ThinkQE: Query Expansion via an Evolving Thinking Process

Yibin Lei[1], **Tao Shen**[2], **Andrew Yates**[3]
[1]University of Amsterdam [2]University of Technology Sydney
[3]Johns Hopkins University, HLTCOE
y.lei@uva.nl, tao.shen@uts.edu.au, andrew.yates@jhu.edu

Abstract

Effective query expansion for web search benefits from promoting both exploration and result diversity to capture multiple interpretations and facets of a query. While recent LLM-based methods have improved retrieval performance and demonstrate strong domain generalization without additional training, they often generate narrowly focused expansions that overlook these desiderata. We propose ThinkQE, a test-time query expansion framework addressing this limitation through two key components: a thinking-based expansion process that encourages deeper and comprehensive semantic exploration, and a corpus-interaction strategy that iteratively refines expansions using retrieval feedback from the corpus. Experiments on diverse web search benchmarks (DL19, DL20, and BRIGHT) show ThinkQE consistently outperforms prior approaches, including training-intensive dense retrievers and rerankers.[1]

1 Introduction

Query expansion (QE) is a common practice in web search scenarios (Robertson, 1990; Qiu and Frei, 1993), particularly for first-stage retrievers such as BM25 (Robertson et al., 1995). Effective expansion involves not only reinforcing the core intent of the query but also introducing terms that capture different facets or interpretations of the information need. This multifaceted coverage helps capture a broader semantic context, enabling the retrieval of a more comprehensive set of relevant documents. Prior studies have shown that such broad-coverage expansion strategies lead to substantial improvements in retrieval quality (Bouchoucha et al., 2013).

Recent advances in large language models (LLMs) have led to strong performance in query expansion (Gao et al., 2022; Wang et al., 2023; Jagerman et al., 2023; Mackie et al., 2023; Shen et al., 2024; Lei et al., 2024), particularly due to their ability to rapidly adapt to new domains without requiring additional training. However, existing LLM-based methods often pay limited attention to exploration and result diversity. As illustrated in Table 1, we observe that current approaches, such as HyDE, tend to generate overly confident expansions that focus narrowly on a single interpretation of the input query. This behavior can be attributed to the model's reliance on its internal knowledge and high-probability completions (Yona et al., 2024; Ohi et al., 2024; Sun et al., 2025), which may suppress alternative formulations or less common aspects of the query. This lack of breadth limits the retrieval of documents reflecting alternative scenarios or requiring more nuanced reasoning.

Query: Who is robert gray
Expansion w/o. Thinking: Robert Gray is best known as the American captain who discovered the Columbia River in 1792. He named the river after his ship, the Columbia Rediviva, and explored it up to Grays Bay. His discovery was later documented by Lieutenant William Broughton during the Vancouver expedition.
ThinkQE: Robert Gray is best known as Captain Robert Gray, an American explorer who played a significant role in the exploration of the Pacific Northwest. In 1792, he captained the ship Columbia Rediviva and became the first American to navigate the Columbia River, which he named after his vessel. On May 11, 1792, he entered the mouth of the river and explored approximately 20 miles upstream as far as Grays Bay, which was later named in his honor by Lieutenant William Broughton of the Vancouver expedition. This expedition contributed to the mapping and understanding of the region, highlighting Gray's importance in early American exploration.

Table 1: Examples comparing a standard expansion with *ThinkQE*, our proposed query expansion method with thinking-augmentation. ThinkQE encourages deeper reasoning and multifaceted contextualization.

To address these limitations, we propose ThinkQE, a new framework that improves exploration and result diversity along two complementary dimensions. First, we introduce a *thinking-based expansion process*, where the model ex-

[1]Our code is publicly available at https://github.com/Yibin-Lei/Think_QE.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 17772–17781
November 4-9, 2025

plicitly accumulates intermediate thoughts and hypotheses before producing final expansions. This encourages the emergence of new and more exploratory terms that can help retrieve documents beyond the initial query scope. Second, inspired by pseudo-relevance feedback (Rocchio Jr, 1971), we propose an *interactive expansion strategy*, where query expansions are progressively refined using feedback from the documents retrieved at each stage. This dynamic interaction with the corpus allows the query to evolve in a context-aware manner, adapting to newly retrieved evidence.

By combining both components, we develop ThinkQE, a test-time query expansion method that achieves strong performance on natural language web search benchmarks, including DL19, DL20, and the StackExchange domain of the BRIGHT benchmark[2]. Remarkably, ThinkQE requires no additional training, yet surpasses recent training-intensive reranking methods, including those based on reinforcement learning and distillation from DeepSeek-R1. Our analysis reveals that: (1) explicitly modeling a thinking process enhances expansion quality, and (2) iteratively refining queries with evolving retrieval feedback is more effective than generating static expansions, even under the same compute budget.

