DARE-RAG: Difficulty-Aware Retrieval Expansion for Retrieval-Augmented Generation

Lixiang Zhu[1], Minghao Li[2], Zhubo Liu[1], and Chao Wu[2⋆]

[1] Shangxing Innovation Division, Tianjin University of Science and Technology, Tianjin 300457, China
[2] College of Artificial Intelligence, Tianjin University of Science and Technology, Tianjin 300457, China
{lixiangzhu, 23101107, liuzhubo, superwoo}@mail.tust.edu.cn

Abstract. Retrieval-Augmented Generation (RAG) systems face a fundamental trade-off: query expansion can improve retrieval effectiveness for ambiguous or underspecified queries, yet indiscriminate expansion introduces unnecessary latency and retrieval noise. Existing RAG pipelines typically apply expansion uniformly, failing to distinguish between easy and retrieval-challenging queries. To address this issue, we propose DARE-RAG, an adaptive retrieval framework that activates LLM-based query expansion only for retrieval-challenging queries. Our method formulates expansion activation as a lightweight binary classification problem using probe retrieval signals, including score margin, variance, entropy, query length, and lexical specificity. A lightweight MLP predicts whether expansion is likely to improve retrieval quality, and expansion is triggered only when the predicted confidence exceeds a percentile-calibrated threshold. DARE-RAG further integrates a dual-path hybrid retrieval architecture combining BM25 sparse retrieval and BGE dense retrieval, fused via Reciprocal Rank Fusion (RRF), followed by a Cross-Encoder reranker for context refinement. Experiments on NQ-Open and HotpotQA demonstrate that DARE-RAG consistently improves retrieval effectiveness and end-to-end QA accuracy while substantially reducing average end-to-end latency compared with corresponding always-expand variants of BM25, BGE-m3, and their RRF-fused hybrid retriever. Extensive ablation studies and efficiency analyses verify the effectiveness of our utility-guided expansion strategy.

Keywords: Retrieval-Augmented Generation,Adaptive Query Expansion,Query Difficulty Estimation,Hybrid Retrieval,Efficiency-Aware Retrieval

1 Introduction

Retrieval-Augmented Generation (RAG) improves factual grounding by augmenting Large Language Models with external knowledge sources [9, 14]. By retrieving external evidence before generation, RAG systems significantly reduce hallucinations and improve knowledge-intensive reasoning.

⋆ Corresponding author.

Despite recent progress, current RAG pipelines still face three major challenges:

Semantic Ambiguity. User queries are often short, underspecified, or ambiguous, limiting retrieval effectiveness and reducing evidence recall [2].

Retrieval Imbalance. Sparse retrievers such as BM25 provide strong lexical matching but weak semantic generalization, whereas dense retrievers capture semantic similarity yet may overlook exact lexical evidence.

Efficiency–Noise Trade-off. Query expansion can improve retrieval effectiveness for retrieval-challenging queries, but indiscriminate expansion introduces substantial latency and retrieval noise for already well-grounded queries.

Recent adaptive RAG methods primarily adapt retrieval during generation through iterative decoding, token-level uncertainty estimation, or retrieval routing, often introducing considerable generation-time overhead. In contrast, DARE-RAG formulates adaptive query expansion as a lightweight retrieval-stage utility prediction problem using probe retrieval statistics before generation begins.

Specifically, DARE-RAG performs lightweight probe retrieval using both sparse and dense retrievers and constructs compact retrieval uncertainty features to estimate the potential utility of query expansion. Query expansion is activated only for high-uncertainty queries, followed by hybrid retrieval and reranking. This retrieval-stage design avoids iterative retrieval–generation interactions and directly estimates the marginal utility of expansion from lightweight retrieval uncertainty signals, resulting in negligible additional overhead.

Our contributions are summarized as follows:

1. We propose DARE-RAG, a pre-retrieval adaptive expansion framework that selectively activates LLM-based query expansion using lightweight probe retrieval signals before generation begins.
2. We introduce a compact retrieval uncertainty representation that distills top-10 probe retrieval statistics into an 8-dimensional feature vector for expansion utility estimation.
3. We design a margin-threshold supervision strategy that converts continuous retrieval gains (ΔMRR) into robust binary labels for stable gate optimization.
4. Extensive experiments on NQ-Open and HotpotQA demonstrate that DARE-RAG preserves most retrieval improvements of uniform expansion while activating expansion for only a small subset of retrieval-challenging queries, substantially reducing inference latency and retrieval overhead.

2 Related Work

2.1 Query Difficulty Estimation and Adaptive Retrieval

Query Difficulty Estimation (QDE) has been widely studied in traditional information retrieval. Early approaches estimate difficulty using hand-crafted features derived from score distributions, query clarity, or pseudo-relevance feedback statistics [3, 8].

With the emergence of LLM-based RAG systems, recent work has shifted toward adaptive retrieval guided by generative signals. Self-RAG [1] employs self-reflection to dynamically retrieve and critique passages during generation. FLARE [11] activates retrieval based on token-level generation confidence. Adaptive-RAG [10] adapts retrieval behavior according to query complexity, while RECOMP [19] jointly optimizes retrieval and generation through end-to-end training.

