) to conduct RLCF optimization and other model-based LLM alignment methods. Subsequently, inference is directly performed for the optimized LLM, followed by comparative analysis. In our experiments, we employ FLAN-T5 [11] as the backbone of LLMs for English datasets, which is an encoder-decoder architecture. We perform experiments using FLAN-T5 models with 770M, 3B, and 11B parameters, respectively. For the Chinese dataset, we utilize BELLE-7B-2M [63], which is a decoder-only architecture and achieve promising instruction-following ability in Chinese. Although GPT-3.5 and GPT-4 demonstrate superior performance, the undisclosed parameters hinder the training and evaluation of alignment. We use Contriever [26] as the unsupervised dual-encoder M in RLCF. Each document is limited to the first 512 tokens, and any tokens beyond 512 are truncated. The λ in Equation 5 is 10, and the group size used in our experiments is 32, i.e., $|\mathbb{G}|$ is 32. The sparse retrieval method used in document expansion experiments is BM25 [47], implemented by Anserini [2] with default parameters.

To maximize the efficient utilization of GPU memory, we optimize all the parameters in FLAN-T5 with 770M parameters, the last 23 layers of FLAN-T5 with 3B parameters, the last 4 layers of FLAN-T5 with 11B parameters, and the last 12 layers of BELLE-7B-2M, respectively. During text generation, we simply use the greedy decoding strategy.

All experiments are implemented with PyTorch and Huggingface. DeepSpeed with ZeRO stage 2 is utilized for efficient training. All the training and evaluation are conducted on 8 NVIDIA Tesla A100 GPUs (with 40G RAM).

4.5 Evaluation

Automatic Evaluation. For data augmentation of dense retrieval, we directly utilize the traditional metrics of document retrieval, i.e., Mean Reciprocal Rank (MRR), Recall, and Normalized Discounted Cumulative Gain (NDCG). We introduce Rouge-diff as an evaluation metric for document summarization, aimed at assessing the

---

distinctiveness of summaries within similar documents. The Rouge-diff is a variant of Rouge-N, which is defined as

$$\text{Rouge-diff}_{o_{d_i}} = \frac{|set(o_{d_i}) \cap (set(d_i) \setminus set(\cup \mathbb{G}_{d_i}))|}{|set(d_i) \setminus set(\cup \mathbb{G}_{d_i})|}. \quad (7)$$

Here, $set(t)$ represents the tokens of text t after deduplication, and $|set(t)|$ denotes the number of tokens in $set(t)$. Additionally, we report GRR for summarization evaluation, which is defined in Equation (5).

Human Evaluation. To make the evaluation more convincing, we further conduct human evaluation on summaries. Firstly, we randomly sample 200 documents that are not used in the RLCF optimization. Subsequently, we identify the 3 most similar documents for each of these documents, forming 200 groups documents with 4 documents in each group. Finally, we generate summaries for these 200*4 documents using both vanilla LLMs and RLCF-optimized LLMs, as well as GPT-4. We recruit three annotators from a pool of Ph.D. students, each with expertise in areas such as natural language processing and information retrieval. We provide annotation guidelines to our human experts and instruct them to conduct a three-level annotation. The annotation guidelines involve three dimensions: distinctiveness, correctness and concision. *Distinctiveness* refers to the ability of the summary to distinguish itself from similar documents. It requires the summary to highlight unique and critical points that set it apart from other similar documents. *Correctness* represents the accuracy and completeness of the information presented. *Concision* concerns the brevity of the summary. A concise summary effectively conveys the main points of the original document in as few words as possible. The annotation process in RLCF is conducted at the group level, wherein the ultimate decision regarding superior responses is made through comprehensive evaluation. We conduct comparisons between the vanilla LLM and the RLCF-optimized LLM, as well as between the RLCF-optimized LLM and GPT-4. During the process of annotation, two LLMs are randomly designated as LLM A and LLM B to avoid bias. The annotator's task is to assess the quality of the summaries and determine which one is superior. The decision is based on the above aspects of the summaries, i.e., distinctiveness, correctness, and concision.