2 Method

We introduce ThinkQE, a query expansion framework that tightly integrates LLM-based thinking process with evolving corpus interaction. ThinkQE follows prior work in generating query expansions using retrieved documents but distinguishes itself through its design of thinking augmentation and iterative corpus interactions. The method is designed to enable exploration of the query space through thinking processes and evolving refinement based on retrieval feedback from the corpus. The overall process proceeds in multiple rounds. At each round, an LLM performs thinking-augmented expansion based on the original query and newly retrieved documents from the corpus, which in turn informs subsequent retrieval and expansion steps. The following subsections describe each component of the method in detail.

[2]We omit math and coding subsets, as ThinkQE relies on natural language expansions, which may not be well-suited for symbolic or structured domains.

ThinkQE Prompt
Given a question "{q}" and its possible answering passages (most of these passages are wrong) enumerated as: 1. {d_1}; 2. {d_2}; 3. {d_3} ... please write a correct answering passage. Use your own knowledge, not just the example passages!

Table 2: Prompt used in ThinkQE for the thinking-based expansion process. {$\cdot$} denotes the placeholder for the corresponding query and top-K documents.

2.1 Retrieving Initial Evidence from Corpus

Let q_0 denote the original user query. To ground the expansion process in corpus evidence, we begin by retrieving an initial set of documents from the corpus $\mathcal{C}$ using a first-stage lexical retriever. In our implementation, we employ BM25. Specifically, we retrieve the top-K documents: $\mathcal{D}_0 = \text{TopK}(\text{BM25}(q_0, \mathcal{C}))$.

Here, $\mathcal{D}_0$ denotes the ranked list of top-K documents retrieved for q_0, ordered by their BM25 relevance scores. This list serves as the initial feedback signal for expansion, providing retrieval-grounded context to the LLM in the first expansion step.

2.2 Expansion via Thinking Process

To produce an initial expansion, we use an R1-distilled LLM trained to generate a thinking chain before answering. Given the original query q_0 and top-K retrieved documents $\mathcal{D}_0$, the model follows a two-phase process:

1. **Thinking Phase:** The model reflects on q_0 and $\mathcal{D}_0$ to identify latent concepts, resolve ambiguities, and surface alternative interpretations or missing aspects of the information need.

2. **Expansion Phase:** Based on the thinking output, the model generates a query expansion segment e_1 that builds upon the original query by introducing additional relevant terms and concepts.

Leveraging the R1-distilled model's natural separation of thought and answer allows us to implement the reasoning-expansion workflow without additional scaffolding or prompt engineering. The prompt shown in Table 2 guides the model to generate expansions by thinking over the input query and the top-retrieved documents.

2.3 Evolution via Corpus Feedback

We propose to iterate the above thinking-based expansion. At each round $t = 1, \ldots, T$, the method performs the following steps:

1. **Retrieval:** The current query q_t is used to retrieve a ranked list of documents from the corpus: $\mathcal{R}_t = \text{BM25}(q_t, \mathcal{C})$.

2. **Redundancy Filtering:** To promote diversity and avoid repetition, we exclude documents that (a) appear in the blocklist $\mathcal{B}_t$, or (b) were among the top-K results in the previous round $\mathcal{D}_{t-1}$. We then select the top-K documents from the remaining candidates: $\mathcal{D}_t^{\text{new}} = \text{TopK}(\mathcal{R}_t \setminus (\mathcal{B}_t \cup \mathcal{D}_{t-1}))$. The blocklist is updated to include all documents that were filtered out in this round.

3. **Expansion via Thinking:** The LLM is prompted with the original query q_0 and the filtered document set $\mathcal{D}_t^{\text{new}}$ to generate the next expansion e_{t+1}, using the same two-phase expansion process described in Section 2.2.

4. **Query Update:** The query is iteratively updated by concatenating the new expansion: $q_{t+1} = q_t \oplus e_{t+1}$.

This loop can be repeated for any number of rounds T, depending on resource constraints or desired depth. Notably, as the query grows longer, successive expansions may dilute or replace the original intent. To mitigate this, we follow Zhang et al. (2024) and repeat the original query n times in the final reformulation, with $n = \frac{\text{len(expansions)}}{\text{len}(q_0)\times\lambda}, \quad \lambda = 3$. Here, len(expansions) refers to the total word count of all expansion segments, and len(q_0) is the word count of the original query. This repetition reinforces the core semantics of the original query during iterative refinement.

Remark. Our method introduces two key innovations: (1) the use of an explicit, LLM-guided thinking process to encourage deeper exploration during expansion, and (2) an evolving loop that dynamically refines the query based on retrieval feedback. Within this evolving process, we design two essential components – *redundancy filtering* and *expansion accumulation* – both of which play a critical role in the effectiveness of ThinkQE, as demonstrated in our results in Section 4.3.