Unlike these approaches, DARE-RAG formulates query expansion as a selective activation problem. DARE-RAG makes expansion decisions directly from probe retrieval statistics before generation begins, thereby avoiding generation-time decoding overhead.

Beyond static QDE heuristics, DARE-RAG learns retrieval feature representations from dual-retriever score distributions and cross-retriever agreement signals.

2.2 LLM-based Query Expansion

Recent retrieval systems increasingly employ LLM-generated query reformulation to improve semantic coverage and retrieval recall. HyDE [6] generates hypothetical relevant documents to guide dense retrieval, while other approaches leverage paraphrasing and decomposition strategies for query augmentation [2].

Unlike these approaches that uniformly apply expansion to all queries, DARE-RAG selectively activates expansion only when probe retrieval signals indicate potential retrieval benefit.

Table 1. Limitations of representative adaptive retrieval approaches.

Method	**Primary Limitation**
Self-RAG	Requires generation-time self-reflection and iterative decoding overhead
FLARE	Depends on token-level uncertainty during autoregressive decoding
Adaptive-RAG	Primarily focuses on retrieval strategy selection under varying query complexity rather than fine-grained expansion activation
RECOMP	Requires expensive end-to-end optimization and retraining
Traditional QDE	Relies heavily on static handcrafted retrieval heuristics

These limitations motivate the development of an efficient retrieval-stage framework for estimating expansion benefits, enabling selective query expansion without generation-time control or end-to-end retraining.

2.3 Hybrid Retrieval and Rank Fusion

Sparse retrieval methods such as BM25 provide strong lexical matching, while dense retrieval models capture semantic similarity [4, 12]. Hybrid retrieval systems combine sparse and dense retrievers to improve robustness and recall across diverse query types [4, 7, 12].

Because sparse and dense retrieval scores are not directly comparable, rank-based fusion approaches such as Reciprocal Rank Fusion (RRF) are widely adopted in modern hybrid retrieval pipelines [5, 16].

2.4 Reranking and Context Optimization

Initial retrieval candidates often contain noisy or weakly relevant passages. Cross-Encoder rerankers improve retrieval precision by modeling fine-grained query-document interactions [15, 17].

Recent studies have shown that pointwise reranking strategies can further improve ranking calibration and downstream QA performance [18]. Motivated by these advantages, DARE-RAG incorporates a pointwise-optimized Cross-Encoder reranker to refine fused retrieval candidates.

3 Methodology

3.1 System Overview

DARE-RAG consists of four stages. First, it performs expansion benefit estimation to determine whether query expansion should be activated. Second, it employs dual-path hybrid retrieval to leverage the complementary strengths of sparse and dense retrievers. Third, retrieval results are combined using Reciprocal Rank Fusion (RRF). Finally, a Cross-Encoder reranker refines the ranking of candidate documents to improve retrieval precision and downstream question-answering performance.

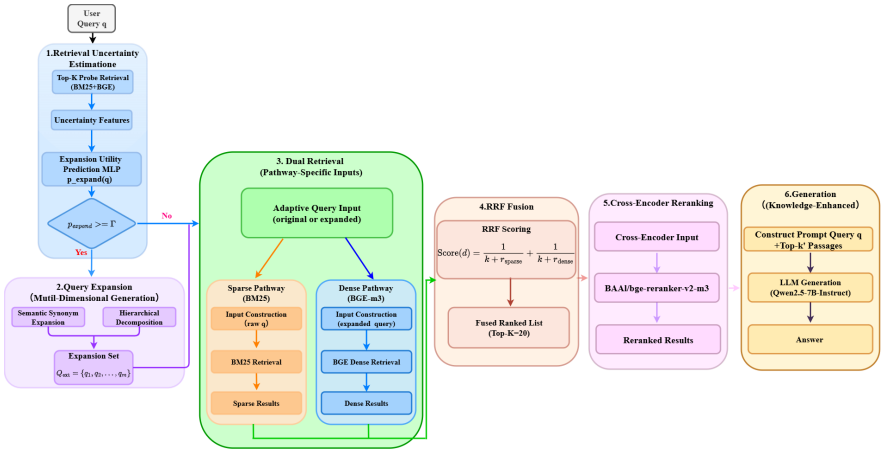

Fig. 1. Overall architecture of the DARE-RAG framework.

3.2 Feature Construction

We formulate expansion activation as a binary classification task based on lightweight probe retrieval signals. Given a query q, we first retrieve the Top-10 candidates independently from both the dense and sparse retrievers to obtain score sets and index lists:

$$\mathcal{S}_q^D = \{s_1^D, s_2^D, \ldots, s_{10}^D\}, \quad \mathcal{I}_q^D = \{d_1^D, \ldots, d_{10}^D\} \tag{1}$$

$$\mathcal{S}_q^S = \{s_1^S, s_2^S, \ldots, s_{10}^S\}, \quad \mathcal{I}_q^S = \{d_1^S, \ldots, d_{10}^S\} \tag{2}$$

From these probe signals, we construct an 8-dimensional retrieval feature vector $\mathbf{x}_q$:

$$\mathbf{x}_q = [\Delta s_{12}, \Delta s_{15}, H^D, H^S, \sigma_D^2, \mathcal{O}_5, L_q, \rho_s]^\top \tag{3}$$

where each component is computed as follows:

Score Margins. $\Delta s_{12} = s_1^D - s_2^D$ and $\Delta s_{15} = s_1^D - s_5^D$ measure the dominance of the top-ranked document and the separation among highly ranked documents, respectively.

