Uplift-RAG: Uplift-Driven Knowledge Preference Alignment for Retrieval-Augmented Generation

Changle Qu[1], Sunhao Dai[1], Hengyi Cai[2], Yiyang Cheng[1], Jun Xu[1]*, Shuaiqiang Wang[2], Dawei Yin[2]
[1]Gaoling School of Artificial Intelligence, Renmin University of China; [2]Baidu Inc.
{changlequ,sunhaodai,chengyiyang041023,junxu}@ruc.edu.cn
caihengyi@ict.ac.cn, wangshuaiqiang@baidu.com, yindawei@acm.org

Abstract

Retrieval-augmented generation (RAG) has proven effective in enhancing the knowledge coverage of large language models (LLMs) and mitigating hallucinations by incorporating external retrieved documents. However, documents deemed relevant by the retriever are not necessarily helpful for answer generation, and including misleading information can even degrade performance. Existing efforts to estimate document utility often rely on downstream generation performance, which conflates the influence of external documents with the intrinsic knowledge of the LLM, thereby obscuring the actual contribution of the retrieved content. To address this, this paper proposes Uplift-RAG, a uplift-driven knowledge preference alignment framework for RAG. Specifically, we first propose an uplift-based definition of document utility that quantifies each document's marginal benefit over the LLM's internal knowledge. We then optimize the reranker with three alignment objectives to identify and prioritize documents based on their uplift. This enables dynamic selection of documents that address the LLM's knowledge gaps, going beyond fixed top-k selection, while reducing reference redundancy and the computational overhead of the LLM's input. Extensive experiments demonstrate the effectiveness of Uplift-RAG.

1 Introduction

Retrieval-augmented generation (RAG), which enhances large language models (LLMs) with access to retrieved documents relevant to the question (Guu et al., 2020; Gao et al., 2023; Li et al., 2025), has substantially improved the performance of LLMs in handling knowledge-intensive tasks while effectively mitigating hallucination issues. A typical RAG system comprises several components that work collaboratively to support effective response generation (Fan et al., 2024), where the retrieval subsystem constitutes the architectural cornerstone for providing reference documents.

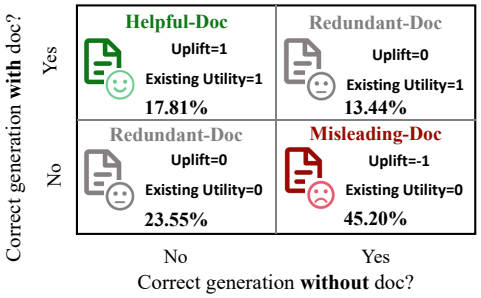


Figure 1: Impact of top-10 retrieved docs from *e5-base-v2* on correctness for *Llama-3-8B-Instruct* on NQ, showing four uplift categories ($\text{Correct}_{\text{with}} - \text{Correct}_{\text{without}}$) and corresponding existing utility ($\text{Correct}_{\text{with}}$).

However, the retrieval component is typically optimized for query-document relevance metrics, and usually fails to address the essential requirement of the generator for utility-driven document selection (Zhang et al., 2024a), where the retrieved content must not only be topically related but also actionable for refining the response accuracy of the LLM. As shown in Figure 1, our analysis reveals that among documents ranked by standard relevance-based methods, only 17.81% actually assist the generator in correcting its potential errors, while 45.20% paradoxically degrade performance by introducing conflicting evidence that overrides the correct internal knowledge of the LLM. The remaining 37% exhibit neutral effects. These results highlight a fundamental misalignment between the retrieval objective and the need of LLM for knowledge supplementation: Retrieval models maximize document recall under lexical/semantic similarity metrics, whereas LLMs require documents that fill specific knowledge voids in their parametric memory. In some cases, seemingly relevant documents may even conflict with the internal knowledge of the LLM, damaging its performance. Bridging this knowledge preference gap is critical to improving

* Corresponding author.

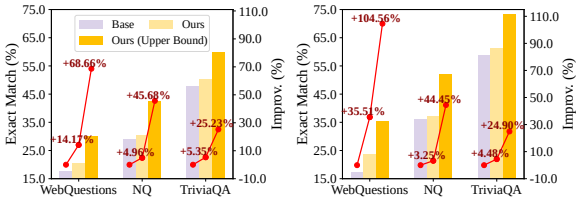
Findings of the Association for Computational Linguistics: EMNLP 2025, pages 9632–9644
November 4-9, 2025

the effectiveness of RAG system.

The community has made preliminary attempts to address this preference misalignment, but three fundamental challenges remain inadequately resolved. **First**, defining and quantifying document utility remains an open problem. Existing approaches often rely on the downstream generator's performance as a proxy for utility (Ma et al., 2023; Shi et al., 2024; Ke et al., 2024). This method, however, suffers from a critical credit assignment problem: it fails to disentangle the retrieved document's actual contribution from the LLM's inherent parametric knowledge or its reasoning capabilities. **Second**, aligning retrieval mechanisms with such a nuanced utility definition poses a significant hurdle. Current retrievers are predominantly optimized for topical relevance. Integrating a utility dimension requires a paradigm shift where the system must not only identify relevant documents but also assess their potential to complement the generator's knowledge and rectify its potential inaccuracies. **Third**, the inherently adaptive nature of document utility complicates its modeling. The usefulness of a particular piece of information is not static; it is query-specific and model-dependent. Therefore, a one-size-fits-all approach, such as retrieving a fixed top-k set of documents, is suboptimal, leading to the inclusion of noisy or redundant information, and increasing LLM's computational overhead without guaranteeing performance gains.

Toward this end, this paper proposes Uplift-RAG. We first propose a novel uplift-based approach to accurately define and quantify document utility. Unlike prior methods relying solely on relevance or entangled utility metrics as supervision signals, our approach leverages uplift, a measure capturing the genuine incremental benefit each document provides to the generator. Specifically, we prompt the LLM to independently generate responses using each retrieved document, evaluating the quality of these responses against ground truth labels and comparing them to the baseline performance without external documents. The resulting performance differential constitutes the uplift of each document.

