Mitigating Source Bias with LLM Alignment

Sunhao Dai
Yuqi Zhou
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China
{sunhaodai,yuqizhou}@ruc.edu.cn

Liang Pang
CAS Key Laboratory of AI Safety
Institute of Computing Technology
Chinese Academy of Sciences
Beijing, China
pangliang@ict.ac.cn

Zhuoyang Li
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China
easonzyli@gmail.com

Zhaocheng Du
Gang Wang
Huawei Noah's Ark Lab
Shenzhen, China
{zhaochengdu,wanggang110}@huawei.com

Jun Xu[*]
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China
junxu@ruc.edu.cn

Abstract

Recent studies have revealed a phenomenon known as source bias, where PLM-based retrievers assign higher relevance scores to LLM-generated content despite its semantic quality being comparable to human-written content. As LLMs rapidly advance and become more widely used, effectively counteracting source bias is crucial for the sustainable development of the information retrieval (IR) ecosystem. Existing methods primarily attempt to address source bias from the retriever side, adopting a "passive defense" approach that intervenes only after biased content has entered the retrieval pipeline. These solutions are limited by frequent retriever updates in industrial applications, high recurring costs, and their inability to address the root cause of source bias.

In this paper, we propose a new perspective for mitigating source bias by actively aligning LLM outputs at the data generation stage. Specifically, we introduce LLM-SBM, a novel LLM alignment framework for source bias mitigation. First, we construct high-quality alignment datasets using an automatic preference data construction pipeline. This pipeline leverages LLMs to generate multiple rephrasings of content and employs a PLM-based retriever to assign corresponding specific preference values for each generated document, thereby forming preference pairs according to these preferences. Moreover, to fully utilize these scalar values of preference and enhance the efficiency of the alignment process, LLM-SBM incorporates these preference differences as weighting factors in the loss function during policy training. Extensive experiments across multiple datasets and PLM-based retrievers demonstrate that LLMs aligned with LLM-SBM successfully reduce source bias while preserving their general capabilities.

CCS Concepts

• **Information systems** → **Information retrieval**.

Keywords

Source Bias, LLM Alignment, Information Retrieval

ACM Reference Format:
Sunhao Dai, Yuqi Zhou, Liang Pang, Zhuoyang Li, Zhaocheng Du, Gang Wang and Jun Xu. 2025. Mitigating Source Bias with LLM Alignment. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '25), July 13–18, 2025, Padua, Italy.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3726302.3730038

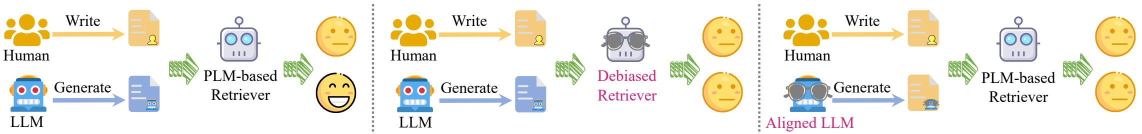
1 Introduction

The recent emergence of large language models (LLMs) [41], which have demonstrated remarkable capabilities in automatically generating human-like text at scale, has led to a significant influx of AI-generated content on the Internet [1, 6, 35]. This surge has fundamentally reshaped information retrieval (IR) systems originally designed to index and retrieve human-written content in response to users' queries. Now, these systems face the challenge of managing corpora that include both human-written and LLM-generated content [5, 8, 39]. Amidst this new landscape, a critical issue termed **source bias** has emerged, referring to the tendency of mainstream pre-trained language model (PLM)-based retrievers to favor LLM-generated content by often ranking it higher than human-written content even when their semantic quality is comparable [6, 8, 39].

A significant concern arising from source bias is its potential to destroy the content ecosystem of existing IR systems. Specifically, as shown in Figure 1(a), many users may rely on LLMs for content creation–for example through prompts for paraphrasing or rewriting, akin to content spinning or plagiarism. Due to source bias, the LLM-rewritten content with similar semantics to original human-written content is more likely to be ranked higher by the platform's retrieval algorithms, gaining increased exposure and traffic. Over time, human-authored original content may become less discoverable, discouraging human creators from creating new material. Ultimately, the content creation platform risks becoming inundated with AI-generated content, potentially leading to several

[*] Corresponding author

SIGIR '25, Padua, Italy
© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-1592-1/2025/07
https://doi.org/10.1145/3726302.3730038



(a) Illustration of Source Bias (b) Mitigating from Retriever Side (c) Mitigating from LLM Side (Ours)

Figure 1: Illustration of source bias and different perspectives for mitigating source bias.

issues, such as the loss of user experience, a decline in content diversity, and the collapse of model performance [8, 25, 39].

To mitigate source bias, existing methods have predominantly focused on debiasing from the retriever's perspective [8, 39], aiming to construct more robust retrieval models that ensure fair and unbiased ranking results under the mixed corpora (Figure 1(b)). This approach acts as a "passive defense", addressing the issue only after biased content has already entered the retrieval pipeline. While effective to some extent, retriever-side methods often face practical challenges: they require frequent re-application as retrieval models are updated—often daily or even hourly in industrial applications—leading to substantial recurring costs. Moreover, source bias often arises unintentionally, as users rely on platform-provided or open-source LLMs for tasks such as paraphrasing or rewriting content. This generated content, when fed into IR systems, inadvertently amplifies source bias, favoring LLM-generated outputs over human-written content. Given this scenario, addressing source bias proactively at the source—the LLM—becomes both a practical and fundamental approach. By "actively" aligning LLMs before their release, we can ensure that the outputs they generate are less likely to introduce or exacerbate source bias in downstream PLM-based retrievers. Crucially, LLM-side solutions involve only a one-time adjustment before the LLM's release, reducing ongoing costs and simplifying the mitigation process. Therefore, aligning LLMs to counteract source bias represents a fundamental solution approach, which is also unexplored and remains an open question.

To this end, we aim to align the outputs of LLMs to ensure that texts generated by LLMs are no longer easily assigned higher estimated relevance scores by PLM-based retrievers, thereby mitigating source bias at its origin, as shown in Figure 1(c). However, directly applying existing LLM alignment methods, such as widely-used Direct Preference Optimization (DPO) [23], to alleviate source bias presents several challenges. **First**, since our objective is to align LLM outputs with the preference values (relevance scores) of IR models, we cannot rely on human annotators to label preference data as is common in typical alignment tasks. Therefore, harnessing feedback from IR models to automate the construction of alignment datasets poses a significant challenge. **Second**, unlike human-annotated preference pairs, which only consider the ordinal relationship (i.e., that the chosen response is preferred over the rejected one) [16, 33], the preference values provided by IR models are precisely quantifiable. Effectively leveraging these fine-grained preferences is crucial for achieving better alignment.