Retrieval Entropy. We compute softmax-normalized probability distributions $p_i^D = \frac{\exp(s_i^D)}{\sum_j \exp(s_j^D)}$ and $p_i^S = \frac{\exp(s_i^S)}{\sum_j \exp(s_j^S)}$, then calculate entropy $H^D = -\sum_i p_i^D \log p_i^D$ and H^S analogously.

Score Variance. $\sigma_D^2 = \mathrm{Var}(\mathcal{S}_q^D)$ captures the dispersion of dense retrieval scores and provides an indicator of retrieval confidence.

Cross-Retriever Overlap. $\mathcal{O}_5 = \frac{|\mathcal{I}_q^D[:5] \cap \mathcal{I}_q^S[:5]|}{5}$ measures the agreement between lexical (BM25) and semantic (BGE) retrievers. Low overlap often indicates ambiguity or disagreement between lexical and semantic retrieval signals. We compute overlap only over the top-5 candidates because agreement among highly ranked documents is generally more informative and empirically correlates better with final retrieval quality than deeper overlap statistics.

Query Length. L_q counts the number of words in q.

Lexical Specificity. $\rho_s = \frac{1}{L_q} \sum_{w \in q} \mathrm{IDF}(w)$ measures the average inverse document frequency of query tokens. Queries containing rare or highly discriminative terms generally provide stronger retrieval constraints, whereas low-specificity queries are more likely to benefit from expansion.

Together, these features capture complementary aspects of retrieval uncertainty and provide an efficient approximation of expansion utility with negligible computational overhead. The feature set was selected based on prior retrieval uncertainty studies and preliminary validation experiments, where additional features yielded only marginal improvements while increasing gate complexity.

3.3 MLP-based Expansion Gate

We train a lightweight three-layer MLP to predict whether query expansion will improve retrieval quality for a given query.

Supervision Signal via ΔMRR. Instead of using binary Hit@k improvements, we adopt the change in Mean Reciprocal Rank (ΔMRR) as the supervision signal, which better captures the position-sensitive nature of evidence utility in generation:

$$\mathrm{MRR}(\mathcal{D}, a) = \begin{cases} \frac{1}{\min\{i | d_i \supset a\}} & \text{if } \exists d_i \in \mathcal{D} \text{ containing } a \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where $\mathcal{D} = [d_1, \ldots, d_K]$ is the ranked retrieval list and a is the gold answer. For each training query, we compute:

$$\Delta\mathrm{MRR} = \mathrm{MRR}(\mathcal{D}_{\mathrm{ext}}, a) - \mathrm{MRR}(\mathcal{D}_{\mathrm{orig}}, a) \tag{5}$$

with $\mathcal{D}_{\mathrm{orig}}$ and $\mathcal{D}_{\mathrm{ext}}$ obtained from the hybrid retrieval pipeline using the original and expanded queries, respectively.

Label Construction. We assign binary labels via a margin-based thresholding strategy:

$$y = \begin{cases} 1 & \text{if } \Delta\mathrm{MRR} \geq \tau \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where $\tau = 0.2$. We set $\tau = 0.2$ based on validation-set sensitivity analysis (see Appendix B), which filters out a large portion of marginal ΔMRR gains that do not translate to generation improvements. This margin constraint reduces false positives by ensuring expansion is triggered only for queries with substantial retrieval benefit. Moreover, expansion utility exhibits highly skewed and sparse gain distributions, where only a small subset of queries benefits meaningfully from expansion. Direct regression on continuous ΔMRR values was empirically unstable due to heavy-tailed gain distributions and large variance among expansion-beneficial queries, whereas binary margin-based supervision yielded more stable optimization.

Training Objective. The gate outputs an unbounded logit $z_q = \text{MLP}(\mathbf{x}_q)$ and is optimized using Binary Cross Entropy with Logits (BCEWithLogits) loss.

During inference, the expansion probability is $p_{\text{expand}} = \sigma(z_q)$, and expansion is triggered if $p_{\text{expand}} > \Gamma$, where Γ is calibrated on validation data to achieve a target expansion rate (e.g., 20%). The threshold Γ is determined on the validation set via percentile-based calibration. Specifically, we select Γ such that only the top P% of queries ranked by confidence trigger expansion, where P denotes the target expansion budget (e.g., $P = 20$). Unless otherwise stated, we use $P = 20$ based on validation-set efficiency–effectiveness trade-off analysis.

Table 2. Confidence gate classification performance on NQ-Open.

Model	Accuracy	Precision	Recall	AUROC
MLP Gate	84.2%	79.5%	76.8%	0.88

The relatively high AUROC suggests that probe retrieval features provide useful discriminative signals despite the imbalanced distribution of expansion-beneficial queries.