3 Experiments

3.1 Setup

Datasets. We evaluate ThinkQE on two categories of natural language web search datasets: (1) **Factoid-style retrieval:** TREC DL19 (Craswell et al., 2020) and DL20 (Craswell et al., 2021), widely used benchmarks based on the MS MARCO document collections (Bajaj et al., 2016); and (2) **Reasoning-oriented datasets:** The StackExchange domain of the BRIGHT benchmark (Su et al., 2025), covering seven diverse sub-domains: Biology (Bio.), Earth Science (Earth.), Economics (Econ.), Psychology (Psy.), Robotics (Rob.), Stack Overflow (Stack.), and Sustainable Living (Sus.). On the BRIGHT benchmark, we omit math and coding datasets to focus on only the StackExchange subsets, as ThinkQE relies on language-model-based natural language expansions, which may not be well-suited for symbolic or structured domains such as code or math.

	DL19			DL20		
	mAP	ndcg@10	R@1k	mAP	ndcg@10	R@1k
BM25	30.1	50.6	75.0	28.6	48.0	78.6
Supervised Fine-Tuned Dense retrievers						
DPR	36.5	62.2	76.9	41.8	**65.3**	81.4
ANCE	37.1	64.5	75.5	40.8	64.6	77.6
Contriever$^{\text{FT}}$	41.7	62.1	83.6	43.6	63.2	85.8
R1-Distilled Rerankers on BM25 Top-20 Docs						
Rank1-32B	-	64.9	-	-	61.2	-
Rank-K-32B	-	66.2	-	-	64.3	-
Zero-shot Query expansions with BM25						
HyDE	41.8	61.3	88.0	38.2	57.9	84.4
Query2doc	-	66.2	-	-	62.9	-
MILL	-	63.8	85.9	-	61.8	85.3
LameR	42.8	64.9	84.2	-	-	-
CSQE	43.6	63.4	87.6	-	-	-
ThinkQE (*ours*)						
w. R1-14B	**45.9**	**68.8**	**89.3**	**43.9**	64.7	87.8
w. QWEN3-8B	44.5	65.0	87.9	41.9	62.8	88.0
w. QWEN3-14B	45.2	64.9	88.4	42.4	63.5	**88.4**
w. OpenThinker2-7B	44.8	65.3	87.3	43.2	63.5	87.9
w. Phi4-Reasoning-14B	44.0	65.0	87.1	43.0	63.9	87.2

Table 3: Results on TREC DL19 and DL20 datasets. In-domain supervised models DPR, ANCE, and Contriever$^{\text{FT}}$ are trained on the MS-MARCO dataset and listed for reference. **Bold** indicates the best result across all models.

Implementation. We use the QWEN-R1-Distill-14B model (DeepSeek-AI, 2025) to generate thinking-based query expansions, sampling outputs with a temperature of 0.7. The BM25 retrieval is performed using Pyserini (Lin et al., 2021) with default hyperparameters. At each round, ThinkQE uses the top-5 retrieved documents (truncated to 128 tokens for DL benchmarks and 512 tokens for BRIGHT) to prompt the LLM, and samples 2 candidate expansions to enhance diversity. We set the total number of interaction rounds to 3, for a balance between efficiency and effectiveness. Besides the QWEN-R1-Distill-14B model, we also evalute ThinkQE on a variety of reasoning models, including QWEN3-8B, QWEN3-14B (Yang et al., 2025a), OpenThinker2-7B (Meta, 2023), and Phi-4-Reasoning-14B (Abdin et al., 2025).

Baselines. On DL19 and DL20, we compare ThinkQE to recent SOTA zero-shot query ex-