Considering the above issues, we propose a preference-aware **LLM** alignment framework for **S**ource **B**ias **M**itigation, named **LLM-SBM**. Initially, we design an automatic preference data construction pipeline tailored to collect high-quality alignment data using feedback from PLM-based retrievers. This pipeline comprises three main stages: leveraging LLMs with diverse rewriting prompts to generate paraphrased texts, receiving preference values from IR models, and constructing alignment pair samples based on the preference value. Through this process, we develop a high-quality alignment dataset specifically targeting source bias with fine-grained preference differences. Furthermore, to better align LLMs with the constructed fine-grained preference datasets, LLM-SBM incorporates these preference value differences as weights in the loss function during policy training, thereby enhancing the efficiency during the LLM alignment process. Gradient analysis demonstrates that LLM-SBM can also effectively mitigate noise issues inherent in automatically generated preference data. Notably, LLMs aligned with our LLM-SBM framework effectively mitigate source bias in LLM-generated content without compromising their general capabilities. Therefore, we advocate for the widespread adoption of LLM-SBM as a general post-training method by both open-source LLM providers and API service providers before the LLM is released, providing a robust solution to mitigate source bias without sacrificing original quality and versatility.

Our main contributions are summarized as follows:

- To the best of our knowledge, this is the first work that proposes mitigating source bias from the perspective of aligning LLMs, providing a fundamental solution from the data generation side that complements existing retriever-side methods.
- We propose LLM-SBM, a novel framework tailored for mitigating source bias, which includes an automatic preference data construction pipeline and a preference-aware policy training method.
- Extensive experiments verify the effectiveness of the proposed LLM-SBM framework in mitigating source bias in LLM-generated content across various IR datasets, PLM-based retrievers, and LLMs.

2 Related Work

Source Bias in Information Retrieval. The rapid advancement of large language models (LLMs) has fueled the expansion of AI-generated content (AIGC) on the internet, leading information retrieval (IR) systems to increasingly handle corpora that encompass both human-written and AI-generated content [1, 5–7, 35]. Dai et al. [8] first revealed that mainstream neural retrievers based on pre-trained language models (PLMs) exhibit a preference for LLM-generated content, a phenomenon referred to as *source bias*. Beyond document retrieval, other works further discovered that this bias also exists in text-image retrieval [39] and video retrieval [10], where retrieval models prefer AI-generated images and videos. These studies attribute the cause of source bias to the complex coupling between LLMs and retrievers—such as similar Transformer-based architectures and pretraining paradigms—which leads to neural retrievers capturing specific patterns embedded in LLM-generated content [8, 10, 39, 44]. Subsequent research has further explored source bias in other IR scenarios, including question answering (QA) [27], retrieval-augmented generation (RAG)[2], and recommender systems (RS) [44]. These works further emphasize the widespread existence of source bias and highlight the importance and urgency of addressing this emerging issue in this LLM era. Moreover, Wang et al. [31] provided a causal explanation for source bias, showing that PLM-based retrievers favor LLM-generated content due to the positive correlation between language modeling and retrieval objectives. To counteract source bias, recent studies have introduced various debiasing constraints into the training objectives of neural retrieval models [8, 39, 44]. These methods aim to correct the skewed estimated relevance score of PLM-based retrievers that favor LLM-generated content, thereby ensuring that the results are not biased toward LLM-generated content. Different from the above approaches that focus on mitigating source bias from the retriever side, our work is the first to address source bias by actively aligning LLMs, which offers a more fundamental solution than passively defending from the retriever side.

Large Language Models Alignment. Empowering LLMs to better understand and follow human instructions—thereby producing responses that align more closely with human expectations—has garnered significant attention in the research community [16, 17, 33]. Early efforts in this direction include OpenAI's exploration of Supervised Fine-Tuning (SFT), resulting in representative works such as InstructGPT [22]. To further enhance alignment with human preferences, a prominent approach named Reinforcement Learning from Human Feedback (RLHF) has been introduced [3, 22, 26, 45], which involves training a reward model on human-annotated outputs and subsequently using it to fine-tune LLMs through reinforcement learning. Despite the remarkable effectiveness of RLHF in aligning LLMs with human preferences, its complex training pipeline can lead to optimization instability [40, 42]. To address these challenges, methods represented by Direct Preference Optimization (DPO) [23] enable the direct optimization of preferences without the need for training a separate reward model, thereby significantly reducing training complexity. This approach has shown practical success and has been widely adopted in new-generation LLMs, such as Llama3 [9]. In our work, we aim to apply DPO to align the outputs of LLMs with the preference values from IR models, thereby addressing the source bias problem. Unlike alignment using human-annotated preference pairs, we focus on designing alignment strategies that leverage fine-grained preference data obtained from IR models, especially for the source bias mitigation problem.

3 Preliminary

3.1 Retrieval with Mixed-Source Corpora

In the LLM era, both human-written and LLM-generated documents coexist within the corpus. Formally, let C^H represent the set of human-written documents and C^G denote the set of LLM-generated documents, where each document $d^G \in C^G$ is generated by a LLM while preserving nearly the same semantic information as its human-written counterpart $d^H \in C^H$. Given a query $q \in Q$, the goal of a retriever is to return the top-K relevant documents $\{d^{(1)}, d^{(2)}, \ldots, d^{(K)}\}$ from the mixed-source corpora $C = C^H \cup C^G$. Specifically, the retriever evaluates each query-document pair (q, d) by assigning an estimated relevance score $\hat{r}(q, d)$ and ranks the documents accordingly, from highest to lowest score.

Then, we define *source bias*: given a query q, and two documents d^H and d^G that are semantically similar, PLM-based retrievers exhibit source bias by assigning a higher estimated relevance score to the LLM-generated document, i.e., $\hat{r}(q, d^G) > \hat{r}(q, d^H)$.

3.2 Task Formulation

In this paper, our objective is to mitigate source bias of text generated by LLMs, ensuring that the rewritten texts do not receive preferential treatment in PLM-based retrieval systems. Formally, given a human-written document d^H, a target LLM π_θ, and a rewriting prompt x, the LLM can be prompted to generate a document $d^G := \pi_\theta(x, d^H)$. Our goal is to achieve the following:

- **Mitigate Source Bias**: The generated document d^G should be ranked comparably or lower than the original document d^H by PLM-based retrievers.
- **Maintain Quality**: The generated document d^G should preserve the semantic integrity and overall quality of the original document d^H, ensuring that the general capabilities of the LLM are not compromised.

In this way, we seek to align the outputs of LLMs with the inversed feedback (lower relevance scores) from retrieval models, thereby effectively counteracting the tendency for LLM-generated content to be favored by PLM-based retrievers.

3.3 Direct Preference Optimization (DPO)

Direct Preference Optimization (DPO) is an offline reinforcement learning approach designed to directly optimize a policy using preference data without the need for reward models or online sampling. In essence, DPO seeks to maximize the margin between the log-likelihoods of preferred responses and rejected ones while ensuring that the model remains close to its initial policy.

DPO consists of two core components:

- **Preference Data:** The training process utilizes a preference dataset $\mathcal{D}$, where each element (x, y_c, y_r) consists of a prompt x, a chosen (preferred) response y_c, and a rejected (non-preferred) response y_r. Each element is also associated with a preference ranking indicating that y_c is preferred over y_r (i.e., $y_c \succ y_r \mid x$), based on evaluations from human annotators.
- **Policy Training:** The objective function for DPO is defined by minimizing the empirical binary cross entropy (BCE) loss over the preference dataset:

$$\mathcal{L}_{\text{DPO}}(\pi_\theta, \pi_{\text{ref}}, \mathcal{D}) =$$
$$-\sum_{(x, y_c, y_r) \in \mathcal{D}} \log \sigma \left(\beta \log \frac{\pi_\theta(y_c \mid x)}{\pi_{\text{ref}}(y_c \mid x)} - \beta \log \frac{\pi_\theta(y_r \mid x)}{\pi_{\text{ref}}(y_r \mid x)} \right), \quad (1)$$

where $\sigma(\cdot)$ denotes the sigmoid function, β is a scaling factor that adjusts the influence of the preference feedback. The term $\frac{\pi_\theta(y|x)}{\pi_{\text{ref}}(y|x)}$ represents an implicit reward defined by current policy π_θ and referenced policy π_{ref}. This loss function allows for effective discrimination between preferred and non-preferred actions, ultimately enhancing the quality of the generated responses in alignment with human expectations without deviating significantly from the reference policy π_{ref}.