Uncertainty Distribution Analysis To further validate the effectiveness of the proposed retrieval feature representation, we analyze the distribution of gate expansion scores across queries that benefit from expansion and those that do not.

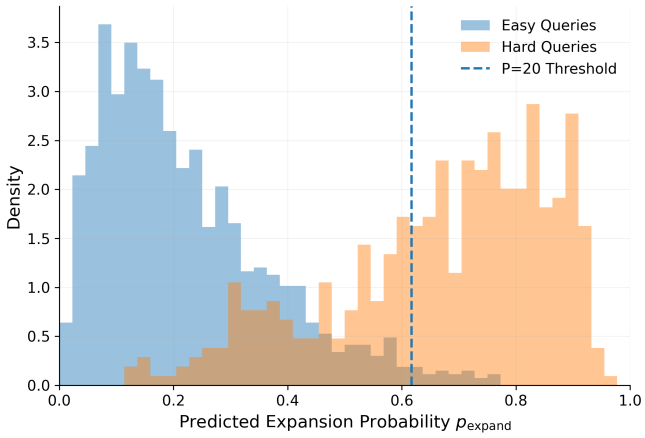

Fig. 2. Distribution of predicted expansion probabilities for easy and hard queries. Hard queries exhibit notably higher expansion probabilities, indicating effective separation between expansion-beneficial and well-grounded queries.

Figure 2 visualizes the predicted expansion probabilities p_{expand} on the NQ-Open validation set. Queries with $\Delta\text{MRR} \geq \tau$ are labeled as expansion-beneficial queries, while the remaining queries are categorized as non-beneficial queries.

We observe substantial distributional separation between the two groups. Hard queries exhibit substantially higher expansion probabilities, whereas well-grounded queries are concentrated near low-confidence regions. These results demonstrate that probe retrieval features effectively separate expansion-beneficial queries from easy ones.

Moreover, the relatively small overlap region suggests that the gate can suppress a substantial portion of unnecessary expansions for already well-grounded queries.

3.4 Adaptive Query Expansion

When the confidence gate triggers expansion ($p_{\text{expand}} > \Gamma$), we use Qwen2.5-7B-Instruct to generate semantically diverse query variants.

The prompt encourages:

- lexical substitution to improve sparse retrieval coverage;
- query decomposition to recover implicit semantic constraints.

To reduce semantically redundant variants, generated variants are filtered using embedding cosine similarity computed by the same BGE-m3 encoder used for dense retrieval. Variants with similarity greater than 0.92 to the original query are discarded.

For sparse retrieval, all retained variants are concatenated:

$$\mathbf{q}_{\text{sparse}} = \mathbf{q} \oplus [\text{SEP}] \oplus \mathbf{q}_1 \oplus \ldots \oplus \mathbf{q}_m \tag{7}$$

For dense retrieval, we apply normalized mean pooling over all query embeddings:

$$\mathbf{v}_{\text{dense}} = \text{Norm}\left(\frac{1}{m+1}\sum_{i=0}^{m}\text{Norm}(\mathbf{v}_i)\right) \tag{8}$$

3.5 Cross-Encoder Reranking

The RRF-fused Top-20 candidates $\mathcal{C}_{\text{top-20}}$ are refined using a Cross-Encoder (BAAI/bge-reranker-v2-m3). To maintain inference efficiency, we construct a single augmented query representation for reranking:

$$q_{\text{aug}} = \begin{cases} q \oplus \texttt{[SEP]} \oplus q_1 \oplus \cdots \oplus q_m, & \text{if expansion triggered} \\ q, & \text{otherwise} \end{cases} \tag{9}$$

The reranker computes a fine-grained relevance score $s(d) = \text{CE}(q_{\text{aug}}, d)$ for each $d \in \mathcal{C}_{\text{top-20}}$, and selects the final Top-5 passages for generation. This design avoids reranking each candidate against multiple query variants, thereby reducing inference overhead. Input sequences are truncated to a maximum of 512 tokens, and batched inference ($B = 16$) yields an average end-to-end latency of approximately 44.6 ms per query.

4 Experiments

4.1 Experimental Setup

Evaluation Datasets

NQ-Open. We evaluate retrieval and generation performance on NQ-Open [13], an open-domain QA benchmark containing 8,757 development queries with free-form answers. Each query q is associated with one or more gold answers $\mathcal{A}_q = \{a_1, \ldots, a_m\}$.

Data Split Protocol. To supervise the confidence gate, we partition the NQ-Open development set into train/validation/test splits with a ratio of 60/20/20. The training split is used for MLP optimization using ΔMRR labels, while the validation split is used for hyperparameter tuning and threshold calibration, including τ and Γ. Final retrieval and QA results are reported only on the held-out test split. All baselines are evaluated using the same corpus, retrieval pipeline, and split configuration to ensure fair comparison.