	Training	Bio.	Earth.	Econ.	Psy.	Rob.	Stack.	Sus.	Avg.
BM25	Zero-shot	18.2	27.9	16.4	13.4	10.9	16.3	16.1	17.0
BM25 + GPT-4o CoT	Zero-shot	**53.6**	53.6	24.3	38.6	18.8	22.7	25.9	33.9
LLM-based dense retrievers									
GritLM-7B	SFT	24.8	32.3	18.9	19.8	17.1	13.6	17.8	20.6
GTE-QWEN-7B	SFT	30.6	36.4	17.8	24.6	13.2	22.2	14.8	22.8
ReasonIR-8B	SFT	26.2	31.4	23.3	30.0	18.0	23.9	20.5	24.8
Rerankers on BM25 Top-100 docs									
RankGPT4	Zeroshot	33.8	34.2	16.7	27.0	22.3	27.7	11.1	24.7
RankZephyr-7b	GPT4-distill	21.9	23.7	14.4	10.3	7.6	13.7	16.6	15.5
Rank-R1-14B	GRPO (RL)	31.2	38.5	21.2	26.4	22.6	18.9	27.5	26.6
Rerankers on BM25+GPT-4o CoT Top-100 docs									
Rank1-14B	R1-distill	49.3	37.7	22.6	35.2	22.5	20.8	**33.6**	31.7
Rank-K-32B*	R1-distill	50.8	49.4	28.2	**46.0**	**27.3**	**30.5**	31.9	**37.9**
Query expansion with BM25									
HyDE-R1-14B	Zero-shot	33.3	44.9	21.1	29.8	16.3	24.1	21.0	27.2
LameR-R1-14B	Zero-shot	35.1	46.1	23.7	31.0	17.7	26.4	25.3	29.3
ThinkQE (*ours*)									
R1-14B	Zero-shot	47.3	52.5	29.2	40.0	19.3	28.0	27.9	34.9
QWEN3-8B	Zero-shot	49.8	**55.3**	27.6	36.7	19.9	29.0	28.3	35.2
QWEN3-14B	Zero-shot	51.5	53.2	27.8	37.2	22.0	16.1	27.5	33.6
OpenThinker2-7B	Zero-shot	50.5	54.1	25.8	36.7	18.1	28.2	28.9	34.6
Phi-4-Reasoning-14B	Zero-shot	51.8	53.5	**29.7**	38.5	21.8	29.3	27.7	36.0

Table 4: Results on the StackExchange domain of the BRIGHT benchmark in terms of nDCG@10. The best and the second best results across all models are in **bold** and underlined font, respectively. All models are performed on the original query. BM25+GPT-4o-CoT refers to using BM25 retrieval results on queries rewritten by GPT-4o with chain-of-thought reasoning traces for reranking. *Rank-K-32B performs computationally expensive listwise reranking over the top-20 documents.

pansion methods including HyDE (Gao et al., 2022), Query2doc (Wang et al., 2023), MILL (Jia et al., 2024), LameR (Shen et al., 2024) and CSQE (Lei et al., 2024), which use strong LLMs like text-davinci-003-175B (Ouyang et al., 2022), GPT-3.5-turbo, LLaMA2-13B-Chat and LLaMA2-70B-Chat (Meta, 2023). We also include two recent rerankers distilled from DeepSeek-R1-685B (DeepSeek-AI, 2025) thinking traces for comparison: Rank1-32B (Weller et al., 2025) and Rank-K-32B (Yang et al., 2025b). For reference, we also report results from supervised dense retrievers trained on MS MARCO: DPR (Karpukhin et al., 2020), ANCE (Xiong et al., 2021), and Contriever^{FT} (Izacard et al., 2022).

On the BRIGHT benchmark, we consider three categories of baselines: (1) **LLM-based embedding models** such as GritLM-7B (Muennighoff et al., 2025), GTE-Qwen-7B (Li et al., 2023), and ReasonIR-8B (Shao et al., 2025), all trained on massive amounts of retrieval data; (2) **LLM-based rerankers**, including RankGPT4 (zero-shot) (Sun et al., 2023), RankZephyr-7B (distilled from GPT-4) (Pradeep et al., 2023), Rank1-14B (distilled from DeepSeek-R1-685B) (Weller et al., 2025), Rank-R1-14B (trained via reinforcement learning) (Zhuang et al., 2025), and Rank-K-32B (distilled from DeepSeek-R1-685B) (Yang et al., 2025b). Rank1-14B, Rank-R1-14B, and Rank-K-32B explicitly incorporate a thinking process during reranking; and (3) **Query expansion methods** such as HyDE and LameR, which use the same underlying model as ThinkQE but do not incorporate any explicit thinking process.[3] Our method ThinkQE is evaluated in a zero-shot configuration across all datasets.

3.2 Main Results

Results are presented in Tables 3 and 4. On DL19 and DL20, ThinkQE outperforms almost all other zero-shot query expansion methods across different underlying models, achieving the highest scores across all metrics. Notably, it performs competitively with supervised dense retrievers such as Contriever^{FT}, despite requiring no additional training. Furthermore, uisng the QWEN-R1-Distill-14B model, ThinkQE surpasses R1-distilled reranking models such as Rank1-32B and Rank-K-32B – which also leverage a thinking process and are significantly more computationally expensive.