5 EXPERIMENTAL RESULTS

5.1 Document Summarization

For Chinese document summarization, we employ BELLE-7B-2M, and for English document summarization, we utilize FLANT5-3B as the initial parameters of LLMs. We conduct both automatic evaluation and human evaluation for document summarization.

Automatic Evaluation. We randomly select 512 documents that are not used in the RLCF optimization to form the initial test set. Subsequently, the four documents most similar to each document in the initial test set are retrieved by a dual-encoder, thereby extending the initial test set and making the evaluation challenging. As a result, the final test set consists of 2048 documents.

The experimental results are presented in Table 2. From this table, we can draw the following findings:

- RLCF optimization significantly improves the Rouge-diff on the test set, demonstrating its effectiveness on document summarization in IR context.

- RLCF optimization leads to significant improvements on both Chinese and English datasets, highlighting its effectiveness across different languages as well as various parameter scales and architectures.
- GPT-3.5 and GPT-4.0 demonstrate superior performance in generating distinctive responses compared to publicly available LLMs. However, as the parameters of GPT-3.5 and GPT-4.0 have not been released, conducting RLCF optimization experiments on them is currently unfeasible.
- Model-based alignment methods (i.e., RLAIF and RLCD) cannot outperform GPT-3.5 and GPT-4.0 on LCSTS, owing to the inherent linguistic capabilities of the Chinese LLM BELLE-7B-2M, which are comparatively weaker than those of the English LLM FLAN-T5. The unreliable feedback constrains the effectiveness of model-based alignment methods.
- In comparison to point-wise based alignment methods, RLCF significantly outperforms them, underscoring the effectiveness of the group-wise manner.

Human Evaluation. We also incorporate human evaluation in our experiments. The settings of human evaluation are presented in Section 4.5. The evaluation results are presented in Figure 5. From this figure, we can draw the following conclusions:

- Responses generated by the RLCF-optimized LLMs contain more distinctive information than those produced by vanilla LLMs, making them more suitable for IR scenarios.
- Gigaword's results are superior to those of LCSTS. This discrepancy can be attributed to Gigaword's larger corpus size (3.8 million vs. 2.4 million) and the higher degree of similarity among its documents, making it has more similar documents to construct contrastive feedback.
- The RLCF-optimized LLMs exhibit performance on par with GPT-4 on both Chinese and English datasets. Given the massive scale of parameters in GPT-4, the effectiveness of RLCF is remarkable, as the LLMs used in our experiments have only 3-7 billion parameters.
- GPT-4 slightly outperforms the RLCF-optimized LLM on the Chinese dataset. This discrepancy could be attributed to the inherent disparities in the fundamental capabilities of LLMs. The publicly available English LLM is superior to its Chinese counterpart.
- We noticed an inconsistency between the results of human evaluation and automatic evaluation. Despite that RLCF optimized LLMs achieve more superior automatic metrics, responses generated by GPT-4 sufficiently satisfy users' requirements in document summarization task, yielding human evaluation scores comparable to those of RLCF-optimized LLMs. Nevertheless, we believe that automatic metrics can reveal more nuanced performance differences than human evaluation can. In the task of document summarization, it is challenging for humans to judge performance accurately.

5.2 Document Expansion for Sparse Retrieval

Document expansion is an effective technique to alleviate the vocabulary mismatch, thus improving the performance of sparse retrieval. Due to constraints in computational resources, we only generate five queries per document for the purpose of document

Table 2: Experimental results of document summarization on LCSTS and Gigaword. Significant improvement or degradation w.r.t. vanilla LLM is indicated (+/-) (p-value$\leq$0.05).