HotpotQA. We further evaluate DARE-RAG on HotpotQA [20], a multi-hop QA benchmark requiring compositional reasoning across multiple supporting documents. Compared with NQ-Open, HotpotQA contains consistently more ambiguous and compositional queries, making it suitable for evaluating adaptive expansion under challenging retrieval conditions. To assess cross-dataset generalization, the confidence gate is trained only on NQ-Open and directly transferred to HotpotQA without retraining. The expansion generator, retrieval models, and confidence gate are kept unchanged during transfer. Unless otherwise specified, all experiments use the same retrieval corpus and hybrid retrieval pipeline as the NQ-Open setup.

Retrieval Corpus: Wikipedia Subset Following efficient RAG research [4], we construct a compact retrieval corpus from the November 2023 Wikipedia dump. Each article is segmented into overlapping 100-word passages with a stride of 20, yielding approximately 1M retrieval passages in total. Both BM25 (sparse) and BGE-m3 (dense) indexes are built on the resulting corpus.
Experimental Scope. Due to hardware constraints, we use a compact retrieval corpus of approximately 1M Wikipedia passages rather than full-scale DPR-style corpora containing tens of millions of passages. Consequently, absolute retrieval and QA scores are lower than those reported on large-scale benchmarks. Our focus is therefore on evaluating the relative effectiveness–efficiency trade-offs introduced by adaptive expansion under a controlled retrieval setting.

Evaluation Protocol. For each NQ-Open query q, we retrieve passages from the Wikipedia corpus and evaluate: (1) *Retrieval quality*: whether the *final top-K passages after reranking* contain any gold answer $a \in \mathcal{A}_q$ (Hit@K, MRR@K); (2) *Generation quality*: whether the LLM-generated answer matches $\mathcal{A}_q$ (Exact

Match, F1). Answer matching follows standard normalized exact-match evaluation used in open-domain QA, including lowercase normalization and punctuation removal.

Evaluation Metrics All experiments use a fixed train/validation/test split. For robustness, we report results averaged over five random initialization seeds $\{42, 123, 456, 789, 1024\}$.

Implementation Details We conducted all experiments on a workstation with an NVIDIA RTX 4090 GPU and an AMD Ryzen 9 7950X CPU. Sparse retrieval was implemented using RankBM25, while dense retrieval employed BAAI/bge-m3 with a FAISS IndexHNSWFlat index. Both sparse and dense retrievers independently retrieved the top-10 candidates before Reciprocal Rank Fusion (RRF). For adaptive query expansion and final answer generation, we used Qwen2.5-7B-Instruct. Retrieved candidates were further refined using the Cross-Encoder reranker BAAI/bge-reranker-v2-m3.

The confidence gate was implemented as a three-layer MLP with architecture $8 \to 32 \to 16 \to 1$, using ReLU activations and dropout rate of 0.2. The model was trained using AdamW with learning rate 3×10^{-4} and batch size 128 for up to 20 epochs, with early stopping based on validation ΔMRR performance. Latency is measured under single-query inference with GPU warm-up enabled and includes retrieval, optional expansion, reranking, and generation overhead.

For fair comparison, Adaptive-RAG was re-implemented under the same retrieval framework as DARE-RAG. Its original complexity-based routing policy was preserved, while the retrieval backend was replaced with the identical BM25+BGE+RRF pipeline used in our system. All methods share the same retrieval corpus, reranker, generator, and evaluation protocol.

5 Main Results

Table 3. Retrieval and End-to-End QA Performance on NQ-Open.

Method	Exp. Rate	Retrieval Metrics			QA Metrics	
		Hit@1	Hit@5	MRR	EM	F1
BM25 Retrieval	0%	$7.6_{\pm 0.3}$	$13.0_{\pm 0.5}$	$9.4_{\pm 0.2}$	6.8	9.3
Dense Retrieval (BGE)	0%	$8.4_{\pm 0.2}$	$15.0_{\pm 0.6}$	$10.8_{\pm 0.3}$	8.4	10.6
Hybrid Retrieval	0%	$9.3_{\pm 0.2}$	$16.1_{\pm 0.4}$	$11.8_{\pm 0.3}$	8.4	11.9
Hybrid + HyDE	100%	$12.5_{\pm 0.4}$	$22.8_{\pm 0.7}$	$16.6_{\pm 0.4}$	10.1	14.7
Hybrid + Uniform Exp.	100%	$13.2_{\pm 0.4}$	$24.0_{\pm 0.7}$	$17.8_{\pm 0.5}$	11.1	16.3
DARE-RAG	19.7%	$12.9_{\pm 0.3}$	$21.4_{\pm 0.6}$	$16.9_{\pm 0.4}$	10.4	15.2

Table 4. Retrieval and End-to-End QA Performance on HotpotQA.