To mitigate the discrepancy between the retrieval objective and the need of the LLM for knowledge supplementation, we then introduce a lightweight reranker that utilizes the computed uplift scores to effectively integrate utility considerations into the retrieval process. Concretely, we incorporate a point-wise loss to directly identify documents beneficial to the generator. To ensure effective prioritization, we further integrate pair-wise and list-wise losses, enabling the reranker to rank documents by their uplift accurately. By jointly optimizing these objectives, the reranker aligns closely with the specific knowledge needs of the generator. In addition, our reranker selectively identifies and includes only those documents that genuinely address the LLM's immediate knowledge gaps, thereby avoiding the retrieval of redundant or misleading content, and reducing unnecessary computational overhead.



(a) Llama-3.2-1B-Instruct (b) Llama-3-8B-Instruct

Figure 2: Empirical performance improvements and potential upper bound analysis of Uplift-RAG.

We conduct extensive experiments on three QA datasets, demonstrating the effectiveness and robustness of our method. Notably, our upper bound analysis of Uplift-RAG demonstrates its maximum achievable improvement of 25%-105% (as shown in Figure 2), along with a 50%–60% reduction in token costs, highlighting its promising capability. In summary, our main contributions are as follows:

- We propose employing uplift modeling to define and quantify the utility of retrieved documents, as it can disentangle the contribution of retrieved documents from the inherent capabilities of LLMs.
- We propose Uplift-RAG, a novel uplift-driven preference alignment framework for RAG, which not only enables ranking based on uplift but also identifies and includes only helpful documents.
- Extensive experiments on three datasets with various LLMs demonstrate the effectiveness of Uplift-RAG, with an in-depth upper bound analysis further highlighting its promising potential.

2 Our Approach: Uplift-RAG

In this section, we first introduce the task formulation. Then we describe the details of Uplift-RAG.

2.1 Task Formulation

Retriever. Formally, given a user query $q \in \mathcal{Q}$, the retriever aims to filter out the top-K most relevant documents $\mathcal{D}_{Ret} = \{d_1, d_2, \ldots, d_K\}$ from a corpus $\mathcal{D}$ (*e.g.,* a Wikipedia dump). In general, the retriever typically adopts a bi-encoder architecture, such as DPR (Karpukhin et al., 2020) and

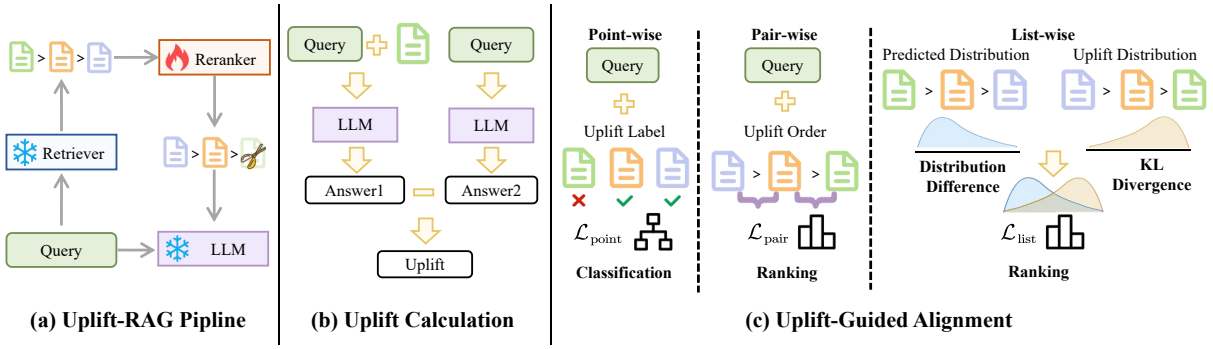


Figure 3: The illustration of our proposed Uplift-RAG. (a) shows the overall pipeline of Uplift-RAG. (b) illustrates the process of uplift calculation. (c) presents the three designed alignment objectives to optimize the reranker.

e5-base-v2 (Wang et al., 2022), where the query and documents are encoded independently.

Reranker. The reranker aims to further refine the results returned by the retriever by reordering or filtering the candidate documents, ultimately producing a more accurate top-k document set, denoted as $\mathcal{D}_{Rer} = \{d_1, d_2, \ldots, d_k\}$. Unlike retrievers, which typically rely on bi-encoder architectures, rerankers generally adopt cross-encoder models that allow for deeper interaction between the query and each candidate document during scoring.

Generator. Finally, the generator $\mathcal{M}$ takes the query along with the top-k reranked documents as input to generate a final response $\hat{y}$:

$$\hat{y} = \mathcal{M}(q, D_{Rer}).$$

Objective. As discussed in §1, the documents deemed relevant by the retriever may not necessarily be beneficial for the generator in answering the query. Therefore, our goal is to bridge this knowledge preference gap by empowering the reranker to (1) include only those documents that genuinely address the immediate knowledge gaps of the LLM and (2) effectively rank these documents based on their true contribution to generation quality.

2.2 Overview of Uplift-RAG

To achieve the above goal, we propose Uplift-RAG, a novel uplift-driven knowledge preference alignment framework for RAG. As illustrated in Figure 3, Uplift-RAG consists of two key phases: uplift calculation and uplift-guided alignment. In the uplift calculation phase, we define and quantify the utility of each document by computing its uplift, which reflects how much it improves the answer quality of the LLM. In the uplift-guided alignment phase, we leverage the calculated uplift to guide the optimization of the reranker. To enable the reranker to identify beneficial documents and rank them by their uplift, we design three alignment objectives that jointly endow it with both classification and ranking capabilities. By jointly optimizing these objectives, the reranker better captures the generator's knowledge needs, ultimately improving the overall effectiveness of the RAG system.

2.3 Uplift Calculation

As shown in Figure 3 (b), the first phase of Uplift-RAG focuses on calculating the uplift of each retrieved document, quantifying its contribution to improving the response quality of LLM. To establish a baseline, we begin by prompting the LLM $\mathcal{M}$ to answer the query q without any supporting documents, yielding the response $\mathcal{M}(q)$. We then assess the quality of this response using a downstream evaluation function $\mathcal{F}$:

$$S_q = \mathcal{F}(\mathcal{M}(q), y), \tag{1}$$

where S_q denotes the score assigned to the response to query q without document, and y represents the expected downstream output (i.e., the ground-truth). The evaluation function $\mathcal{F}$ can be instantiated with any suitable metric, such as EM or F1.