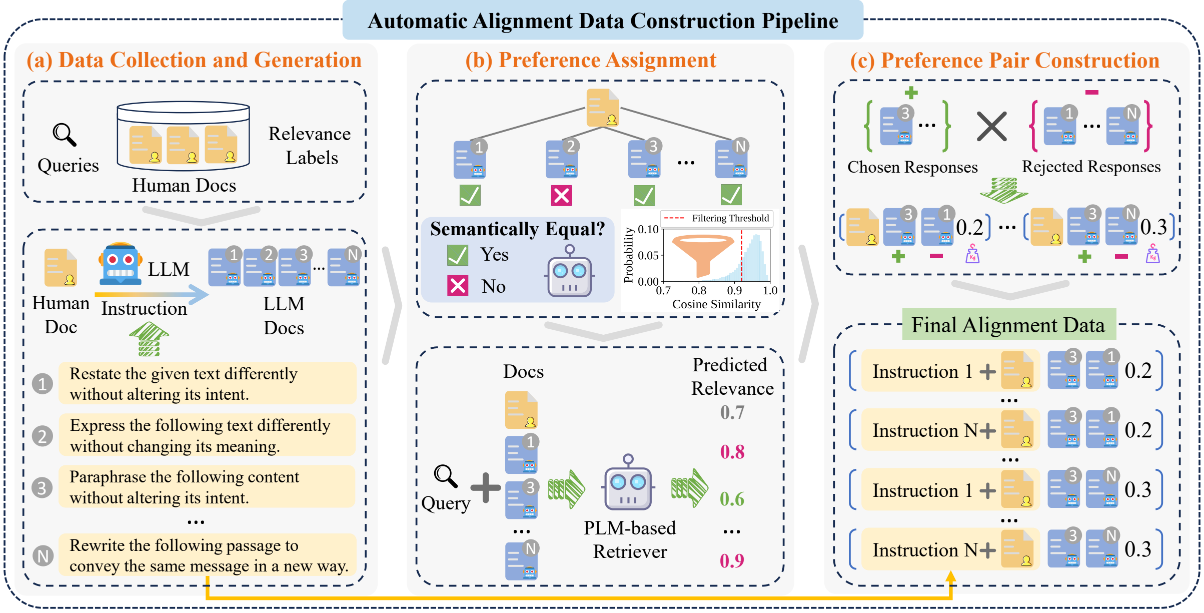

Automatic Alignment Data Construction Pipeline
(a) Data Collection and Generation
Queries
Human Docs
Relevance Labels
Human Doc
LLM
Instruction
LLM Docs
Restate the given text differently without altering its intent.
Express the following text differently without changing its meaning.
Paraphrase the following content without altering its intent.
Rewrite the following passage to convey the same message in a new way.
(b) Preference Assignment
Semantically Equal?
Yes
No
Filtering Threshold
Probability
Cosine Similarity
Docs
Query
PLM-based Retriever
Predicted Relevance
0.7
0.8
0.6
0.9
(c) Preference Pair Construction
Chosen Responses
Rejected Responses
0.2
0.3
Final Alignment Data
Instruction 1 +
Instruction N+
0.2
0.2
0.3
0.3

Figure 2: An overview of the automatic alignment data construction pipeline. (a) Data Collection and Generation: we simulate user interactions with LLMs for rewriting tasks multiple times with diverse instructions to generate several semantically similar documents. (b) Preference Assignment: we employ a specific PLM-based retriever to assign preference scores to all the documents after filtering. (c) Preference Pair Construction: we construct preference pairs for alignment data by identifying documents that are ranked below and above human-written content according to the assigned preference.

4 Our Approach: LLM-SBM

4.1 Overview

As illustrated in Figure 1, source bias arises from the complex coupling between LLMs and PLM-based retrievers. Addressing this issue from the retriever side serves only as a passive defense and has several drawbacks. Therefore, we adopt a different perspective to counteract source bias by aligning the outputs of LLMs with the inverse preference values of IR models, representing a more fundamental approach. In this way, we aim to reduce the likelihood that LLM-generated texts are ranked higher by PLM-based retrievers without sacrificing the general capabilities of LLMs.

To achieve this, we design a novel LLM alignment framework for source bias mitigation, consisting of two key components:

- **Automatic Preference Data Construction Pipeline**: This pipeline automatically generates high-quality alignment datasets by leveraging LLMs to rewrite human-written content with diverse rephrasing instructions and assigning fine-grained preference values to these rewrites using PLM-based retrievers. By constructing preference pairs based on these values, the pipeline produces alignment data that captures nuanced differences in alignment targets.
- **Preference-Aware Policy Training**: With the constructed alignment data, we propose a preference-aware optimization method that incorporates fine-grained preference differences as weights in the training loss. This approach allows the model to prioritize reliable data samples while mitigating the impact of noisy or ambiguous data, resulting in robust alignment.

In the following sections, we first detail how our proposed automatic preference data construction pipeline can generate high-quality alignment data with fine-grained preference differences. Subsequently, we introduce the preference-aware policy training method, accompanied by gradient analysis to highlight its effectiveness in leveraging fine-grained preferences and mitigating noise.

4.2 Automatic Alignment Data Construction

To effectively mitigate source bias by aligning LLM outputs, we require high-quality alignment data. Note that our alignment goal is to align with the preference value of the IR model rather than human values. Consequently, we need to construct alignment data based on feedback from IR models rather than human annotations. To achieve this, we propose an automatic data construction pipeline designed to generate this alignment data efficiently. As illustrated in Figure 2, our pipeline comprises three key components: data collection and generation, preference assignment, and preference pair construction. In the following, we provide the details of each stage of the construction pipeline.

4.2.1 Data Collection and Generation. To construct alignment data, we first collect a representative human-written corpus and then simulate real user interactions with LLMs to generate multiple semantically similar LLM-generated documents. These LLM-generated documents are subsequently used for constructing preference pairs.

We first collect queries and corresponding human-written documents from the MS MARCO dataset [21] to simulate a real-world IR environment. MS MARCO is a widely used benchmark in the IR field [19, 28], containing a large number of real user search queries derived from Bing search logs and documents from diverse domains. Many open-source PLM-based retrievers are trained on this dataset [13, 36, 38]. To ensure data quality, we utilize the filtered and processed MS MARCO dataset provided by the Cocktail benchmark[1] [5], which includes approximately 540K human-written documents. We denote this human-written corpus as C^H.