Method	Exp. Rate	Retrieval Metrics			QA Metrics	
		Hit@1	Hit@5	MRR	EM	F1
BM25 Retrieval	0%	$4.8_{\pm 0.4}$	$9.1_{\pm 0.7}$	$6.2_{\pm 0.3}$	4.1	6.8
Dense Retrieval (BGE)	0%	$6.0_{\pm 0.5}$	$11.9_{\pm 0.8}$	$8.1_{\pm 0.4}$	5.4	8.5
Hybrid Retrieval	0%	$6.7_{\pm 0.4}$	$13.2_{\pm 0.7}$	$9.0_{\pm 0.4}$	6.0	9.7
Hybrid + HyDE	100%	$9.4_{\pm 0.6}$	$17.8_{\pm 1.0}$	$12.8_{\pm 0.6}$	8.0	12.5
Hybrid + Uniform Exp.	100%	$10.8_{\pm 0.5}$	$20.4_{\pm 0.9}$	$14.7_{\pm 0.5}$	9.3	14.4
DARE-RAG	32.6%	$9.7_{\pm 0.5}$	$17.9_{\pm 0.8}$	$13.1_{\pm 0.5}$	8.4	12.8

Tables 3 and 4 report retrieval and end-to-end QA performance on NQ-Open and HotpotQA, respectively. On NQ-Open, DARE-RAG consistently improves retrieval quality over non-expansion baselines while activating query expansion for only 19.7% of queries. Compared with Hybrid Retrieval without expansion, DARE-RAG improves Hit@5 from 16.1% to 21.4% and increases F1 from 11.9% to 15.2%, while requiring substantially fewer query expansions than uniform expansion strategies.

On the more challenging HotpotQA benchmark, retrieval and QA scores are lower across all methods due to the higher compositional complexity and evidence dependency of multi-hop queries. Nevertheless, DARE-RAG maintains competitive performance under zero-shot transfer, improving Hit@5 from 13.2% to 17.9% while activating expansion for only 32.6% of queries.

Although uniform expansion achieves the highest retrieval performance, DARE-RAG preserves a large portion of the retrieval gains while substantially reducing the expansion rate. These results demonstrate the effectiveness of adaptive query expansion in balancing retrieval effectiveness and computational efficiency.

6 Efficiency–Effectiveness Trade-off

Table 5. Efficiency–Effectiveness Trade-off on NQ-Open

Method	Expansion Rate	Hit@5	Avg Latency (ms)	Relative Cost
Hybrid RAG (No Expansion)	0%	16.1%	188	1.0×
Hybrid + HyDE	100%	22.8%	1186	6.3×
Hybrid + Uniform Expansion	100%	**24.0%**	964	5.1×
Adaptive-RAG	46.8%	22.6%	694	3.7×
DARE-RAG	19.7%	21.4%	342	1.8×

Table 6. Efficiency–Effectiveness Trade-off on HotpotQA

Method	Expansion Rate	Hit@5	Avg Latency (ms)	Relative Cost
Hybrid RAG (No Expansion)	0%	13.2%	195	1.0×
Hybrid + HyDE	100%	17.8%	1210	6.2×
Hybrid + Uniform Expansion	100%	**20.4%**	985	5.1×
Adaptive-RAG	55.2%	19.4%	811	4.2×
DARE-RAG	32.6%	17.9%	448	2.3×

Tables 5 and 6 demonstrate that expansion utility is highly concentrated among a relatively small subset of retrieval-challenging queries. As query complexity increases in HotpotQA, the expansion activation rate rises from 19.7% to 32.6%, indicating that retrieval uncertainty provides a meaningful proxy for adaptive expansion triggering. This behavior is consistent with the higher compositional complexity and ambiguity of multi-hop retrieval scenarios. Despite using substantially fewer expansions than uniform expansion, DARE-RAG preserves most retrieval gains while significantly reducing inference cost.

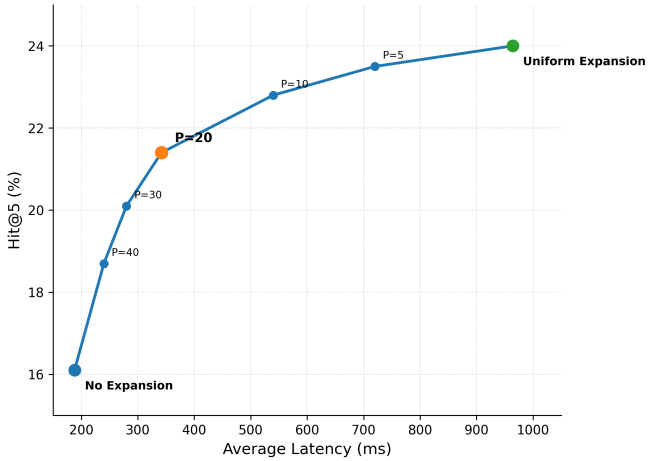

Fig. 3. Efficiency–effectiveness trade-off between retrieval quality and average latency under different expansion thresholds.

Figure 3 illustrates the trade-off between retrieval effectiveness and computational cost under different expansion trigger thresholds. Lower percentile thresholds activate query expansion more aggressively, improving retrieval quality at the cost of notably higher latency.

As the threshold increases, expansion becomes more selective, reducing computational overhead while sacrificing part of the retrieval gains. DARE-RAG

with $P = 20$ achieves a favorable balance between effectiveness and efficiency, recovering most of the benefits of uniform expansion while incurring significantly lower inference cost. Validation analysis further shows that expansion gains are highly concentrated within a relatively small subset of retrieval-challenging queries, motivating a low expansion budget.