Next, for each document $d \in \mathcal{D}_{Ret}$ retrieved for the query, we prompt the LLM to generate a response $\mathcal{M}(q, d)$ conditioned on both the query and the document. Similarly, the quality of each response is assessed using the function $\mathcal{F}$:

$$S_{q,d} = \mathcal{F}(\mathcal{M}(q, d), y), \tag{2}$$

where $S_{q,d}$ denotes the score of the response to the query q when using document d.

We define the uplift of the document d to the query as the difference in response quality between using the document and not using any documents:

$$\text{Uplift}(q, d) = S_{q,d} - S_q. \tag{3}$$

Based on this definition, a positive uplift indicates that the document contributes positively to the generation quality, while a negative uplift suggests it introduces noise or distracts the LLM from the correct answer. Additionally, a document with zero uplift is also considered unhelpful, as it offers no measurable benefit and merely increases unnecessary computational overhead of LLM's input.

Discussion. The uplift approach differs from existing methods that rely solely on response quality after document augmentation in two key aspects. First, when the LLM performs correctly regardless of document presence, such harmlessly redundant documents offer no benefit while increasing token usage, yet traditional methods may still incorrectly classify them as helpful. Second, documents that turn correct answers into incorrect ones should be penalized, yet conventional approaches treat them equivalently to unhelpfully redundant documents. In contrast, the uplift-based formulation enables us to isolate and quantify the individual contribution of each document, providing a more reliable supervision signal for aligning the reranker with the actual knowledge preferences of the generator.

2.4 Uplift-Guided Alignment

To bridge the knowledge preference gap between the retriever and the generator, the calculated uplift is employed as a supervision signal to optimize the reranker, encouraging it to identify and prioritize documents that truly enhance the response quality of the generator. As shown in Figure 3 (c), we introduce an uplift-guided alignment framework that integrates three complementary alignment strategies to make full use of the uplift signal.

Point-wise Alignment. Given that not all documents retrieved based on relevance are truly helpful to the generator, as some may even be misleading and degrade the quality of the generated response. To address this, the reranker must be capable of distinguishing genuinely helpful documents. We therefore formulate document selection as a binary classification task, labeling each document as positive if it has a positive uplift score and negative otherwise. Since existing lightweight rerankers only output a single relevance score without native binary classification support, we adapt them by modifying the final linear layer to produce two scores, applying softmax to compute the probability of a document being helpful.

The reranker is trained using a point-wise binary cross-entropy loss. Formally, given a document d_j with an associated uplift label $u_{ij} \in \{0, 1\}$ with respect to the query q_i, the loss is defined as:

$$\mathcal{L}_{point} = -\sum_{i=1}^{N}\sum_{j=1}^{|\mathcal{D}_{Ret}|}(u_{ij}\cdot\log(p_{ij}) + (1-u_{ij})\cdot\log(1-p_{ij})),$$

where N is the number of queries, p_{ij} represents the predicted probability that the reranker considers the document d_j to be helpful for the query q_i.

Pair-wise Alignment. Beyond binary classification of document usefulness, the reranker must also be able to rank documents based on their relative uplift to the generator. To achieve this, we introduce a pair-wise alignment objective that prioritizes documents with higher uplift values. Specifically, for each query q_i, we construct document pairs (d_j^+, d_m^-), where d_j^+ has a higher uplift than d_m^-. The reranker is then trained to respect this preference order through the following pair-wise loss:

$$\mathcal{L}_{pair} = -\sum_{i=1}^{N}\sum_{j=1}^{|\mathcal{D}_{Ret}|}\sum_{m=j+1}^{|\mathcal{D}_{Ret}|}(\log(\sigma(p_{ij}-p_{im}))).$$

List-wise Alignment. While point-wise and pair-wise objectives equip the reranker with the ability to identify helpful documents and preserve their relative order, they rely only on coarse-grained signals such as binary uplift labels or uplift-based ranking orders, and do not fully leverage the actual uplift values. Moreover, since the generator ultimately consumes a list of documents, optimizing only individual or pairwise relations may not reflect overall ranking quality. To better align the reranker with the knowledge preferences of the generator at the list level, we introduce a list-wise alignment objective. Specifically, we first compute the ideal likelihood distribution based on the uplift value:

$$P_U(d_j|q_i) = \frac{e^{\text{Uplift}(q_i,d_j)}}{\sum_{d_l\in\mathcal{D}_{Ret}} e^{\text{Uplift}(q_i,d_l)}}. \quad (4)$$

Similarly, we obtain the predicted distribution:

$$P_R(d_j|q_i) = \frac{e^{p_{ij}}}{\sum_{d_l\in\mathcal{D}_{Ret}} e^{p_{il}}}. \quad (5)$$

We then train the reranker to minimize the KL divergence between the two likelihood distributions:

$$\mathcal{L}_{list} = \sum_{i=1}^{N}\sum_{j=1}^{|\mathcal{D}_{Ret}|} P_U(d_j|q_i)\cdot\log(\frac{P_U(d_j|q_i)}{P_R(d_j|q_i)}).$$

This objective encourages the reranker to align the ideal distribution implied by the uplift values, thereby matching the generator's knowledge preferences in a fine-grained, list-aware manner.

Joint Training. To combine the complementary strengths of the three alignment objectives, we optimize the reranker through weighted loss fusion:

$$\mathcal{L} = \lambda_1 \cdot \mathcal{L}_{point} + \lambda_2 \cdot \mathcal{L}_{pair} + \mathcal{L}_{list}$$

where λ_1 and λ_2 are coefficients to balance the contribution of each component.