On the BRIGHT benchmark, ThinkQE remains the strongest among zero-shot query expansion methods, achieving an average nDCG@10 of 36.0 using the Phi-4-Reasoning-14B model. While Rank-K-32B achieves the highest overall score (37.9), it relies on R1 distillation and listwise reranking over GPT-4o-augmented retrieval results, making it significantly more resource-intensive. In contrast, ThinkQE operates in a fully training-free setting and still outperforms several more expensive rerankers, including RankGPT4 (24.7) and Rank1-14B (31.7). Beyond its efficiency, ThinkQE

[3]We provide a detailed analysis of the no-thinking setting for fair comparison with ThinkQE in Section 4.1.

Model	BRIGHT Avg.
QWEN-14B	27.6
QWEN-R1-14B *w/o. thinking*	29.8
QWEN-R1-14B *w. thinking*	32.5

Table 5: Impact on the thinking process. We compare three configurations: the base model QWEN-14B without any thinking involved, a *NoThinking* variant that bypasses actual thinking by prefilling a dummy thinking trace, and the ThinkQE model with the thinking process enabled.

delivers consistently strong performance across domains, ranking first or second in four sub-domains.

4 Analysis

In this section, we conduct a detailed analysis of ThinkQE on the StackExchange domain of the BRIGHT benchmark.

4.1 Impact of the Thinking Process

To evaluate the impact of the thinking process, we conduct two ablation studies on ThinkQE: (1) replacing the used model with its base version, QWEN-14B-Base, which has not been trained to produce reasoning traces, and (2) applying the *NoThinking* (Ma et al., 2025) method, where we prefill the response with a fabricated thinking block (i.e., *<think>Okay, I think I have finished thinking.</think>*) and allow the model to generate the answer directly from that point. As shown in Table 5, ThinkQE with thinking significantly outperforms both variants, underscoring the importance of generating thinking output. We use the *NoThinking* variant as the main baseline.

4.2 Impact of Corpus Interaction

To evaluate the corpus interaction process, we compare ThinkQE to a baseline that performs all LLM expansions in a single round – referred to as parallel scaling. In contrast, ThinkQE uses corpus-interaction scaling, distributing expansions across multiple rounds with retrieval feedback. As shown in Figure 1, this interaction strategy consistently outperforms the static baseline, indicating that iterative refinement with evolving context is more effective than isolated expansions.

4.3 Impact on Expansion Accumulation and Redundancy Filter Mechanisms

We conduct a final ablation study on the two core components of the interaction process in ThinkQE:

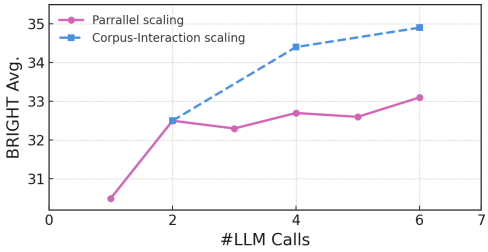

Figure 1: Impact of evolving corpus interaction process.

expansion accumulation, where query expansions from different rounds are concatenated to form the new query, and the semantic filter, which excludes top-retrieved documents from the previous round to encourage the introduction of new information. As shown in Table 6, both components are essential for maximizing performance. Disabling either mechanism leads to a noticeable performance drop, highlighting their complementary roles in refining the query and diversifying retrieved evidence across rounds.

Accum.	Filter	BRIGHT Avg.
✓	✗	34.2
✗	✓	33.4
✓	✓	34.9

Table 6: Impact of the expansion accumulation and redundancy filtering mechanisms.

5 Conclusion

We presented ThinkQE, a test-time query expansion method enhancing exploration and diversity through a thinking-based expansion process and evolving interactions with the corpus. Without requiring any training, ThinkQE consistently improves retrieval performance across multiple benchmarks by encouraging deeper coverage and adaptive refinement, offering a lightweight yet effective alternative to training-based dense retrievers and rerankers.

Limitations

The thinking process and evolving interaction process introduce higher inference-time latency and computational cost compared to single-shot expansion methods, which may limit its practicality in latency-sensitive or large-scale deployment scenarios. Furthermore, since our experiments focus exclusively on English web search tasks, the effectiveness of ThinkQE in multilingual settings remains unexplored.

Acknowledgments

This research was supported by project VI.Vidi.223.166 of the NWO Talent Programme which is (partly) financed by the Dutch Research Council NWO). We acknowledge the Dutch Research Council for awarding this project access to the LUMI supercomputer, owned by the EuroHPC Joint Undertaking, hosted by CSC (Finland) and the LUMI consortium through project number NWO-2024.050.

References

Marah Abdin, Sahaj Agarwal, Ahmed Awadallah, Vidhisha Balachandran, Harkirat Behl, Lingjiao Chen, Gustavo de Rosa, Suriya Gunasekar, Mojan Javaheripi, Neel Joshi, Piero Kauffmann, Yash Lara, Caio César Teodoro Mendes, Arindam Mitra, Besmira Nushi, Dimitris Papailiopoulos, Olli Saarikivi, Shital Shah, Vaishnavi Shrivastava, and 4 others. 2025. Phi-4-reasoning technical report. *arXiv preprint arXiv:2504.21318*.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, and 1 others. 2016. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*.