7 Latency Analysis

Table 7. Module-wise Latency Breakdown

Module	Mean (ms)	P95 (ms)	Trigger Rate
Confidence Probe	1.8	2.4	100%
LLM Expansion	782	1053	19.7%
BM25 Retrieval	14.2	19.8	100%
Dense Retrieval	8.3	11.1	100%
RRF Fusion	0.9	1.3	100%
Cross-Encoder Reranking	44.6	58.2	100%
Generation	118	187	100%

Table 7 shows that the confidence probe introduces negligible additional overhead ($\sim$1.8 ms per query) compared with the overall retrieval and generation pipeline. The dominant latency source remains LLM-based query expansion, motivating the need for selective expansion triggering. By activating expansion for only 19.7% of queries, DARE-RAG substantially reduces amortized inference cost while preserving most retrieval gains.

8 Ablation Study

8.1 Expansion Trigger Strategy Comparison

Table 8. Comparison of Expansion Trigger Strategies

Trigger Strategy	Expansion Rate	Hit@5	Avg Latency (ms)
Random Trigger	20.0%	18.7%	334
Query Length Threshold	20.1%	19.2%	336
Entropy Threshold	19.9%	20.1%	339
Top-1/Top-2 Margin Threshold	20.0%	20.5%	340
DARE-RAG (MLP Gate)	19.7%	**21.4%**	342

Table 8 compares the proposed MLP-Based Expansion Gate against several heuristic expansion triggering strategies under the same expansion budget.

The proposed MLP-based expansion gate consistently outperforms heuristic triggering strategies, suggesting that jointly modeling multiple retrieval signals yields more accurate expansion utility estimation than relying on individual retrieval features.

Table 9. Ablation Study on DARE-RAG Components

Configuration	Hit@5	EM	Avg Latency (ms)	Expansion Rate
DARE-RAG (Full)	**21.4%**	**10.4%**	342	19.7%
Uniform Expansion (No Gate)	24.0%	11.1%	964	100%
No Expansion	16.1%	8.4%	188	0%
w/o Redundancy Filter	20.2%	10.2%	389	19.7%
w/o Lexical Specificity Feature	20.4%	10.0%	335	20.1%
w/o Cross-Encoder	18.7%	10.1%	274	19.7%

Table 9 reports the contribution of each major component in DARE-RAG. Removing the adaptive gate improves retrieval quality but incurs a substantial latency increase due to uniform expansion. Removing the redundancy filter increases latency by +15.8% and slightly reduces Hit@5, suggesting that filtering near-duplicate variants improves expansion efficiency. The lexical specificity feature (ρ_s) provides modest but consistent improvements in retrieval performance, suggesting that lexical priors complement distributional uncertainty signals in retrieval difficulty estimation.

8.2 Qualitative Retrieval Transition Analysis

To better understand the behavior of selective expansion, we analyze representative retrieval transitions sampled from the held-out evaluation subset. Table 10 presents both successful and unsuccessful retrieval scenarios.

We observe that adaptive expansion is particularly effective when the original query lacks explicit lexical grounding or entity specificity. In contrast, failure cases are often associated with semantic over-specialization or ambiguity amplification introduced during expansion.

8.3 Error Analysis

To better understand the limitations of DARE-RAG, we manually analyze 100 randomly sampled failure cases from the NQ-Open test split.Error categories were manually assigned by the authors based on the primary cause of retrieval failure, and each example was assigned to one dominant category.Table 11 summarizes the dominant error categories.

Table 10. Representative retrieval transitions observed under adaptive expansion.

Query	Expansion Output	Rank Change	Observation
who developed general relativity	einstein theory of relativity	$17 \rightarrow 2$	Expansion introduced explicit entity grounding, improving lexical alignment with relevant Wikipedia passages and substantially improving retrieval ranking quality.
apple release date	apple iphone launch	$6 \rightarrow 18$	Expansion over-specialized the query toward consumer electronics, suppressing passages related to the intended entity and reducing retrieval precision.

Table 11. Failure mode analysis of DARE-RAG on NQ-Open.

Error Type	Ratio	Representative Cause
Entity-free / underspecified queries	38%	Insufficient lexical grounding
Semantic drift after expansion	29%	Ambiguous entity interpretation
Rare lexical mismatch	21%	Missing exact lexical evidence
Multi-hop reasoning failure	12%	Incomplete evidence composition

Entity-free and underspecified queries remain challenging because expansion often fails to introduce sufficiently discriminative retrieval cues. Semantic drift is the second major failure mode, occurring when expansion introduces irrelevant concepts for ambiguous entities.

Dense retrieval also occasionally overemphasizes semantic similarity while missing rare lexical evidence, especially for long-tail entities. Multi-hop failures are mainly caused by incomplete evidence aggregation across passages.

Future work may explore entity-aware expansion and improved multi-passage evidence organization.

9 Conclusion

We propose DARE-RAG, a difficulty-aware adaptive retrieval framework that selectively performs query expansion for Retrieval-Augmented Generation. By estimating expansion utility from lightweight probe retrieval signals, DARE-RAG activates LLM-based expansion only for queries likely to benefit from additional retrieval coverage.