2.5 Inference

During the inference phase, the trained reranker performs a two-step process: first identifying potentially helpful documents, then ranking them according to their predicted usefulness probabilities. Specifically, for each retrieved document $d_j \in \mathcal{D}_{Ret}$ corresponding to the query q_i, the reranker outputs a probability p_{ij} representing the estimated likelihood of d_j being helpful. The system retains only those documents with $p_{ij} > 0.5$ (indicating they are more likely to be helpful than not) and sorts them in descending order based on their p_{ij} values. This strategy guarantees the selection and prioritization of documents with positive estimated utility while eliminating the requirement to specify a fixed top-k documents. Consequently, it effectively filters out redundant or misleading content and minimizes unnecessary token consumption.

3 Experiments

3.1 Experimental Setups

Datasets. To verify the effectiveness of Uplift-RAG, we conduct experiments on three datasets: WebQuestions (Berant et al., 2013), Natural Questions (NQ) (Kwiatkowski et al., 2019), and TriviaQA (Joshi et al., 2017). The details of these datasets are shown in Appendix A.1.

Evaluation Metrics. Following previous works (Ma et al., 2023; Shi et al., 2024; Jia et al., 2024), we evaluate performance using two widely adopted metrics: (1) Exact Match (EM), which measures whether the predicted answer and the ground truth are the same. (2) F1 score, which measures the number of overlapping words between them.

Baselines. We compare Uplift-RAG with the following baseline methods: (1) Naive Generation, (2) Standard RAG, (3) BGE-reranker (Xiao et al., 2024), (4) AAR (Yu et al., 2023), (5) REPLUG (Shi et al., 2024), (6) SKR (Wang et al., 2023), (7) Adaptive-RAG (Jeong et al., 2024), (8) DPA-RAG (Dong et al., 2025). For more details, please refer to Appendix A.2.

Implementation Details. We implement all baseline methods using the FlashRAG (Jin et al., 2025) library, except for DPA-RAG, which is directly implemented using the released codes by the authors. Following previous works (Jin et al., 2025; Jia et al., 2024), we employ *e5-base-v2* [1] (Wang et al., 2022) as the retriever and use *LLaMA-3-8B-Instruct* and *LLaMA-3.2-1B-Instruct* as generators. To ensure a fair comparison, we retrieve the top-5 documents for methods without a reranker, while for those with a reranker, we first retrieve the top-10 and then use the reranker to select the top-5 to be fed into the LLMs. We use the *roberta-base* as our backbone model and set the training epochs to 10 with a learning rate of $1e$–5. All experiments are conducted on NVIDIA RTX A6000 48G GPUs. Our code is available at `https://github.com/quchangle1/Uplift-RAG`.

3.2 Experimental Results

We present the experimental results in Table 1, from which we derive the following observations:

• Firstly, compared to Naive Generation, RAG-based methods generally perform better across most datasets, demonstrating the benefits of incorporating external documents. However, on the WebQuestions dataset, Naive Generation achieves better performance using *Llama-3-8B-Instruct*, suggesting that harmful documents may sometimes mislead the LLM rather than assist it.

• Secondly, compared to methods that either perform no alignment or align only the retriever, DPA-RAG achieves superior performance by optimizing the reranker based on preference alignment using both aligned and unaligned knowledge. This highlights the advantage of aligning the reranker with the preferences of the generator.

• Furthermore, compared to all baseline methods, Uplift-RAG achieves the best results, demonstrating its superior effectiveness. This improvement can be attributed to the use of uplift as a supervision signal, which accurately captures the contribution of each document to answer generation. By jointly optimizing three alignment objectives, Uplift-RAG equips the reranker with both classifi-

[1] https://huggingface.co/intfloat/e5-base-v2

Methods	Llama-3-8B-Instruct						Llama-3.2-1B-Instruct					
	WebQuestions		NQ		TriviaQA		WebQuestions		NQ		TriviaQA	
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
Naive Generation	20.62	38.43	22.43	31.57	55.29	61.74	11.61	24.78	9.05	15.37	22.27	27.77
Standard RAG	17.32	34.06	35.95	47.08	58.68	68.02	17.71	32.23	29.05	38.42	47.64	55.54
BGE-reranker	19.04	35.22	35.84	47.18	59.69	69.17	19.38	34.17	28.67	37.56	49.15	57.20
AAR	17.71	33.12	31.05	41.33	55.44	64.52	16.63	30.82	24.21	32.67	43.57	51.23
REPLUG	19.68	35.79	31.38	41.56	57.53	65.44	18.30	34.10	21.99	31.11	44.17	52.45
SKR	18.70	35.40	33.68	44.13	58.03	66.43	16.28	30.54	25.62	34.16	41.07	48.39
Adaptive-RAG	14.96	31.12	36.01	47.09	52.13	61.92	15.15	29.23	29.41	38.62	40.97	49.45
DPA-RAG	19.29	36.56	36.81	47.24	60.56	**69.42**	19.58	34.73	30.19	**39.64**	49.43	57.44
Uplift-RAG (Ours)	**23.47**	**39.48**	**37.12**	**48.14**	**61.31**	68.75	**20.22**	**35.38**	**30.49**	39.24	**50.19**	**57.88**
Uplift-RAG (Upper Bound)	35.43	54.52	51.93	64.66	73.29	81.01	29.87	48.89	42.32	54.90	59.66	68.95

Table 1: Performance comparison between Uplift-RAG and the baselines on three datasets with two LLMs. The best and second-best performance methods are highlighted in bold and underlined fonts, respectively.

Methods	NQ		TriviaQA	
	EM	F1	EM	F1
Llama-3-8B-Instruct				
Uplift-RAG	37.12	48.14	61.31	68.75
w/o point-wise alignment	24.43	33.46	59.88	66.98
w/o pair-wise alignment	34.93	45.46	60.66	68.01
w/o list-wise alignment	36.56	46.92	60.55	67.93
Llama-3.2-1B-Instruct				
Uplift-RAG	30.49	39.24	50.19	57.88
w/o point-wise alignment	27.64	35.48	26.06	31.26
w/o pair-wise alignment	30.19	39.01	48.23	55.77
w/o list-wise alignment	30.08	39.03	48.53	56.02

Table 2: Ablation study of the proposed Uplift-RAG.