To generate LLM-generated documents, we employ one of the most advanced open-source large language models, Llama3[2], with various rephrasing instructions to rewrite each human-written document in C^H, ensuring that the semantic content remains unchanged. Inspired by recent studies highlighting the importance of instruction diversity [4, 32, 34], we construct N (e.g., $N = 6$ in our experiments [3]) different common rephrasing instructions, examples of which are shown in Figure 2(a). These instructions enhance the diversity of the generated data and better simulate real-world scenarios where users employ LLMs for text rewriting. Using these N instructions, we obtain LLM-generated corpora $\{C_1^G, \cdots, C_N^G\}$, where C_i^G is the corpus generated using the i-th instruction.

4.2.2 Preference Assignment. With the multiple LLM-generated corpora $\{C_1^G, \cdots, C_N^G\}$, we then filter out low-quality data and utilize a PLM-based retriever to assign preference scores.

Specifically, to ensure that the rewritten content maintains semantic equivalence with the original human-written text and to avoid introducing noise or bias, we employ a widely adopted sentence embedding model BGE[4] [37] to compute the semantic similarity between each human-written document d^H and its different corresponding LLM-generated documents d_i^G. As shown in Figure 2(b), the similarity distribution indicates that the majority of similarities exceed 0.9. To further enhance data quality, we set a higher filtering threshold of 0.92 to exclude low-quality rewritten texts. This filtering process results in a refined LLM-generated corpus, denoted as C^G.

With the mixed-source corpus $C = C^H \cup C^G$, we proceed to construct preference data based on the estimated relevance scores assigned by a specific PLM-based retriever. Specifically, we utilize SBERT [24] checkpoint from Cocktail benchmark [5], to evaluate each query-document pair (q, d) and assign an estimated relevance score $\hat{r}(q, d)$ as the preference value. Despite the presence of source bias causing the retriever to generally favor LLM-generated content, the sensitivity of LLMs to different instructions results in some LLM-generated documents receiving lower relevance scores than their human-written counterparts. This variability allows us to construct meaningful preference pairs that reflect the nuanced differences in relevance as determined by the PLM-based retriever.

4.2.3 Preference Pair Construction. Using the preference scores assigned to LLM-generated documents by the PLM-based retriever, we compare these scores with those of the corresponding human-written documents to construct the final preference pairs.

Table 1: Statistics of the final constructed alignment data. Avg. Chosen Doc and Avg. Rejected Doc represents the average chosen LLM-generated documents and rejected LLM-generated documents per human-written doc, respectively.

# Sample	# Human Doc	Avg. Chosen Doc	Avg. Rejected Doc
10,830	627	1.70	1.87

Specifically, as seen in Figure 2(c), we construct preference pairs based on the following criterion: For a given query q and human-written document d^H, if there exist two LLM-generated rewrites d_i^G and d_j^G such that $\hat{r}(q, d_i^G) < \hat{r}(q, d^H) < \hat{r}(q, d_j^G)$, we form a preference quadruple $(d^H, d_i^G, d_j^G, \delta)$, where d_i^G is the preferred response, d_j^G is the non-preferred response, and $\delta = \hat{r}(q, d_j^G) - \hat{r}(q, d_i^G)$ represents the preference difference. Our goal is to align the LLM to generate outputs similar to d_i^G, which are less favored by the retriever, thereby mitigating source bias.

Considering the implementation of N rephrasing instructions designed to produce semantically equivalent outputs, we further expand the alignment dataset by pairing each instruction with the constructed preference quadruples, thereby rapidly increasing the dataset size by a factor of N. This approach not only increases the volume of the alignment data but also enhances its diversity and the generalization ability of the alignment method to unseen instructions out of the training alignment data.

4.2.4 Data Statistics. Based on our proposed pipeline, we finally constructed 6 (i.e., $N = 6$) different rephrasing instructions to perform the rewriting tasks on the human-written documents from MS MARCO. This resulted in generating about 6 × 540K LLM-generated documents. After filtering and utilizing the validation set of MS MARCO along with a PLM-based retriever, we constructed $6 \times 1,805 = 10,830$ alignment data samples for LLM alignment training, each of which is quadruples (x, y_c, y_r, δ), where x is the rephrasing instruction concatenated with the human-written content to be rewritten, y_c is the chosen (preferred) response, y_r is the rejected (non-preferred) response, and δ is the fine-grained preference difference provided by the IR model. Note that δ is normalized to a range of 0 to 1 using min-max normalization.

From the statistics shown in Table 1, we can observe that the Avg. Rejected Doc is greater than the Avg. Chosen Doc, which indicates that in the LLM-generated documents (from unaligned LLM), there are more documents ranked higher than the corresponding human-written documents. This observation further verifies the existence of source bias.

4.3 Preference-aware Policy Training

4.3.1 Preference-aware Loss. With the constructed dataset $\mathcal{D}$ containing fine-grained preference differences, the standard DPO method cannot distinguish between preference pairs based on the degree of their preference differences. Treating all pairs equally may result in suboptimal performance, as it ignores the varying degrees to which one response is preferred over another. To better align LLMs with the constructed fine-grained preference datasets, our proposed LLM-SBM introduces a simple yet effective modification

[1] https://huggingface.co/datasets/IR-Cocktail/msmarco
[2] https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct
[3] We utilize similar rephrasing instructions from https://github.com/kevinamiri/Instructgpt-prompts to construct the alignment data for fine-tuning.
[4] https://huggingface.co/BAAI/bge-base-en-v1.5

policy training method of DPO, which utilizes the preference difference δ as weights in the loss function, allowing the model to prioritize adjustments based on the degree of the difference.

Formally, let δ quantify the preference difference for the pair $y_c \succ y_r$, indicating the degree to which the preferred response y_c is better than the non-preferred response y_r. We incorporate δ as a coefficient for each response pair in the standard DPO loss function:

$$\mathcal{L}_{\text{LLM}-\text{SBM}}(\pi_\theta, \pi_{\text{ref}}, \mathcal{D}) = \\ - \sum_{(x, y_c, y_r, \delta) \in \mathcal{D}} \delta^\alpha \cdot \log \sigma \left(\beta \log \frac{\pi_\theta(y_c \mid x)}{\pi_{\text{ref}}(y_c \mid x)} - \beta \log \frac{\pi_\theta(y_r \mid x)}{\pi_{\text{ref}}(y_r \mid x)} \right), \tag{2}$$

where $\alpha \geq 0$ is a hyper-parameter that controls the influence of the preference difference on the LLM-SBM loss. By adjusting α, we can control the extent to which larger preference differences influence the model's learning process. In particular, when $\alpha = 0$, the preference difference δ has no effect on the loss function, and LLM-SBM loss in Eq. (2) reduces to the standard DPO loss in Eq. (1).