Experiments on NQ-Open and HotpotQA demonstrate that DARE-RAG consistently improves retrieval and downstream QA performance over non-expansion

baselines while preserving a large portion of the gains obtained by uniform expansion at substantially lower expansion rates and inference cost.

Furthermore, DARE-RAG is complementary to generation-time adaptive retrieval methods and can be integrated with iterative retrieval-generation frameworks.

Future work will explore entity-aware expansion, dynamic retrieval fusion, and reinforcement learning-based retrieval policies to further improve expansion utility estimation and retrieval efficiency.

References

1. Asai, A., Wu, Z., Wang, Y., Sil, A., Hajishirzi, H.: Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection. In: The Twelfth International Conference on Learning Representations (ICLR) (2024)
2. Azad, H.K., Deepak, A.: A Survey on Query Expansion Techniques for Information Retrieval. Information Processing & Management 60(2), 103189 (2023)
3. Carmel, D., Yom-Tov, E.:Estimating the Query Difficulty for Information Retrieval.In: Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 811–812 (2010)
4. Chen, J., Xiao, S., Zhang, P., Luo, K., Lian, D., Liu, Z.: M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation. In: Findings of the Association for Computational Linguistics: ACL 2024, pp. 15889–15907 (2024)
5. Cormack, G.V., Clarke, C.L., Buettcher, S.: Reciprocal Rank Fusion outperforms Condorcet and individual Rank Learning Methods. In: Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 783–784 (2009)
6. Gao, L., Ma, X., Lin, J., Callan, J.: Precise Zero-Shot Dense Retrieval without Relevance Labels. arXiv preprint arXiv:2212.10496 (2022)
7. Gao, T., Yao, X., Chen, D.: SimCSE: Simple Contrastive Learning of Sentence Embeddings. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 6894–6910 (2021)
8. Hauff, C., Murdock, V., Baeza-Yates, R.: Query Difficulty Estimation: A Comprehensive Survey. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 2801–2805 (2022)
9. Izacard, G., Shi, B., Zhou, S., et al.: Atlas: Few-shot Learning with Retrieval Augmented Language Models. Journal of Machine Learning Research (JMLR) 24(251), 1–43 (2023)
10. Jeong, S., Baek, J., Kim, S., et al.: 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 (NAACL) (2024)
11. Jiang, Z., Xu, F.F., Araki, J., Neubig, G.: FLARE: Iterative Retrieval-Augmented Generation via Active Reasoning. arXiv preprint arXiv:2305.06983 (2023)
12. Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., Yih, W.T.: Dense Passage Retrieval for Open-Domain Question Answering. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 6769–6781 (2020)

13. Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., et al.: Natural Questions: A Benchmark for Question Answering Research. Transactions of the Association for Computational Linguistics (TACL) 7, 453–466 (2019)
14. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., et al.: Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In: Advances in Neural Information Processing Systems (NeurIPS) (2020)
15. Liu, N.F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., Liang, P.: Lost in the Middle: How Context Length Impacts Language Models. arXiv preprint arXiv:2307.03172 (2023)
16. Ma, X., Lin, J., et al.: Multi-Stage Document Ranking with BERT. In: European Conference on Information Retrieval (ECIR), pp. 608–621 (2021)
17. Nogueira, R., Cho, K.: Passage Re-ranking with BERT. arXiv preprint arXiv:1901.04085 (2019)
18. Sharifymoghaddam, S., et al.: RankLLM: Open Source Pipeline for LLM-based Ranking. In: Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (2025)
19. Xu, F., Chen, Y., Wang, Y., Hajishirzi, H., Iyyer, M.: RECOMP: Improving Retrieval-Augmented LMs with Compressors and Selective Augmentation. In: Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL), pp. 15889–15907 (2024)
20. Yang, Z., Qi, P., Zhang, S., Zhang, B., Chen, R., Lu, Y., ... & Manning, C.D.: HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 2369–2380 (2018)

A Query Expansion Prompt and Generation Protocol

The expansion process uses the following structured system prompt and strict decoding constraints:

> **[System]** You are an expert query expansion assistant for information retrieval. Your task is to generate 2-3 semantically equivalent but lexically diverse variants of the given user query. Focus on: 1. Lexical Substitution: Replace key terms with precise synonyms or related domain terms. 2. Hierarchical Decomposition: Break down complex or underspecified queries into focused sub-queries that capture implicit constraints (e.g., time, location, entity type). Output ONLY a JSON array of strings. Do not add explanations.
>
> **[User]** Original Query: query

Decoding and Filtering Rules Temperature $T = 0.3$, Top-$p = 0.9$. Generated variants are filtered by cosine similarity against q; those with cosine ≥ 0.92 are discarded. Maximum $m = 3$ variants are retained for retrieval.

B Threshold Sensitivity Analysis

We swept $\tau \in \{0.05, 0.1, 0.2, 0.3, 0.5\}$ on the validation set. $\tau = 0.2$ yields the optimal trade-off: higher values under-trigger on moderately retrieval-challenging queries, while lower values admit noisy expansions that degrade precision.