Methods	NQ		TriviaQA	
	|N|	Token (Ratio)	|N|	Token (Ratio)
Llama-3-8B-Instruct				
Base	5.00	712.59 (100.0%)	5.00	717.13 (100.0%)
Uplift-RAG	1.63	232.77 (32.66%)	0.82	117.76 (16.42%)
Oracle	1.85	263.17 (36.93%)	1.21	173.96 (24.25%)
Llama-3.2-1B-Instruct				
Base	5.00	712.59 (100.0%)	5.00	717.13 (100.0%)
Uplift-RAG	1.97	281.05 (39.44%)	2.10	301.09 (41.98%)
Oracle	2.22	315.19 (44.23%)	2.51	359.50 (50.13%)

Table 3: Comparison of the number of input documents (|N|) and token usage between Base (fixed top-5), Uplift-RAG, and the Oracle (Uplift-RAG Upper Bound).

cation and ranking capabilities, thereby enhancing the overall performance of the RAG system.

• Finally, we observe that the upper bound performance of Uplift-RAG significantly exceeds that of all baseline methods across various datasets and LLMs, with improvements ranging from 15% to 72%. Notably, under oracle conditions, *LLaMA-3.2-1B-Instruct* with Uplift-RAG even outperforms *LLaMA-3-8B-Instruct* without it across all datasets, highlighting the promising potential of our method.

3.3 Ablation Study

To assess the impact of each alignment objective, we perform ablation studies on NQ and TriviaQA by systematically removing one objective at a time from Uplift-RAG. The results presented in Table 2 highlight the significance of each component:

***w/o* point-wise alignment** refers to a variant where the reranker is optimized without the point-wise loss. This substitution leads to a notable decline in performance, which can be attributed to the diminished classification capability of the reranker, causing useful documents to be misclassified as unhelpful. This confirms the importance of point-wise alignment as a foundation for effective ranking.

***w/o* pair-wise alignment** represents the variant that optimizes the reranker without using pair-wise loss. This omission also leads to a noticeable performance drop, suggesting that beyond basic classification, the reranker also needs the ability to effectively rank documents based on their uplift.

***w/o* list-wise alignment** denotes the variant that the reranker is aligned without list-wise loss. The performance drop highlights the importance of leveraging not only coarse-grained uplift labels and ranking orders, but also fine-grained uplift values at the list level to fully capture document utility.

3.4 Token Consumption Analysis

A key advantage of Uplift-RAG over existing methods is that it is not constrained by a fixed top-k document selection strategy. To examine whether this flexibility reduces token consumption, we compare the average number of input documents and the corresponding token consumption between Base, Uplift-RAG, and the Oracle. As shown in Table 3, under the Oracle setting, the average number of truly useful documents is well below five, indicating that fixed top-5 approaches often include noise or misleading content. In contrast, Uplift-RAG dynamically selects fewer but more beneficial documents, achieving better generation quality while

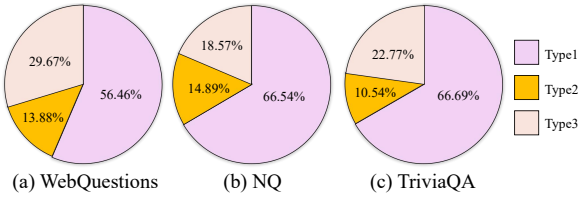


Figure 4: Error Type Distribution on *Llama-3.2-1B-Instruct*. Type1, Type2, and Type3 represent missing all helpful documents, partially missing helpful documents, and containing all helpful documents, respectively.

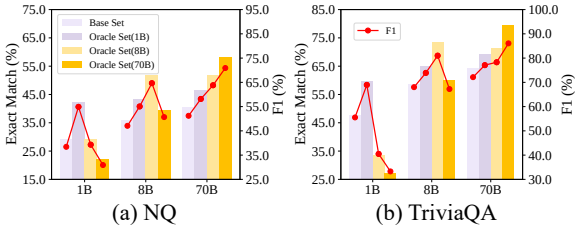


Figure 5: Cross-model oracle transfer between LLMs of different scales. 1B, 8B, and 70B represent the *Llama-3.2-1B-Instruct*, *Llama-3-8B-Instruct*, and *Llama-3-70B-Instruct*, respectively.

using only 16%–42% of the tokens compared to the baseline. Furthermore, Uplift-RAG adapts the number of selected documents based on the capacity of the generator, providing more support for weaker models like *LLaMA-3.2-1B-Instruct*. On the other hand, fixed top-5 methods offer the same number of documents regardless of model strength, highlighting the flexibility and practicality of Uplift-RAG.

3.5 Error Type Analysis

Despite the promising results, there remains a noticeable performance gap between Uplift-RAG and the upper bound. To better understand this gap, we compare the document sets selected by Uplift-RAG with the Oracle document sets, allowing us to assess whether the gap stems mainly from missing helpful documents or including noisy ones. As shown in Figure 4, the primary source of error is the omission of help documents, indicating that the reranker sometimes misclassifies help content as unhelpful. This suggests that the model's ability to identify helpful evidence still requires improvement. Moreover, even with all helpful documents correctly included, performance is not always optimal. This implies that redundant or misleading documents can still introduce noise and degrade generation quality. These findings highlight the need to both improve the recall of helpful documents and enhance the filtering of unhelpful ones.

3.6 Cross-Model Transfer

To investigate the relationship between document preference and model capacity, we conduct a cross-model oracle transfer experiment. As shown in Figure 5, the oracle document set for a smaller LLM can partially generalize to a larger LLM. In contrast, the document set tailored for larger LLMs fails to transfer effectively to smaller ones. This discrepancy arises from the fact that larger LLMs increasingly rely on internal knowledge, often resulting in empty oracle document sets. In comparison, smaller LLMs depend more on external documents to generate high-quality responses. Moreover, while document sets optimized for smaller LLMs can provide some benefit to larger LLMs, they still underperform compared to oracle sets tailored to the larger LLMs, suggesting that unnecessary documents may introduce noise and degrade performance. These results highlight the importance of aligning the reranker with the specific capacities and preferences of the target generator.