4.3.2 *Gradient Analysis of LLM-SBM Loss.* To gain a deeper understanding of the mechanics of LLM-SBM, it is insightful to analyze the gradient of its loss function $\mathcal{L}_{\text{LLM}-\text{SBM}}$ with respect to the parameters θ. Following the derivation in Appendix A.4 in DPO [23], the gradient of our LLM-SBM can be written as

$$\nabla_\theta \mathcal{L}_{\text{LLM}-\text{SBM}} = \\ - \sum_{(x, y_c, y_r, \delta) \in \mathcal{D}} w_\theta \Big[\underbrace{\nabla_\theta \log \pi_\theta (y_c \mid x)}_{\text{increase likelihood of } y_c} - \underbrace{\nabla_\theta \log \pi_\theta (y_r \mid x)}_{\text{decrease likelihood of } y_r} \Big], \tag{3}$$

where the weights w_θ are defined as

$$w_\theta = \beta \cdot \underbrace{\delta^\alpha}_{(a)} \cdot \underbrace{\left[\sigma \left(\beta \log \frac{\pi_\theta(y_r \mid x)}{\pi_{\text{ref}}(y_r \mid x)} - \beta \log \frac{\pi_\theta(y_c \mid x)}{\pi_{\text{ref}}(y_c \mid x)} \right) \right]}_{(b)}. \tag{4}$$

Intuitively, this formulation of LLM-SBM gradient can be dissected into three key components:

- Direction of Improvement: In Eq. (3), similar to standard DPO, the gradient of the loss function increases the likelihood of the preferred responses y_c and decreases the likelihood of the non-preferred responses y_r.
- Preference Signal Magnitude: In Eq. (4), the part (b) of the weighting term w_θ amplifies gradient contributions when the current predictions of the policy deviate from the desired preference ordering.
- Preference-aware Weighting: The key difference in our method lies in part (a) of the weighting term w_θ in Eq. (4), where we introduce the additional preference-aware factor $\delta^\alpha (\alpha > 0)$. Here, $\delta = \hat{r}(x, y_r) - \hat{r}(x, y_c) > 0$ represents the preference difference learned from the retriever model. A larger preference differences indicates the preferred response is clearly better than the non-preferred one, suggesting a more reliable (less noisy) sample. Conversely, smaller differences suggest that the preference signal is weaker and potentially noisy.

In standard DPO ($\alpha = 0$), every sample contributes equally, increasing the risk of overfitting to noisy examples. In contrast, by incorporating δ^α, our LLM-SBM effectively leverages the external preference value differences from the IR model (i.e., the alignment target), which allows the model to assign higher weights to samples with larger preference differences (i.e., cleaner samples where the preferred response is significantly better than the non-preferred one) and lower weights to samples with smaller preference differences (which may be noisy samples). As a result, LLM-SBM focuses more on learning from reliable data, thereby reducing the risk of overfitting to noisy samples in alignment data.

5 Experiments

In this section, we conduct experiments to evaluate the effectiveness of the LLM-SBM framework. The constructed alignment data and code are available in https://github.com/KID-22/LLM-SBM.

5.1 Experimental Settings

5.1.1 *Datasets.* Following previous works [5, 8, 31], we conduct experiments on three widely-used IR datasets from varying domains, including SciFact [30], NQ [18], and TREC-COVID [29]. For all datasets, we use the filtered and processed corpus provided by the Cocktail benchmark [5].

5.1.2 *Evaluation Protocols and Metrics.* Following standard practices [5, 8, 39], we adopt the RelaDiff to quantify source bias, which represents the relative percentage difference in NDCG scores between human-written and LLM-generated content:

$$\text{RelaDiff} = \frac{\text{NDCG}_{\text{Human}} - \text{NDCG}_{\text{LLM}}}{(\text{NDCG}_{\text{Human}} + \text{NDCG}_{\text{LLM}})/2} \times 100\%,$$

where the $\text{NDCG}_{\text{Human}}$ and NDCG_{LLM} denote the NDCG scores corresponding to human-written and LLM-generated content, respectively [8, 39]. RelaDiff > 0 implies that the given IR model ranks human-written content higher than LLM-generated content, while RelaDiff < 0 indicates the opposite trend. The RelaDiff metric ranges from -200% to 200%, with smaller values indicating a more severe source bias favoring LLM-generated content. Therefore, our goal is to increase the RelaDiff value to reduce source bias.

Note that our work aims to align LLMs so that the rewritten texts are less likely to introduce source bias when ranked by PLM-based retrievers. As a result, the evaluation of source bias depends on the specific PLM-based retrievers employed. To verify the generalization capability of our proposed method, we evaluate its performance not only on SBERT—the PLM-based retriever used in our automatic data construction pipeline (**in-domain evaluation**)—but also on several other mainstream PLM-based retrieval models (**out-of-domain evaluation**). For the out-of-domain evaluation, we employ the officially released checkpoints of the following state-of-the-art PLM-based retrieval models trained on MS MARCO: (1) ANCE [38]; (2) TAS-B [13]; (3) Contriever [15]; (4) coCondenser [11]; (5) RetroMAE [36]; (6) DRAGON [20]. For more details of these models, please refer to the Cocktail benchmark [5].

5.1.3 *Baseline Methods.* As we are the first to address source bias from the LLM side[5] , there are no existing baselines directly comparable to our approach. Therefore, we adapt several common alignment methods to serve as baselines for comparison:

[5] Note that retriever-side methods are not directly comparable to LLM-side methods.

Table 2: Performance of different alignment methods evaluated by various PLM-based retrievers on three datasets. The original LLM without alignment (denoted as "Raw") corresponds to 0%, and their absolute value of RelaDiff are provided in parentheses. The best result for each dataset and PLM-based retriever is highlighted in bold.

Dataset	Method	Evaluator (PLM-based Retrieval Model)							
		SBERT	ANCE	TAS-B	Contriever	Cocondenser	RetroMAE	DRAGON	Avg.
SciFact	Raw	0.0%(−29.0)	0.0%(−53.3)	0.0%(−37.4)	0.0%(−38.0)	0.0%(−36.1)	0.0%(−20.2)	0.0%(−47.3)	0.0%(−37.3)
	+SFT	+0.9%	+6.6%	+33.6%	+6.7%	+8.3%	+20.2%	+15.4%	+13.1%
	+DPO	−9.7%	+41.0%	+18.3%	+4.6%	+28.1%	+26.5%	+25.9%	+19.2%
	+LLM-SBM (Ours)	**+2.9%**	**+67.9%**	**+59.9%**	**+53.0%**	**+45.7%**	**+80.1%**	**+55.8%**	**+52.2%**
NQ	Raw	0.0%(−17.4)	0.0%(−13.1)	0.0%(−27.7)	0.0%(−21.8)	0.0%(−12.1)	0.0%(−13.1)	0.0%(−47.7)	0.0%(−21.8)
	+SFT	+15.0%	+20.0%	+14.0%	+14.5%	+6.9%	+17.8%	+18.5%	+15.2%
	+DPO	+52.1%	+54.2%	+48.6%	+49.2%	+47.6%	+59.6%	+60.6%	+53.1%
	+LLM-SBM (Ours)	**+59.5%**	**+67.2%**	**+57.3%**	**+56.3%**	**+52.1%**	**+65.7%**	**+70.5%**	**+61.2%**
TREC-COVID	Raw	0.0%(−95.4)	0.0%(−68.8)	0.0%(−137.7)	0.0%(−98.2)	0.0%(−89.9)	0.0%(−78.5)	0.0%(−74.0)	0.0%(−91.8)
	+SFT	+14.2%	+13.2%	+55.0%	+40.6%	−20.4%	+30.2%	+32.1%	+23.6%
	+DPO	+30.9%	+36.4%	+43.3%	+24.9%	+5.1%	**+63.9%**	+45.4%	+35.7%
	+LLM-SBM (Ours)	**+79.0%**	**+36.9%**	**+56.2%**	**+77.5%**	**+33.2%**	+59.9%	**+71.4%**	**+59.2%**

(1) Supervised Fine-Tuning (SFT) [22]: We perform supervised fine-tuning using the chosen responses from our constructed paired alignment dataset, paired with the corresponding instructions, to create the SFT dataset.