Arbi Bouchoucha, Jing He, and Jian-Yun Nie. 2013. Diversified query expansion using conceptnet. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*, CIKM '13, page 1861–1864. Association for Computing Machinery.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2020. Overview of the trec 2019 deep learning track. *arXiv preprint arXiv:2003.07820*.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2021. Overview of the trec 2020 deep learning track. *arXiv preprint arXiv:2102.07662*.

DeepSeek-AI. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2022. Precise zero-shot dense retrieval without relevance labels. *arXiv preprint arXiv:2212.10496*.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*.

Rolf Jagerman, Honglei Zhuang, Zhen Qin, Xuanhui Wang, and Michael Bendersky. 2023. Query expansion by prompting large language models. *arXiv preprint arXiv:2305.03653*.

Pengyue Jia, Yiding Liu, Xiangyu Zhao, Xiaopeng Li, Changying Hao, Shuaiqiang Wang, and Dawei Yin. 2024. MILL: Mutual verification with large language models for zero-shot query expansion. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2498–2518, Mexico City, Mexico. Association for Computational Linguistics.

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 *EMNLP*, pages 6769–6781, Online.

Yibin Lei, Yu Cao, Tianyi Zhou, Tao Shen, and Andrew Yates. 2024. Corpus-steered query expansion with large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 393–401. Association for Computational Linguistics.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.

Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: A python toolkit for reproducible information retrieval research with sparse and dense representations. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 2356–2362, Virtual Event, Canada. Association for Computing Machinery.

Wenjie Ma, Jingxuan He, Charlie Snell, Tyler Griggs, Sewon Min, and Matei Zaharia. 2025. Reasoning models can be effective without thinking. *arXiv preprint arXiv:2504.09858*.

Iain Mackie, Shubham Chatterjee, and Jeffrey Dalton. 2023. Generative relevance feedback with large language models. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, page 2026–2031, Taipei, Taiwan. Association for Computing Machinery.

Meta. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Niklas Muennighoff, Hongjin SU, Liang Wang, Nan Yang, Furu Wei, Tao Yu, Amanpreet Singh, and Douwe Kiela. 2025. Generative representational instruction tuning. In *The Thirteenth International Conference on Learning Representations*.

Masanari Ohi, Masahiro Kaneko, Ryuto Koike, Mengsay Loem, and Naoaki Okazaki. 2024. Likelihood-based mitigation of evaluation bias in large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*,

pages 3237–3245, Bangkok, Thailand. Association for Computational Linguistics.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*.

Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. 2023. Rankzephyr: Effective and robust zero-shot listwise reranking is a breeze! *arXiv preprint arXiv:2312.02724*.

Yonggang Qiu and Hans-Peter Frei. 1993. Concept based query expansion. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '93, page 160–169, Pittsburgh, Pennsylvania. Association for Computing Machinery.

Stephen Robertson. 1990. On term selection for query expansion. *Journal of Documentation*, 46:359–364.

Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, and 1 others. 1995. Okapi at trec-3. *Nist Special Publication Sp*, 109:109.

Joseph John Rocchio Jr. 1971. Relevance feedback in information retrieval. *The SMART retrieval system: experiments in automatic document processing*.

Rulin Shao, Rui Qiao, Varsha Kishore, Niklas Muennighoff, Xi Victoria Lin, Daniela Rus, Bryan Kian Hsiang Low, Sewon Min, Wen tau Yih, Pang Wei Koh, and Luke Zettlemoyer. 2025. Reasonir: Training retrievers for reasoning tasks. *arXiv preprint arXiv:2504.20595*.

Tao Shen, Guodong Long, Xiubo Geng, Chongyang Tao, Yibin Lei, Tianyi Zhou, Michael Blumenstein, and Daxin Jiang. 2024. Retrieval-augmented retrieval: Large language models are strong zero-shot retriever. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 15933–15946, Bangkok, Thailand. Association for Computational Linguistics.

Hongjin Su, Howard Yen, Mengzhou Xia, Weijia Shi, Niklas Muennighoff, Han yu Wang, Liu Haisu, Quan Shi, Zachary S Siegel, Michael Tang, Ruoxi Sun, Jinsung Yoon, Sercan O Arik, Danqi Chen, and Tao Yu. 2025. BRIGHT: A realistic and challenging benchmark for reasoning-intensive retrieval. In *The Thirteenth International Conference on Learning Representations*.

Fengfei Sun, Ningke Li, Kailong Wang, and Lorenz Goette. 2025. Large language models are overconfident and amplify human bias. *arXiv preprint arXiv:2505.02151*.

Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. 2023. Is ChatGPT good at search? investigating large language models as re-ranking agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14918–14937, Singapore. Association for Computational Linguistics.

Liang Wang, Nan Yang, and Furu Wei. 2023. Query2doc: Query expansion with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9414–9423, Singapore. Association for Computational Linguistics.

Orion Weller, Kathryn Ricci, Eugene Yang, Andrew Yates, Dawn Lawrie, and Benjamin Van Durme. 2025. Rank1: Test-time compute for reranking in information retrieval. *arXiv preprint arXiv:2502.18418*.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *ICLR*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025a. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Eugene Yang, Andrew Yates, Kathryn Ricci, Orion Weller, Vivek Chari, Benjamin Van Durme, and Dawn Lawrie. 2025b. Rank-k: Test-time reasoning for listwise reranking. *arXiv preprint arXiv:2505.14432*.

Gal Yona, Roee Aharoni, and Mor Geva. 2024. Can large language models faithfully express their intrinsic uncertainty in words? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7752–7764, Miami, Florida, USA. Association for Computational Linguistics.

Le Zhang, Yihong Wu, Qian Yang, and Jian-Yun Nie. 2024. Exploring the best practices of query expansion with large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1872–1883, Miami, Florida, USA. Association for Computational Linguistics.

Shengyao Zhuang, Xueguang Ma, Bevan Koopman, Jimmy Lin, and Guido Zuccon. 2025. Rank-r1: Enhancing reasoning in llm-based document rerankers via reinforcement learning. *arXiv preprint arXiv:2503.06034*.

A Appendix

A.1 Dataset Statistics

Details about the retrieval datasets are shown in Table 7.

Dataset	#Test	#Corpus
DL19	43	8,841,823
DL20	50	8,841,823
Biology	103	57,359
Earth Science	116	121,249
Economics	103	50,220
Psychology	101	52,835
Robotics	101	61,961
Stack Overflow	117	107,081
Sustainable Living	108	60,792

Table 7: Dataset Statistics.

A.2 Impact of the Thinking Process

Results across all domains on the impact of the thinking process are provided in Table 10.

A.3 Impact of Evolving Corpus Interaction

Results across all domains on the impact of the evolving corpus interaction are provided in Table 11.

A.4 Core Components of the Evolving Interaction Process

Results across all domains on the impact of the expansion accumulation and redundancy filter mechanisms are provided in Table 12.

A.5 Impact of λ

Table 13 presents detailed results analyzing the impact of λ, which influences the repetition frequency of the original query during expansion. The results demonstrate that performance differences are small when varying λ from 3 to 6. However, lower λ values tend to cause excessive repetition of the original query, which generally hurts performance.

A.6 Effectiveness-Efficiency Analysis on ThinkQE

We provide latency and performance trade-off results comparing model size scaling, the thinking process, and multiple rounds in Table 14, using the DeepSeek-R1-Distill-Qwen-14B and DeepSeek-R1-Distill-Qwen-32B models evaluated on a single H100 GPU. Our results show that involving the thinking process and multiple rounds increases latency. However, scaling model size alone yields limited improvements relative to latency increase: moving from R1-14B without thinking (3.71 s/query, 29.8) to R1-32B without thinking (7.88 s/query) improves Bright Avg. by just +0.6 points (30.4) while more than doubling latency. Adding the thinking process, although it increases latency more, is substantially more effective: applying 1 round of thinking to R1-14B boosts performance by +2.7 points (32.5) compared to R1-14B without thinking, with latency rising to 15.40 s/query. Even with thinking enabled, scaling from R1-14B to R1-32B brings only a small additional gain (+0.4 points). Meanwhile, multi-round corpus interaction offers a more efficient path to higher effectiveness, with R1-14B 3-round reaching 34.9 (+2.4 over its 1-round version) and outperforming R1-32B 1-round.

A.7 ThinkQE on Non-Web Search Datasets

We evaluate ThinkQE on two additional non-web search datasets, TREC-Covid and Scifact. The results in Table 8 show that ThinkQE remains highly competitive and often outperforms the baselines. Note that both MILL and LameR are based on the powerful, closed-source GPT-3.5-Turbo model.

Method	TREC-Covid	Scifact
BM25	59.5	67.9
Contriever (FT)	59.6	67.7
HyDE	59.3	69.1
Query2Doc	72.2	68.6
MILL	75.3	71.4
LameR	75.8	73.5
CSQE	74.2	69.6
ThinkQE	76.1	73.3

Table 8: Results (NDCG@10) on non-web search datasets.