3.7 Case Study

In this section, we present two representative cases to demonstrate the effectiveness of Uplift-RAG, as shown in Table 4. In Case 1, the LLM can answer correctly based on its internal knowledge. However, DPA-RAG introduces extraneous information related to Russia and France, which misleads the LLM and results in an incorrect answer. In contrast, Uplift-RAG correctly avoids injecting misleading documents and preserves the correct answer. This illustrates that the inclusion of noisy documents can hinder the model's use of its own knowledge. In Case 2, the LLM alone lacks sufficient internal knowledge to answer the question. Uplift-RAG successfully identifies the helpful document and gives the correct answer. While DPA-RAG also finds this document, it also includes distracting content that confuses the LLM. This indicates that misleading documents can also interfere with the ability of the LLM to utilize external helpful knowledge effectively. These cases highlight the ability of Uplift-RAG to identify helpful documents and avoid the limitations of fixed top-k selection.

4 Related Work

4.1 Preference Alignment in RAG

Preference alignment aims to align the objectives of other components in the RAG pipeline with the preferences of the generator (Wu et al., 2025). Early methods (Ma et al., 2023; Mao et al., 2024;

Case 1: Question:what country did germany invade first in ww1?	**Ground Truth: Belgium**
Naive Generation: Belgium [CORRECT]	
Documents Given by DPA-RAG: [1.Germany disregarded Belgian neutrality and invaded the country to launch an offensive towards Paris. 2.Germany entered into World War I on August 1, 1914, when it declared war on Russia. 3.In accordance with its war plan, it ignored Russia and moved first against France–declaring war on August 3 and sending its main armies through Belgium to attack Paris from the north. 4.Russia entered into World War I on August 1, 1914, when Germany declared war on it. 5.The history of Belgium in World War I traces Belgium's role between the German invasion in 1914.] **Answer Given by DPA-RAG**: Russia [WRONG]	
Documents Given by Uplift-RAG: [] **Answer Given by Uplift-RAG**: Belgium [CORRECT]	
Case 2: Question:who is john cena married to?	**Ground Truth: Elizabeth Huberdeau**
Naive Generation: Shelly Beattie [WRONG]	
Documents Given by DPA-RAG: [1.John Cena: While promoting his 2009 film "12 Rounds", Cena announced his engagement to Elizabeth Huberdeau. They were married on July 11, 2009. 2.David Otunga: Seven months after meeting singer and actress Jennifer Hudson, Otunga proposed marriage on Hudson's 27th birthday in 2008. 3.Nikki Bella: On April 15, 2018, John Cena called off the engagement and cancelled their wedding which was planned for May 5, 2018. 4.John Cena: John Cena John Felix Anthony Cena Jr. (; born April 23, 1977) is an American professional wrestler, actor, rapper, and television host. 5.John Cena: Cena proposed marriage to Bella after the match and she accepted.] **Answer Given by DPA-RAG**: John Cena is not married. [WRONG]	
Documents Given by Uplift-RAG: [1.John Cena: While promoting his 2009 film "12 Rounds", Cena announced his engagement to Elizabeth Huberdeau. They were married on July 11, 2009.] **Answer Given by Uplift-RAG**: Elizabeth Huberdeau [CORRECT]	

Table 4: Case studies of solving questions with Naive Generation, DPA-RAG, and Uplift-RAG on the *Llama-3-8B-Instruct*. Text in purple text highlights both helpful and potentially misleading information from the document, and [CORRECT] or [WRONG] indicates whether the answer is correct or incorrect, respectively.

Wang et al., 2025; Zhang et al., 2025b,a) align the query rewriter by using generation quality or ranking feedback as a supervision signal and optimizing it. Subsequently, some studies (Zhang et al., 2024b; Salemi and Zamani, 2024; Zamani and Bendersky, 2024; Qu et al., 2024; Xu et al., 2024) focus on aligning the retriever based on feedback from the generator. More recently, efforts have shifted toward aligning the reranker (Ke et al., 2024; Dong et al., 2025; Jia et al., 2024) to further improve the overall quality of retrieved documents. Despite these advancements, existing methods primarily use the performance of the generator conditioned on retrieved documents as a direct supervision signal, making it difficult to disentangle the actual contribution of the documents from inherent capabilities of LLMs. In contrast, our approach uses the marginal improvement in generation quality when a specific document is utilized, compared to when it is not, as the supervision signal. This enables a more accurate estimation of each document's contribution, allowing the reranker to better identify truly helpful content for the generator.

4.2 Uplift Modeling

Uplift modeling, a concept widely adopted in marketing, aims to measure the behavioral difference between individuals who receive a treatment promotional offer (the treated group) and those who do not (the control group) (Liu et al., 2022; Zhang et al., 2021; Wang et al., 2024). This setup closely parallels the RAG scenario, where the "treatment" corresponds to incorporating a specific document during generation, and the goal is to assess its impact on response quality. Despite extensive research on uplift modeling in marketing (He et al., 2024; Zhang et al., 2024c; Ibragimov and Vakhrushev, 2024), it has not yet been explored in the context of RAG. In this paper, we propose to leverage uplift as a supervision signal to align the preferences of the reranker with documents that meaningfully improve generation quality, thereby enhancing the overall effectiveness of the RAG system.

5 Conclusion

In this paper, we propose Uplift-RAG, a novel uplift-driven preference alignment framework for RAG. Uplift-RAG first computes the uplift of each document by comparing response quality with and without that document. Subsequently, we leverage uplift as the alignment signal to achieve three alignment objectives, enabling the reranker to simultaneously possess both classification and ranking capabilities. Extensive experiments demonstrate that Uplift-RAG not only reduces computational resources but also improves the overall performance.

Limitations

While our uplift-driven knowledge preference alignment framework demonstrates promising results, we acknowledge two key limitations: (1) Although our lightweight reranker contributes to improved retrieval, it still falls short of effectively identifying truly useful documents, resulting in a noticeable gap from the upper bound performance. We will explore the use of LLM as the reranker to enhance the accuracy of document utility assessment in the future work. (2) Our current evaluation framework measures uplift at the level of individual documents. However, the input to the LLM is a set of documents. Extending our uplift-based preference modeling from single documents to document sets remains a key challenge for future research.

Ethics Statement

This work was conducted in strict compliance with the ACL Ethics Policy. All datasets and models used for experiment are publicly available. Furthermore, our work aims to bridge the gap between retrieval objectives and LLMs' knowledge needs. We do not foresee any negative ethical impacts arising from our work.