Model	LCSTS		Gigaword	
	Rouge-diff	GRR	Rouge-diff	GRR
GPT-3.5	23.8	90.9	15.5	78.9
GPT-4.0	25.6	90.9	17.6	78.9
Vanilla LLM BELLE/FLAN-T5	22.1	90.4	11.9	75.2
w/ RLAIF	22.7	90.6	18.5	77.3
w/ RLCD	23.4	90.8	19.3	78.2
w/ RLCF	**32.2**[+]	**91.7**[+]	**32.5**[+]	**80.9**[+]

(a) LCSTS

(b) Gigaword
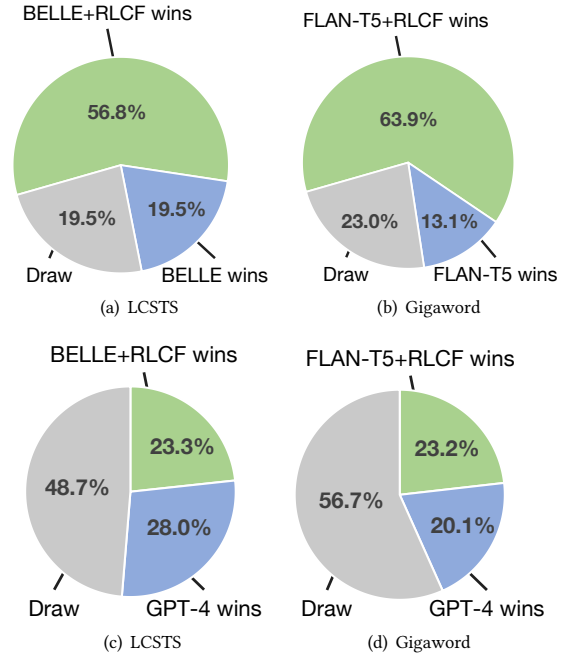

(c) LCSTS

(d) Gigaword

Figure 5: The results of human evaluation.

expansion, and employ BM25 [47] to assess the effectiveness of the expanded contents, and we restricted our document expansion experiments to seven datasets from BEIR [52], each containing fewer than one million documents. The generation of five queries per document for a dataset comprising millions of documents requires a significant investment of GPU hours, amounting to thousands.

The effectiveness of document expansion is influenced by inherent characteristics of the dataset. Therefore, we incorporate statistical information, including the average length of documents and intra-list similarity, into Table 3. We divided this table into two groups based on the average document length to facilitate the analysis. Intra-list similarity (ILS) [58, 68] is defined as

$$\text{ILS}_{\mathcal{D}} = \frac{\sum_{i=1}^{|\mathcal{D}|} \sum_{j=i+1}^{|\mathcal{D}|} S(d_i, d_j)}{\sum_{i=1}^{|\mathcal{D}|} \sum_{j=i+1}^{|\mathcal{D}|} 1}, \tag{8}$$

where $S(d_i, d_j)$ is the similarity score between d_i and d_j.

Table 3: Experimental results of sparse retrieval methods on BEIR. Significant improvement w.r.t. the best baseline is indicated (+) (p-value$\leq$0.05). DocExp is the abbreviation for document expansion. The metrics used in this table is NDCG@10.

| Dataset	#Avg. Doc. Length200					#Avg. Doc. Length200			
	FiQA	SCIDOCS	ArguAna	TREC-COVID	Avg.	SciFact	NFCorpus	Touche	Avg.
ILS	1.27	1.29	1.32	1.49	1.34	1.21	1.26	1.39	1.28
BM25	.162	.133	.231	.456	.246	.666	.318	.489	.491
DocExp FLAN-T5(3B)	.215	.138	.253	.563	.292	.668	**.325**	.489	.494
w/ RLAIF	.208	.138	.259	.560	.291	.671	.318	.489	.493
w/ RLCD	.207	.137	.253	.562	.290	.669	.319	.494	.494
w/ RLCF	**.216**	**.139**	**.270$^+$**	**.582$^+$**	**.302**	**.674$^+$**	.320	**.498$^+$**	**.497**
Improv.	0.5%	0.7%	4.2%	3.4%	3.4%	0.4%	-	0.8%	0.6%

The experimental results are shown in Table 3. Compared with document expansion from off-the-shelf LLM, the average BM25 performance is improved from .246 to .292 when the average document length is less than 200. This highlights that the problem of mismatched vocabulary is more prominent in shorter documents. Thus, shorter documents are better suited for assessing the effectiveness of various alignment techniques in document expansion. Besides, DocExp w/ RLCF outperforms other methods on six out of seven datasets, with notable excellence on ArguAna and TREC-COVID. These two datasets exhibit higher ILS scores. A heightened ILS score signifies increased similarity among documents in the corpus, indicating a more pronounced inclination for distinctive expanded contents. This demonstrates that as the similarity among documents within the corpus grows, LLMs increasingly depend on RLCF to enhance their performance.