A.8 Significance Test Results of ThinkQE

We conduct significance testing by comparing ThinkQE with the two most relevant baselines we reimplemented on BRIGHT: HyDE and LameR using the same QWEN-R1-Distill-14B model. The results are presented in Table 15. Significance tests were performed using a t-test with a p-value threshold of 0.05. The results show that ThinkQE is significantly better than HyDE and LameR on 6 out of 7 and 5 out of 7 domains, respectively, demonstrating its effectiveness.

A.9 Impact of Number of Rounds on Domain-Specific Dataset

To further analyze the potential topic drift issue during the iterative process, we examine performance changes across one to three rounds using the

domain-specific TREC-Covid dataset. The results, presented in Table 9, demonstrate that increasing from one to two rounds improves the performance, while extending to three rounds largely maintains performance.

Round	NDCG@10
1	75.2
2	76.2
3	76.1

Table 9: Performance across iterative rounds on the domain-specific TREC-COVID dataset.

	Bio.	Earth.	Econ.	Psy.	Rob.	Stack.	Sus.	Avg.
QWEN-BASE-14B	36.7	45.1	21.9	27.7	16.8	23.3	21.7	27.6
QWEN-R1-14B *w/o. thinking*	39.1	45.6	25.0	30.0	18.0	26.5	24.4	29.8
QWEN-R1-14B *w. thinking*	42.6	50.6	26.2	35.8	18.8	28.4	25.1	32.5

Table 10: Detailed results on the impact of the thinking process.

#LLM calls	Bio.	Earth.	Econ.	Psy.	Rob.	Stack.	Sus.	Avg.
Parallel scaling								
1	42.6	47.3	25.1	30.3	18.1	24.8	25.2	30.5
2	42.6	50.6	26.2	35.8	18.8	28.4	25.1	32.5
3	44.2	50.4	26.6	33.6	18.0	26.5	26.5	32.3
4	42.4	49.8	27.7	35.5	17.8	28.0	27.4	32.7
5	41.7	50.7	26.7	35.2	19.3	27.5	27.4	32.6
6	45.3	50.3	26.4	34.5	19.0	28.2	28.0	33.1
Corpus-interaction scaling								
2	42.6	50.6	26.2	35.8	18.8	28.4	25.1	32.5
4	45.9	52.6	28.3	39.0	18.7	28.5	28.0	34.4
6	47.3	52.5	29.2	40.0	19.3	28.0	27.9	34.9

Table 11: Detailed results on the impact of the evolving corpus interaction.

Accum.	Filter	Bio.	Earth.	Econ.	Psy.	Rob.	Stack.	Sus.	Avg.
✓	✗	46.4	51.5	27.8	39.5	17.9	28.2	28.0	34.2
✗	✓	47.5	50.7	27.9	34.8	17.7	26.5	28.4	33.4
✓	✓	47.3	52.5	29.2	40.0	19.3	28.0	27.9	34.9

Table 12: Detailed results on the impact of the expansion accumulation and redundancy filter mechanism.

λ	Bio.	Earth.	Econ.	Psy.	Rob.	Stack.	Sus.	Avg.
1	38.2	47.5	25.0	32.1	18.4	25.5	24.0	30.1
2	44.9	50.1	28.2	37.0	19.3	27.0	26.8	33.3
3	46.7	51.2	29.4	39.3	20.8	28.3	28.4	34.9
4	48.9	52.3	28.2	38.4	20.2	28.1	29.4	35.1
5	49.7	52.5	29.5	37.7	19.6	28.9	27.6	35.1
6	49.1	51.8	29.5	39.1	19.4	29.0	28.8	35.2

Table 13: Detailed results on the impact of λ.

Model	Thinking	Round	Latency (second/query)	Bright Avg.
R1-14B	No	1	3.71	29.8
R1-32B	No	1	7.88	30.4
R1-14B	Yes	1	15.40	32.5
R1-32B	Yes	1	30.53	32.9
R1-14B	Yes	3	45.44	34.9

Table 14: Effectiveness-Efficiency Analysis on ThinkQE.

Method	Bio.	Earth.	Econ.	Psy.	Rob.	Stack.	Sus.	Avg.
HyDE	33.3	44.9	21.1	29.8	16.3	24.1	21.0	27.2
LameR	35.1	46.1	23.7	31.0	17.7	26.4	25.3	29.3
ThinkQE	$47.3^{\dagger\ddagger}$	$52.5^{\dagger\ddagger}$	$29.2^{\dagger\ddagger}$	$40.0^{\dagger\ddagger}$	19.3	$28.0^{\dagger}$	$27.9^{\dagger\ddagger}$	34.9

Table 15: Significance testing results. † and ‡ mean ThinkQE performs significantly better than HyDE and LameR, respectively, as determined by a t-test with p-value 0.05 as threshold.