Acknowledgements

This work was funded by the National Natural Science Foundation of China (62472426), fund for building world-class universities (disciplines) of Renmin University of China. Work partially done at Beijing Key Laboratory of Research on Large Models and Intelligent Governance, and Engineering Research Center of Next-Generation Intelligent Search and Recommendation, Ministry of Education.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1533–1544.

Guanting Dong, Yutao Zhu, Chenghao Zhang, Zechen Wang, Ji-Rong Wen, and Zhicheng Dou. 2025. Understand what llm needs: Dual preference alignment for retrieval-augmented generation. In *Proceedings of the ACM on Web Conference 2025*, pages 4206–4225.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6491–6501.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yixin Dai, Jiawei Sun, Haofen Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2:1.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

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

Bowei He, Yunpeng Weng, Xing Tang, Ziqiang Cui, Zexu Sun, Liang Chen, Xiuqiang He, and Chen Ma. 2024. Rankability-enhanced revenue uplift modeling framework for online marketing. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5093–5104.

Bulat Ibragimov and Anton Vakhrushev. 2024. Uplift modelling via gradient boosting. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1177–1187.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong Park. 2024. Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*.

Pengyue Jia, Derong Xu, Xiaopeng Li, Zhaocheng Du, Xiangyang Li, Xiangyu Zhao, Yichao Wang, Yuhao Wang, Huifeng Guo, and Ruiming Tang. 2024. Bridging relevance and reasoning: Rationale distillation in retrieval-augmented generation. *arXiv preprint arXiv:2412.08519*.

Jiajie Jin, Yutao Zhu, Guanting Dong, Yuyao Zhang, Xinyu Yang, Chenghao Zhang, Tong Zhao, Zhao Yang, Zhicheng Dou, and Ji-Rong Wen. 2025. Flashrag: A modular toolkit for efficient retrieval-augmented generation research. In *Proceedings of the ACM on Web Conference 2025*.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly

supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Zixuan Ke, Weize Kong, Cheng Li, Mingyang Zhang, Qiaozhu Mei, and Michael Bendersky. 2024. Bridging the preference gap between retrievers and LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, and 1 others. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Yuchen Li, Hengyi Cai, Rui Kong, Xinran Chen, Jiamin Chen, Jun Yang, Haojie Zhang, Jiayi Li, Jiayi Wu, Yiqun Chen, and 1 others. 2025. Towards ai search paradigm. *arXiv preprint arXiv:2506.17188*.

Zihan Liu, Yun Luo, Lirong Wu, Zicheng Liu, and Stan Z Li. 2022. Towards reasonable budget allocation in untargeted graph structure attacks via gradient debias. *Advances in neural information processing systems*.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. 2023. Query rewriting in retrieval-augmented large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5303–5315.

Shengyu Mao, Yong Jiang, Boli Chen, Xiao Li, Peng Wang, Xinyu Wang, Pengjun Xie, Fei Huang, Huajun Chen, and Ningyu Zhang. 2024. RaFe: Ranking feedback improves query rewriting for RAG. In *Findings of the Association for Computational Linguistics: EMNLP 2024*.

Changle Qu, Sunhao Dai, Xiaochi Wei, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, Jun Xu, and Ji-Rong Wen. 2024. Towards completeness-oriented tool retrieval for large language models. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 1930–1940.

Alireza Salemi and Hamed Zamani. 2024. Evaluating retrieval quality in retrieval-augmented generation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2395–2400.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2024. REPLUG: Retrieval-augmented black-box language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*.

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

Wenjie Wang, Changsheng Wang, Fuli Feng, Wentao Shi, Daizong Ding, and Tat-Seng Chua. 2024. Uplift modeling for target user attacks on recommender systems. In *Proceedings of the ACM Web Conference 2024*, pages 3343–3354.

Yile Wang, Peng Li, Maosong Sun, and Yang Liu. 2023. Self-knowledge guided retrieval augmentation for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*.

Yujing Wang, Hainan Zhang, Liang Pang, Binghui Guo, Hongwei Zheng, and Zhiming Zheng. 2025. Maferw: Query rewriting with multi-aspect feedbacks for retrieval-augmented large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 25434–25442.

Jiayi Wu, Hengyi Cai, Lingyong Yan, Hao Sun, Xiang Li, Shuaiqiang Wang, Dawei Yin, and Ming Gao. 2025. PA-RAG: RAG alignment via multi-perspective preference optimization. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. 2024. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th international ACM SIGIR conference on research and development in information retrieval*, pages 641–649.

Qiancheng Xu, Yongqi Li, Heming Xia, and Wenjie Li. 2024. Enhancing tool retrieval with iterative feedback from large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen2.5 technical report. *arXiv preprint arXiv:2505.09388*.

Zichun Yu, Chenyan Xiong, Shi Yu, and Zhiyuan Liu. 2023. Augmentation-adapted retriever improves generalization of language models as generic plug-in. In *Proceedings of ACL*.

Hamed Zamani and Michael Bendersky. 2024. Stochastic rag: End-to-end retrieval-augmented generation through expected utility maximization. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2641–2646.

Hengran Zhang, Ruqing Zhang, Jiafeng Guo, Maarten de Rijke, Yixing Fan, and Xueqi Cheng. 2024a. Are large language models good at utility judgments? In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1941–1951.

Kepu Zhang, Zhongxiang Sun, Weijie Yu, Xiaoxue Zang, Kai Zheng, Yang Song, Han Li, and Jun Xu. 2025a. Qe-rag: A robust retrieval-augmented generation benchmark for query entry errors. *arXiv preprint arXiv:2504.04062*.

Kepu Zhang, Zhongxiang Sun, Xiao Zhang, Xiaoxue Zang, Kai Zheng, Yang Song, and Jun Xu. 2025b. Trigger3: Refining query correction via adaptive model selector. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 13260–13268.

LingXi Zhang, Yue Yu, Kuan Wang, and Chao Zhang. 2024b. ARL2: Aligning retrievers with black-box large language models via self-guided adaptive relevance labeling. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Weijia Zhang, Jiuyong Li, and Lin Liu. 2021. A unified survey of treatment effect heterogeneity modelling and uplift modelling. *ACM Computing Surveys (CSUR)*, 54(8):1–36.