(2) Direct Preference Optimization (DPO) [23]: We apply the standard DPO loss function (i.e., Eq. (1)) to our constructed alignment dataset for fine-tuning. This serves as an ablation study for our proposed LLM-SBM alignment framework, allowing us to assess the effectiveness of incorporating fine-grained preference differences.

5.1.4 Implementation Details. We use the LLaMA-Factory [43] framework[6] to implement both the baseline methods and our proposed LLM-SBM framework. To ensure fairness in evaluation, most of the training hyper-parameters are kept at their default settings. Due to computational resource limitations, we employ LoRA [14] for all experiments involving fine-tuning of LLMs and set the low rank to 8. Each experiment is trained for a total of 3 epochs, with a batch size of 4 and gradient accumulation steps set to 8. If not specified, we use Llama-3-8B-Instruct for experiments and set the α in LLM-SBM to 2.

5.2 Experimental Results

5.2.1 Evaluation of Mitigating Source Bias. We first test the source bias of content generated by Llama3 with different alignment methods on three datasets. The source bias results, assessed using various PLM-based retrievers, are reported in Table 2. Based on the results, we can draw the following observations and conclusions:

(1) The initial models without alignment (denoted as "Raw") exhibit significant source bias towards LLM-generated content across all datasets and most retrievers. Notably, on the TREC-COVID dataset, the average RelaDiff across all neural models exceeds −70%. These findings confirm the widespread presence of source bias in different PLM-based retrieval models [8], highlighting the urgent need to address this issue.

(2) Compared to the unaligned models, all models fine-tuned using our constructed alignment data demonstrate a significant reduction in source bias across the three datasets. Importantly, this improvement is observed not only on SBERT—the PLM-based retriever used in our data construction pipeline (in-domain evaluation)—but also on other PLM-based retrievers (out-of-domain evaluation). This outcome validates the effectiveness of our alignment data in providing valuable guidance to LLMs.

Table 3: Human evaluation for the quality of LLM-generated documents before and after alignment. Human annotators are asked to select which document exhibits higher quality based on language fluency and semantic completeness.

Dataset	Choice				
	Raw	+SFT	+ DPO	+ Ours	Equal
SciFact	2.0%	2.0%	6.0%	4.0%	86.0%
NQ	2.0%	0.0%	2.0%	4.0%	92.0%
TREC-COVID	0.0%	0.0%	4.0%	6.0%	90.0%
Avg.	1.3%	0.7%	4.0%	4.7%	89.3%

(3) The DPO method consistently outperforms SFT, which can be attributed to its explicit alignment of the model's outputs with preference pairs rather than only the positive responses (i.e., preferred responses), thereby achieving better generalization ability. Our proposed LLM-SBM takes this a step further by assigning different weights to preference samples based on their fine-grained preference differences, enabling the model to focus more on the "more reliable" preference pairs during alignment. As a result, our method shows substantial improvements across almost all cases, effectively eliminating source bias.

5.2.2 Evaluation of Maintaining Generation Quality. To further validate that the alignment process does not compromise the quality of LLM-generated responses in rewriting tasks, we randomly sample 50 documents for each dataset to conduct a human evaluation. Human annotators, comprising the authors and their highly educated colleagues, are tasked with evaluating the quality of the documents based on language fluency and semantic completeness. The experimental results in Table 3 indicate that, after alignment, the quality of LLM-generated documents did not decrease and even demonstrated

[6] https://github.com/hiyouga/LLaMA-Factory

Table 4: General capabilities evaluation across different domains on the MMLU dataset.

Domain	STEM	Social Sciences	Humanities	Other	Avg.
Raw	54.73	75.53	61.27	71.19	65.28
+SFT	54.17	74.58	60.32	70.94	64.58
+DPO	54.37	75.56	61.30	71.53	65.30
+Ours	54.67	75.50	61.25	71.69	65.37

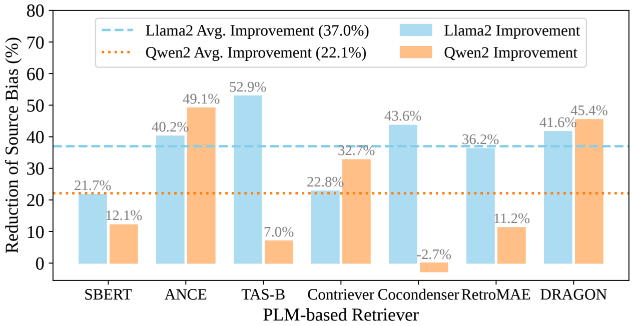


Figure 3: Reduction of source bias in Llama2 and Qwen2 after applying our proposed LLM-SBM alignment framework.

slight improvements, possibly due to the enhancing instruction-following capabilities during the alignment. This demonstrates that our alignment method effectively mitigates source bias without sacrificing the inherent quality of the generated content.

5.2.3 Evaluation of Maintaining General Capabilities. Moreover, to assess whether fine-tuning the LLM with our constructed source bias alignment dataset affects its general capabilities, we conduct experiments with various alignment methods on the MMLU benchmark [12], which is widely used for LLM general capabilities evaluation [9, 43]. The results across different domains are reported in Table 4. As observed, all alignment methods, including SFT, DPO, and our proposed LLM-SBM, effectively preserve the general capabilities of the original LLM. The variations in accuracy are minimal, indicating that the alignment process does not lead to catastrophic forgetting of previously acquired knowledge. This result demonstrates that our source bias alignment task can significantly reduce source bias without compromising the LLM's general capabilities.

5.3 Further Analysis

Due to the high training and inference costs of LLMs, we further conduct more in-depth analyses on SciFact dataset.

5.3.1 Applying Alignment Data to Other LLMs. One of the key contributions is the constructed alignment data for counteracting source bias. In this part, we investigate whether the alignment data constructed using Llama3 can also be effectively applied to other LLMs to eliminate source bias. Specifically, we conduct experiments on two other commonly used open-source LLMs: Llama2[7] and Qwen2[8], utilizing our constructed alignment datasets for alignment. As shown in Figure 3, after applying our alignment method, the source bias in both LLMs is substantially reduced. These experimental results further underscore the effectiveness of our constructed alignment data, demonstrating its applicability in aligning other LLMs to eliminate source bias. Combined with the previous experiments verifying that LLMs aligned through LLM-SBM retain general capabilities—and even show slight improvements—while greatly reducing source bias, we suggest that LLM-SBM can be widely adopted as a general post-training framework before releasing the checkpoint or providing the API service. By doing so, the LLM service providers can ensure that their LLMs or API services do not contribute to source bias in downstream applications.

Table 5: Analysis of generalization of our proposed method to other commonly used rephrasing instructions.

Instruction	Method	Avg. RelaDiff
Paraphrase the provided text while maintaining its meaning.	Raw	0.0%(−21.1)
	+Ours	+46.1%
Rephrase the given text using alternative expressions.	Raw	0.0%(−34.8)
	+Ours	+39.4%
Summarize the following passage in a concise manner.	Raw	0.0%(−30.5)
	+Ours	+36.8%
Reword the passage below to make it more succinct.	Raw	0.0%(−47.9)
	+Ours	+21.2%

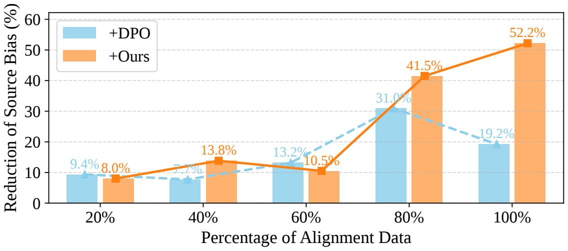
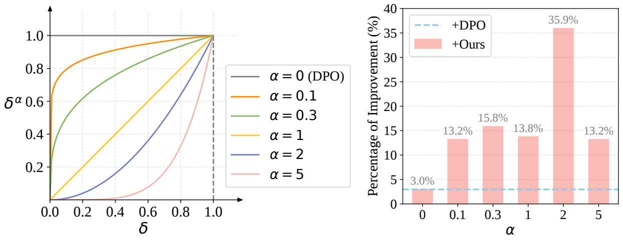


Figure 4: Reduction of source bias after aligning with different amounts of data.

5.3.2 Generalization to Unseen Rephrasing Instructions. Considering that users in real-world scenarios may employ a variety of instructions when using LLMs for text rewriting, we aim to verify whether LLMs aligned by our LLM-SBM framework can effectively eliminate source bias across different rephrasing instructions. To this end, we select unseen instructions during alignment [9] and report the average results across seven PLM-based retrievers on Table 5. We observe that these common prompts can readily trigger source bias in LLM-generated content, further emphasizing the necessity of addressing this issue. Notably, after applying our LLM-SBM alignment framework, the source bias in the texts generated under these different prompts is significantly reduced. This demonstrates that our solution from the LLM side can fundamentally mitigate source bias, effectively resisting malicious users who might exploit source bias to attack neural retrieval models in today's search engines.

5.3.3 Analysis of the Amount of Alignment Data. In this set of experiments, we aim to explore the impact of the amount of alignment data on the performance of our proposed method. We trained our

[7] https://huggingface.co/meta-llama/Llama-2-7b-chat-hf
[8] https://huggingface.co/Qwen/Qwen2-7B-Instruct
[9] https://github.com/kevinamiri/Instructgpt-prompts#rephrase-a-passage



(a)

(b)

Figure 5: Illustration and analysis of the hyper-parameter α in weight. (a): The weight curves with different α values. (b): Reduction of source bias w.r.t different α values.

model using varying proportions of the alignment data and evaluated its performance after each training iteration. The average results across seven PLM-based retrievers are presented in Figure 4. As expected, the performance of both DPO and our LLM-SBM improves with increasing amounts of training data, indicating that alignment methods benefit from a larger alignment dataset, leading to more effective mitigation of source bias. Furthermore, as the data volume increases, the debiasing performance of DPO declines at full data, indicating that larger datasets may introduce more noise. However, our approach effectively mitigates this issue and outperforms DPO, as analyzed in Section 4.3.2. These results validate the necessity of the data expansion strategy employed in our data construction pipeline for automatically constructing alignment data at scale, demonstrating that increasing the amount of alignment data enhances overall alignment performance and underscores the efficacy of our LLM-SBM optimization method.

5.3.4 Analysis of the Hyper-Parameter α. As described in Section 4.3.1, our LLM-SBM introduces a hyper-parameter α in Eq (2) to control the effect of the preference difference on the alignment loss. Specifically, Figure 5(a) illustrates the weighting curves with respect to the preference difference for varying values of α in the range $\{0, 0.1, 0.3, 1, 2, 5\}$. In particular, when $\alpha = 0$, LLM-SBM reduces to the standard DPO formulation, assigning equal loss weights to all preference pairs regardless of their preference differences.

Figure 5(b) presents the experimental results with different values of α. From these results, we observe that LLM-SBM generally outperforms DPO, especially when $\alpha = 2$. We attribute this improvement to the weight mechanism of LLM-SBM: samples with smaller preference differences receive lower weights when α is greater than zero. Such samples indicate that, according to the specific PLM-based retriever (i.e., SBERT in our experiments), the chosen document is not significantly better than the rejected document. Consequently, when evaluated with other PLM-based retrievers (i.e., out-of-domain evaluation setting), the preference might be reversed, making these samples more likely to be noisy. By setting $\alpha = 2$, the weight curve becomes concave downward, further diminishing the influence of samples with small preference differences. This allows the model to focus more on samples with larger and more reliable preference differences, thereby enhancing the overall alignment performance and reducing the source bias.

Table 6: Statistics of data count and performance of bias reduction for different values of N.

Method	Raw	N = 2	N = 4	N = 6
Data Count	-	$77 \times 2 = 154$	$671 \times 4 = 2,683$	$1,805 \times 6 = 10,830$
Bias Reduction	0.0% (-37.3)	-9.5%	+33.2%	+52.2%

5.3.5 Analysis of the Hyper-Parameter N. Note that increasing N can directly amplify the amount of alignment data by enabling more pairwise combinations between chosen and rejected responses, thus facilitating a more comprehensive alignment process. We further conduct experiments to analyze the effect of varying the number of N on both the quantity of alignment data constructed and the resulting bias reduction. The results are summarized in the Table 6.

From these results, we observe that increasing N significantly boosts the amount of alignment data, which in turn leads to more effective bias reduction. Specifically, as N increases from 2 to 6, the bias reduction improves from -9.5% to +52.2%, highlighting the importance of N in enhancing the alignment process. These findings suggest that a larger N provides a richer and more diverse set of alignment examples, which strengthens the model's ability to counteract source bias. While it is possible to create additional rephrasing instructions to generate more alignment data, we find that $N = 6$ is sufficient to achieve significant bias reduction. Therefore, we did not explore higher values of N, as the current results already demonstrate significant effectiveness.

6 Conclusion

This paper introduces the LLM-SBM framework to align LLMs for mitigating source bias from the data generation side. The LLM-SBM framework encompasses an automatic preference data construction pipeline that generates high-quality alignment samples by leveraging multiple rewrites from LLMs and assigned preferences from the retriever. Furthermore, LLM-SBM incorporates fine-grained preference differences as weights in the loss function, enhancing the efficiency of policy training. Extensive experiments across three diverse IR datasets and seven different PLM-based retrievers demonstrate that LLMs aligned by LLM-SBM effectively reduce source bias while maintaining their general capabilities.

In future work, we will further explore strategies to address source bias simultaneously from both the LLM side and the retriever side, thereby better jointly promoting the sustainable development of the entire information content ecosystem.

Acknowledgments

This work was funded by the National Key R&D Program of China (2023YFA1008704), the National Natural Science Foundation of China (No. 62472426, 62276248), Engineering Research Center of Next-Generation Intelligent Search and Recommendation, Ministry of Education, Major Innovation & Planning Interdisciplinary Platform for the "Double-First Class" Initiative, fund for building world-class universities (disciplines) of Renmin University of China and the Youth Innovation Promotion Association CAS under Grants No.2023111. This work was supported by the Fundamental Research Funds for the Central Universities, and the Research Funds of Renmin University of China (RUC24QSDL013).

References

[1] Yihan Cao, Siyu Li, Yixin Liu, Zhiling Yan, Yutong Dai, Philip S Yu, and Lichao Sun. 2023. A comprehensive survey of ai-generated content (aigc): A history of generative ai from gan to chatgpt. *arXiv preprint arXiv:2303.04226* (2023).

[2] Xiaoyang Chen, Ben He, Hongyu Lin, Xianpei Han, Tianshu Wang, Boxi Cao, Le Sun, and Yingfei Sun. 2024. Spiral of Silences: How is Large Language Model Killing Information Retrieval?–A Case Study on Open Domain Question Answering. *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics* (2024).

[3] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems* 30 (2017).

[4] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research* 25, 70 (2024), 1–53.

[5] Sunhao Dai, Weihao Liu, Yuqi Zhou, Liang Pang, Rongju Ruan, Gang Wang, Zhenhua Dong, Jun Xu, and Ji-Rong Wen. 2024. Cocktail: A Comprehensive Information Retrieval Benchmark with LLM-Generated Documents Integration. *Findings of the Association for Computational Linguistics: ACL 2024* (2024).

[6] Sunhao Dai, Chen Xu, Shicheng Xu, Liang Pang, Zhenhua Dong, and Jun Xu. 2024. Bias and Unfairness in Information Retrieval Systems: New Challenges in the LLM Era. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 6437–6447.

[7] Sunhao Dai, Chen Xu, Shicheng Xu, Liang Pang, Zhenhua Dong, and Jun Xu. 2025. Unifying Bias and Unfairness in Information Retrieval: New Challenges in the LLM Era. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*. 998–1001.

[8] Sunhao Dai, Yuqi Zhou, Liang Pang, Weihao Liu, Xiaolin Hu, Yong Liu, Xiao Zhang, Gang Wang, and Jun Xu. 2024. Neural Retrievers are Biased Towards LLM-Generated Content. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 526–537.

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

[10] Haowen Gao, Liang Pang, Shicheng Xu, Leigang Qu, Tat-Seng Chua, Huawei Shen, and Xueqi Cheng. 2025. Generative Ghost: Investigating Ranking Bias Hidden in AI-Generated Videos. *arXiv preprint arXiv:2502.07327* (2025).

[11] Luyu Gao and Jamie Callan. 2022. Unsupervised Corpus Aware Language Model Pre-training for Dense Passage Retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2843–2853.

[12] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300* (2020).

[13] Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. 2021. Efficiently teaching an effective dense retriever with balanced topic aware sampling. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 113–122.

[14] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.

[15] Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised Dense Information Retrieval with Contrastive Learning. *Transactions on Machine Learning Research* (2022).

[16] Jiaming Ji, Tianyi Qiu, Boyuan Chen, Borong Zhang, Hantao Lou, Kaile Wang, Yawen Duan, Zhonghao He, Jiayi Zhou, Zhaowei Zhang, et al. 2023. Ai alignment: A comprehensive survey. *arXiv preprint arXiv:2310.19852* (2023).

[17] Zachary Kenton, Tom Everitt, Laura Weidinger, Iason Gabriel, Vladimir Mikulik, and Geoffrey Irving. 2021. Alignment of language agents. *arXiv preprint arXiv:2103.14659* (2021).

[18] 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.

[19] Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: A Python toolkit for reproducible information retrieval research with sparse and dense representations. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2356–2362.

[20] Sheng-Chieh Lin, Akari Asai, Minghan Li, Barlas Oguz, Jimmy Lin, Yashar Mehdad, Wen-tau Yih, and Xilun Chen. 2023. How to Train Your DRAGON: Diverse Augmentation Towards Generalizable Dense Retrieval. *arXiv preprint arXiv:2302.07452* (2023).

[21] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. *choice* 2640 (2016), 660.

[22] 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.

[23] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems* 36 (2024).

[24] 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)*. 3982–3992.

[25] Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. 2024. AI models collapse when trained on recursively generated data. *Nature* 631, 8022 (2024), 755–759.

[26] 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.

[27] Hexiang Tan, Fei Sun, Wanli Yang, Yuanzhuo Wang, Qi Cao, and Xueqi Cheng. 2024. Blinded by Generated Contexts: How Language Models Merge Generated and Retrieved Contexts for Open-Domain QA? *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics* (2024).

[28] Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A Heterogeneous Benchmark for Zero-shot Evaluation of Information Retrieval Models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

[29] Ellen Voorhees, Tasmeer Alam, Steven Bedrick, Dina Demner-Fushman, William R Hersh, Kyle Lo, Kirk Roberts, Ian Soboroff, and Lucy Lu Wang. 2021. TREC-COVID: constructing a pandemic information retrieval test collection. In *ACM SIGIR Forum*, Vol. 54. 1–12.

[30] David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman Cohan, and Hannaneh Hajishirzi. 2020. Fact or Fiction: Verifying Scientific Claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 7534–7550.

[31] Haoyu Wang, Sunhao Dai, Haiyuan Zhao, Liang Pang, Xiao Zhang, Gang Wang, Zhenhua Dong, Jun Xu, and Ji-Rong Wen. 2025. Perplexity Trap: PLM-Based Retrievers Overrate Low Perplexity Documents. In *13th International Conference on Learning Representations, ICLR 2025*.

[32] Jiahao Wang, Bolin Zhang, Qianlong Du, Jiajun Zhang, and Dianhui Chu. 2024. A Survey on Data Selection for LLM Instruction Tuning. *arXiv preprint arXiv:2402.05123* (2024).

[33] 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. *arXiv preprint arXiv:2307.12966* (2023).

[34] Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652* (2021).

[35] Jiayang Wu, Wensheng Gan, Zefeng Chen, Shicheng Wan, and Hong Lin. 2023. Ai-generated content (aigc): A survey. *arXiv preprint arXiv:2304.06632* (2023).

[36] Shitao Xiao, Zheng Liu, Yingxia Shao, and Zhao Cao. 2022. RetroMAE: Pre-Training Retrieval-oriented Language Models Via Masked Auto-Encoder. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. 538–548.

[37] Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. 2023. C-pack: Packaged resources to advance general chinese embedding. *arXiv preprint arXiv:2309.07597* (2023).

[38] 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).

[39] Shicheng Xu, Danyang Hou, Liang Pang, Jingcheng Deng, Jun Xu, Huawei Shen, and Xueqi Cheng. 2024. Invisible Relevance Bias: Text-Image Retrieval Models Prefer AI-Generated Images. *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (2024).

[40] 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. *arXiv preprint arXiv:2304.05302* (2023).

[41] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223* (2023).

[42] Yao Zhao, Rishabh Joshi, Tianqi Liu, Misha Khalman, Mohammad Saleh, and Peter J Liu. 2023. Slic-hf: Sequence likelihood calibration with human feedback. *arXiv preprint arXiv:2305.10425* (2023).

[43] Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. LlamaFactory: Unified Efficient Fine-Tuning of 100+ Language Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*.

Association for Computational Linguistics.

[44] Yuqi Zhou, Sunhao Dai, Liang Pang, Gang Wang, Zhenhua Dong, Jun Xu, and Ji-Rong Wen. 2025. Exploring the Escalation of Source Bias in User, Data, and Recommender System Feedback Loop. *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval* (2025).

[45] 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).