5.3 Data Augmentation for Dense Retrieval

The experimental results of BEIR are presented in Table 4. Notably, the settings used in this table are entirely zero-shot. To facilitate the evaluation of data augmentation's quality, we utilize the BERT-based-uncased [16] as the initial parameters of dense retrieval. From this table, we can draw the following findings:

- Across all datasets, RLCF-optimized LLMs consistently outperform other alignment methods in data augmentation for dense retrieval. This demonstrates the effectiveness of our RLCF framework for aligning the capability of LLMs with the data augmentation in IR.
- Compared with DataAug without any alignment, point-wise alignment methods, RLAIF and RLCD, do not exhibit significant improvements. This underscores the superiority of group-wise contrastive feedback used in RLCF.

To further analyze the impact of different scales of LLMs on data augmentation, we conduct an analysis of scaling LLMs on MS-MARCO. We use LLMs with 770 million, 3 billion, and 11 billion parameters in the experiment. The results are illustrated in Figure 6, which demonstrates that RLCF can consistently improve the performance of data augmentation with different parameter scales. Notably, the metrics we employ are MRR@10 and Recall@1000, as these are widely used metrics in the MS-MARCO benchmark. Additionally, as depicted in Figure 6, the RLCF-optimized LLMs outperform the LLMs with equivalent parameters.
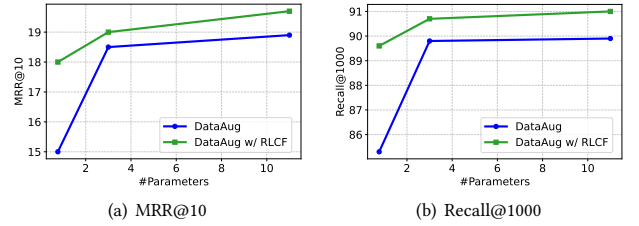
Figure 6: Scaling law of LLMs on the application of data augmentation. The x-axis is parameter scale in billions.
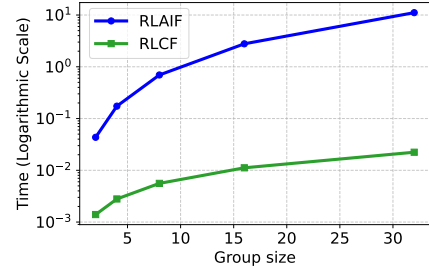
Figure 7: The relationship between the inference time and group size for RLAIF and RLCF. The y-axis represents the logarithmic scale of time (seconds) for better visualization.

5.4 Efficiency Analysis

In this section, we conduct an analysis of the group-wise feedback computation inference times. As depicted in Figure 7, the inference time of RLAIF exhibits an approximate exponential increase with the growth of the group size. The time overhead in RLCF is primarily lies in encoding documents and responses, while the computational cost of inner product calculations is negligible. Notably, the LLM used in Figure 7 for RLAIF is FLAN-T5-3B, while the dual encoder used in RLCF is Contriever with only 110 million parameters. When the group size is 32, RLAIF requires a time overhead of 11.3 seconds, utilizing 6.2 GB of additional GPU memory for group-wise feedback computation. Upon increasing the parameters of the LLM to 7 billion, the time overhead escalates to 50.9 seconds, and additional GPU memory usage rises to 14.3 GB. As a comparison, RLCF requires only an additional GPU memory overhead

Table 4: Experimental results of dense retrieval methods on BEIR. Significant improvement w.r.t. the best baseline is indicated (+) (p-value≤0.05). DataAug is the abbreviation of data augmentation.

Dataset(→)	ArguAna	FiQA	NFCorpus	SCIDOCS	SciFact	COVID	Touche	DBPedia	HotpotQA	Fever	MARCO	Avg.
Method(↓)						NDCG@10						
DataAug FLAN-T5(3B)	.166	.152	.144	.073	.339	.452	.155	.173	.335	.292	.133	.219
w/ RLAIF	.162	.159	.144	.072	.345	.467	.145	.183	.348	.291	.128	.222
w/ RLCD	.125	.144	.153	.065	.352	.473	.112	.166	.340	.314	.122	.215
w/ RLCF	.210$^+$	.168$^+$	.155	.081$^+$	.379$^+$	.502$^+$	.164$^+$	.192$^+$	.390$^+$	.347$^+$	.144$^+$	.245
Improv.	26.5%	5.7%	1.3%	11.0%	7.7%	6.1%	5.8%	4.9%	12.1%	10.5%	8.3%	10.4%
						Recall@100						
DataAug FLAN-T5(3B)	.795	.444	.184	.203	.708	.068	.279	.280	.526	.574	.814	.442
w/ RLAIF	.789	.448	.197	.207	.737	.071	.281	.298	.527	.579	.809	.445
w/ RLCD	.724	.427	.191	.185	.717	.072	.238	.299	.525	.601	.792	.434
w/ RLCF	.844$^+$	.450	.202$^+$	.209	.744$^+$	.076$^+$	.304$^+$	.319$^+$	.564$^+$	.646$^+$	.834$^+$	.471
Improv.	6.2%	0.4%	2.5%	1.0%	0.9%	5.6%	9.2%	6.7%	7.0%	7.5%	2.5%	5.8%
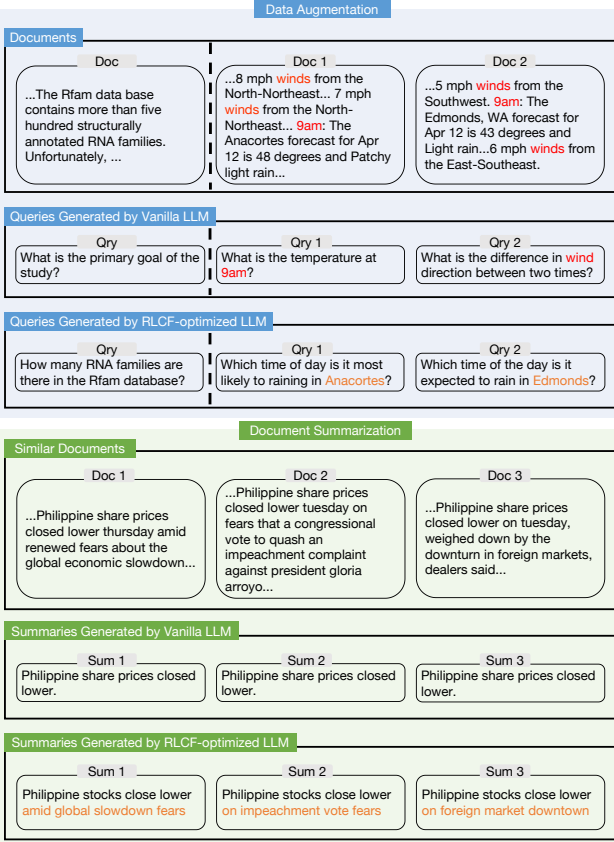
Figure 8: The cases of responses generated by vanilla LLMs and RLCF-optimized LLMs for highly similar documents.

of 0.25GB and a time overhead of 2.2e-3 seconds. Consequently, RLCF is an efficient alignment method for adopting LLMs to IR.

5.5 Case Study

In this subsection, we present several cases to illustrate the effectiveness of RLCF, as shown in Figure 8. Since LLMs are employed for generating queries in both document expansion and data augmentation, we represent these two applications solely by the use of "Data Augmentation" in Figure 8. In the first case of data augmentation, the query generated by vanilla LLMs could even match all documents. In the second case, despite the generated queries being relatively more relevant to the documents, they still lack distinctiveness. A query generated by vanilla LLMs for one document can still match another document.

For the task of document summarization, despite that the generated summarizes are accurate for individual documents by vanilla LLMs, they are not suitable within the pipeline of IR. As shown in Figure 8, after RLCF optimization, the summaries generated by the LLMs not only precisely summarize the main idea of the document, but also capture the specific part of documents.

6 CONCLUSION

In this work, we propose a novel framework, namely RLCF, that leverages contrastive feedback to optimize large language models. Specifically, we construct groups of similar documents by an unsupervised dual encoder model, and then use an LLM to generate a response for each document. Next, we leverage the generated response to construct a contrastive feedback for the LLM optimization, which is implemented by a group-wise reward function, i.e., group-wise reciprocal rank. By doing this, the LLM can be optimized via PPO algorithm to be aligned with the requirements of information retrieval. We conduct experiments on three popular applications of LLMs in information retrieval, demonstrating the effectiveness of RLCF.

ACKNOWLEDGMENTS

REFERENCES

[1] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073* (2022).

[2] Florian Böhm, Yang Gao, Christian M Meyer, Ori Shapira, Ido Dagan, and Iryna Gurevych. 2019. Better rewards yield better summaries: Learning to summarise without references. *arXiv preprint arXiv:1909.01214* (2019).

[3] Luiz Bonifacio, Hugo Abonizio, Marzieh Fadaee, and Rodrigo Nogueira. 2022. In-Pars: Data Augmentation for Information Retrieval using Large Language Models. *arXiv preprint arXiv:2202.05144* (2022).

[4] Mrinmoi Borah, Pankaj Dadure, Partha Pakray, et al. 2022. Comparative analysis of T5 model for abstractive text summarization on different datasets. (2022).

[5] Arthur Bražinskas, Mirella Lapata, and Ivan Titov. 2019. Unsupervised opinion summarization as copycat-review generation. *arXiv preprint arXiv:1911.02247* (2019).

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[7] Ziqiang Cao, Furu Wei, Li Dong, Sujian Li, and Ming Zhou. 2015. Ranking with recursive neural networks and its application to multi-document summarization. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 29.

[8] Jia Chen, Yiqun Liu, Yan Fang, Jiaxin Mao, Hui Fang, Shenghao Yang, Xiaohui Xie, Min Zhang, and Shaoping Ma. 2022. Axiomatically Regularized Pre-training for Ad hoc Search. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1524–1534.

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*. PMLR, 1597–1607.

[10] Anfeng Cheng, Yiding Liu, Weibin Li, Qian Dong, Shuaiqiang Wang, Zhengjie Huang, Shikun Feng, Zhicong Cheng, and Dawei Yin. 2023. Layout-aware Webpage Quality Assessment. *arXiv preprint arXiv:2301.12152* (2023).

[11] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416* (2022).

[12] Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. *arXiv preprint arXiv:1804.05685* (2018).

[13] Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Zihao Wu, Lin Zhao, Wei Liu, Ninghao Liu, Sheng Li, Dajiang Zhu, et al. 2023. Chataug: Leveraging chatgpt for text data augmentation. *arXiv preprint arXiv:2302.13007* (2023).

[14] Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong Yang. 2023. Safe rlhf: Safe reinforcement learning from human feedback. *arXiv preprint arXiv:2310.12773* (2023).

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

[16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[17] Qian Dong, Yiding Liu, Qingyao Ai, Haitao Li, Shuaiqiang Wang, Yiqun Liu, Dawei Yin, and Shaoping Ma. 2023. I^3 Retriever: Incorporating Implicit Interaction in Pre-trained Language Models for Passage Retrieval. *arXiv preprint arXiv:2306.02371* (2023).

[18] Qian Dong, Yiding Liu, Suqi Cheng, Shuaiqiang Wang, Zhicong Cheng, Shuzi Niu, and Dawei Yin. 2022. Incorporating Explicit Knowledge in Pre-trained Language Models for Passage Re-ranking. *arXiv preprint arXiv:2204.11673* (2022).

[19] Qian Dong and Shuzi Niu. 2021. Latent Graph Recurrent Network for Document Ranking. In *International Conference on Database Systems for Advanced Applications*. Springer, 88–103.

[20] Qian Dong and Shuzi Niu. 2021. Legal judgment prediction via relational learning. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 983–992.

[21] Qian Dong, Shuzi Niu, Tao Yuan, and Yucheng Li. 2022. Disentangled Graph Recurrent Network for Document Ranking. *Data Science and Engineering* 7, 1 (2022), 30–43.

[22] Mehrdad Farahani. 2020. Summarization using Bert2Bert model on WikiSummary dataset. https://github.com/m3hrdadfi/wiki-summary.

[23] Mitko Gospodinov, Sean MacAvaney, and Craig Macdonald. 2023. Doc2Query−: When Less is More. In *European Conference on Information Retrieval*. Springer, 414–422.

[24] Baotian Hu, Qingcai Chen, and Fangze Zhu. 2015. Lcsts: A large scale chinese short text summarization dataset. *arXiv preprint arXiv:1506.05865* (2015).

[25] Jie Huang and Kevin Chen-Chuan Chang. 2023. Citation: A key to building responsible and accountable large language models. *arXiv preprint arXiv:2307.02185* (2023).

[26] Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118* (2021).

[27] Pengyue Jia, Yiding Liu, Xiangyu Zhao, Xiaopeng Li, Changying Hao, Shuaiqiang Wang, and Dawei Yin. 2023. Mill: Mutual verification with large language models for zero-shot query expansion. *arXiv preprint arXiv:2310.19056* (2023).

[28] Huan Yee Koh, Jiaxin Ju, Ming Liu, and Shirui Pan. 2022. An empirical survey on long document summarization: Datasets, models, and metrics. *ACM computing surveys* 55, 8 (2022), 1–35.

[29] Bonan Kou, Muhao Chen, and Tianyi Zhang. 2023. Automated Summarization of Stack Overflow Posts. *arXiv preprint arXiv:2305.16680* (2023).

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

[31] Harrison Lee, Samrat Phatale, Hassan Mansoor, Kellie Lu, Thomas Mesnard, Colton Bishop, Victor Carbune, and Abhinav Rastogi. 2023. Rlaif: Scaling reinforcement learning from human feedback with ai feedback. *arXiv preprint arXiv:2309.00267* (2023).

[32] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461* (2019).

[33] Haitao Li, Qingyao Ai, Jia Chen, Qian Dong, Yueyue Wu, Yiqun Liu, Chong Chen, and Qi Tian. 2023. SAILER: Structure-aware Pre-trained Language Model for Legal Case Retrieval. *arXiv preprint arXiv:2304.11370* (2023).

[34] Nelson F Liu, Tianyi Zhang, and Percy Liang. 2023. Evaluating verifiability in generative search engines. *arXiv preprint arXiv:2304.09848* (2023).

[35] Yang Liu. 2019. Fine-tune BERT for extractive summarization. *arXiv preprint arXiv:1903.10318* (2019).

[36] Yiding Liu, Weixue Lu, Suqi Cheng, Daiting Shi, Shuaiqiang Wang, Zhicong Cheng, and Dawei Yin. 2021. Pre-trained Language Model for Web-scale Retrieval in Baidu Search. *arXiv preprint arXiv:2106.03373* (2021).

[37] Guangyuan Ma, Xing Wu, Peng Wang, Zijia Lin, and Songlin Hu. 2023. Pretraining with Large Language Model-based Document Expansion for Dense Passage Retrieval. *arXiv preprint arXiv:2308.08285* (2023).

[38] Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023* (2016).

[39] Shashi Narayan, Shay B Cohen, and Mirella Lapata. 2018. Ranking sentences for extractive summarization with reinforcement learning. *arXiv preprint arXiv:1802.08636* (2018).

[40] Rodrigo Nogueira, Jimmy Lin, and AI Epistemic. 2019. From doc2query to docTTTTTquery. *Online preprint* 6 (2019), 2.

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

[42] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems* 35 (2022), 27730–27744.

[43] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training. (2018).

[44] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog* 1, 8 (2019), 9.

[45] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* 21, 140 (2020), 1–67.

[46] Elizabeth Reid. 2023. Supercharging search with Generative AI. https://blog.google/products/search/generative-ai-search/

[47] Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval* 3, 4 (2009), 333–389.

[48] Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. *arXiv preprint arXiv:1509.00685* (2015).

[49] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[50] Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368* (2017).

[51] Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. 2020. Learning to

summarize with human feedback. *Advances in Neural Information Processing Systems* 33 (2020), 3008–3021.

[52] Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models. *arXiv preprint arXiv:2104.08663* (2021).

[53] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*. 5998–6008.

[54] Kexin Wang, Nandan Thakur, Nils Reimers, and Iryna Gurevych. 2021. Gpl: Generative pseudo labeling for unsupervised domain adaptation of dense retrieval. *arXiv preprint arXiv:2112.07577* (2021).

[55] Liang Wang, Nan Yang, and Furu Wei. 2023. Query2doc: Query Expansion with Large Language Models. *arXiv preprint arXiv:2303.07678* (2023).

[56] Chenxi Whitehouse, Monojit Choudhury, and Alham Fikri Aji. 2023. LLM-powered Data Augmentation for Enhanced Crosslingual Performance. *arXiv preprint arXiv:2305.14288* (2023).

[57] Likang Wu, Zhi Zheng, Zhaopeng Qiu, Hao Wang, Hongchao Gu, Tingjia Shen, Chuan Qin, Chen Zhu, Hengshu Zhu, Qi Liu, et al. 2023. A Survey on Large Language Models for Recommendation. *arXiv preprint arXiv:2305.19860* (2023).

[58] Xiaohui Xie, Qian Dong, Bingning Wang, Feiyang Lv, Ting Yao, Weinan Gan, Zhijing Wu, Xiangsheng Li, Haitao Li, Yiqun Liu, et al. 2023. T2Ranking: A large-scale Chinese Benchmark for Passage Ranking. *arXiv preprint arXiv:2304.03679* (2023).

[59] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval. *arXiv preprint arXiv:2007.00808* (2020).

[60] Hemant Yadav, Nehal Patel, and Dishank Jani. 2023. Fine-Tuning BART for Abstractive Reviews Summarization. In *Computational Intelligence: Select Proceedings of InCITe 2022*. Springer, 375–385.

[61] Kevin Yang, Dan Klein, Asli Celikyilmaz, Nanyun Peng, and Yuandong Tian. 2023. RLCD: Reinforcement Learning from Contrast Distillation for Language Model Alignment. *arXiv preprint arXiv:2307.12950* (2023).

[62] Shenghao Yang, Yiqun Liu, Xiaohui Xie, Min Zhang, and Shaoping Ma. 2022. Enhance Performance of Ad-hoc Search via Prompt Learning. In *China Conference on Information Retrieval*. Springer, 28–39.

[63] Yan Gong Yiping Peng Qiang Niu Baochang Ma Xiangang Li Yunjie Ji, Yong Deng. 2023. BELLE: Bloom-Enhanced Large Language model Engine. https://github.com/LianjiaTech/BELLE.

[64] Xin Zheng, Aixin Sun, Jing Li, and Karthik Muthuswamy. 2019. Subtopic-driven multi-document summarization. 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)*. 3153–3162.

[65] Qingyu Zhou, Nan Yang, Furu Wei, Shaohan Huang, Ming Zhou, and Tiejun Zhao. 2018. Neural document summarization by jointly learning to score and select sentences. *arXiv preprint arXiv:1807.02305* (2018).

[66] Youchao Zhou, Heyan Huang, and Zhijing Wu. 2023. Boosting legal case retrieval by query content selection with large language models. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*. 176–184.

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

[68] Cai-Nicolas Ziegler, Sean M McNee, Joseph A Konstan, and Georg Lausen. 2005. Improving recommendation lists through topic diversification. In *Proceedings of the 14th international conference on World Wide Web*. 22–32.

[69] Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593* (2019).