Xin Zhang, Kai Wang, Zengmao Wang, Bo Du, Shiwei Zhao, Runze Wu, Xudong Shen, Tangjie Lv, and Changjie Fan. 2024c. Temporal uplift modeling for online marketing. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6247–6256.

Dataset	# Train	# Dev	# Test
WebQuestions	3,778	2,032	2,032
Natural Question (NQ)	79,168	8,757	3,610
TriviaQA	78,785	8,837	11,313

Table 5: Statistics of the experimental datasets.

A Appendix

A.1 Dataset Details

In this section, we introduce the detailed description of the datasets used in this paper. The statistics of these datasets are shown in Table 5.

WebQuestions (Berant et al., 2013) consists of real user questions sourced from Google Suggest, with corresponding answers annotated based on the Freebase knowledge graph. It focuses on short, entity-centric answers derived from a large-scale knowledge base, making it a benchmark for knowledge-based question answering systems.

NQ (Kwiatkowski et al., 2019) presents a large-scale, realistic QA benchmark built from anonymized Google search queries, where each question is paired with a Wikipedia page and annotated by humans to indicate both short and long-form answers.

TriviaQA (Joshi et al., 2017) is a large-scale question answering dataset composed of independently authored trivia questions that span a broad range of domains and topics. Each question is paired with multiple supporting evidence documents, retrieved from web sources or Wikipedia, where the correct answer is located.

A.2 Baselines

In this section, we introduce the baseline methods used for comparison in our experiments:

Naive Generation refers to generating responses directly from the LLM without any external retrieval or augmentation.

Standard RAG is the original RAG framework that combines a retriever with a generator to produce responses based on retrieved documents.

BGE-reranker (Xiao et al., 2024) utilizes *bge-reranker-base* for reranking retrieved passages to improve the quality of the provided documents.

AAR (Yu et al., 2023) is a retrieval plug-in that learns retrieval preferences from a known source LM to improve document retrieval for unseen target LMs without joint fine-tuning.

REPLUG (Shi et al., 2024) optimizes the retriever by using the output probability of a black-box LLM to improve RAG performance.

SKR (Wang et al., 2023) enhances retrieval by guiding LLMs to decide whether to retrieve based on their self-knowledge, avoiding unnecessary or harmful external information.

Adaptive-RAG (Jeong et al., 2024) dynamically selects between retrieval and non-retrieval strategies based on query complexity using a trained classifier to improve QA accuracy and efficiency.

DPA-RAG (Dong et al., 2025) aligns the reranker and the generator by modeling and integrating LLM-specific knowledge preferences to improve the reliability of RAG systems.

A.3 More Experiments

In this section, we further evaluate the feasibility and effectiveness of Uplift-RAG across LLMs of different sizes and families. Specifically, we test the upper bound performance of Uplift-RAG on a diverse set of models, including *LLaMA-3-70B-Instruct* [2] (Grattafiori et al., 2024), *LLaMA-3.2-3B-Instruct* [3] (Grattafiori et al., 2024), *GPT-3.5-Turbo* [4] (Achiam et al., 2023), *Qwen-2.5-14B-Instruct* [5] (Yang et al., 2025), *Qwen-2.5-7B-Instruct* [6] (Yang et al., 2025), and *Qwen-2.5-3B-Instruct* [7] (Yang et al., 2025). The results are shown in Table 6.

From the results, we can find that Uplift-RAG (Upper Bound) consistently achieves substantial performance improvements across all model sizes and families. These results validate the generalizability and scalability of Uplift-RAG across diverse model backbones.

A.4 Prompts

In this section, we present the prompts used in our experiments. Specifically, we adopt the prompt provided in FlashRAG (Jin et al., 2025), the prompt template for naive generation is shown below:

[2] https://huggingface.co/meta-llama/Llama-3-70B-Instruct

[3] https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct

[4] https://platform.openai.com/playground/chat?models=gpt-3.5-turbo

[5] https://huggingface.co/Qwen/Qwen2.5-14B-Instruct

[6] https://huggingface.co/Qwen/Qwen2.5-7B-Instruct

[7] https://huggingface.co/Qwen/Qwen2.5-3B-Instruct

Methods	Llama-3-70B-Instruct				Llama-3.2-3B-Instruct				GPT-3.5-Turbo			
	NQ		TriviaQA		NQ		TriviaQA		NQ		TriviaQA	
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
Naive Generation	33.68	45.28	66.25	72.87	17.64	27.23	44.17	51.03	35.45	48.31	69.61	78.08
Standard RAG	40.49	51.21	64.29	72.14	38.55	48.73	60.74	68.39	41.74	53.11	63.68	72.23
Uplift-RAG (Upper Bound)	58.25	70.91	79.41	86.07	52.18	64.87	71.15	78.90	53.65	67.19	74.29	82.42

Methods	Qwen-2.5-14B-Instruct				Qwen-2.5-7B-Instruct				Qwen-2.5-3B-Instruct			
	NQ		TriviaQA		NQ		TriviaQA		NQ		TriviaQA	
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
Naive Generation	21.71	31.21	55.08	62.06	14.40	23.28	43.66	50.78	12.13	19.35	32.21	38.66
Standard RAG	32.88	45.97	58.75	68.97	36.37	47.76	61.20	69.88	37.95	47.72	57.42	65.68
Uplift-RAG (Upper Bound)	43.18	58.20	68.14	77.31	48.50	62.31	69.75	78.65	51.10	64.56	67.66	76.68

Table 6: Performance comparison between Uplift-RAG (Upper Bound) and base methods on two datasets.

Prompt Template of Naive Generation

System Prompt: Answer the question based on your own knowledge. Only give me the answer and do not output any other words.
User Prompt: Question: {question}
Answer:

The prompt template for RAG-based methods is shown below:

Prompt Template of RAG-based Methods

System Prompt: Answer the question based on the given document. Only give me the answer and do not output any other words. The following are given documents. {reference}
User Prompt: Question: {question}
Answer: