June 16, 2025

TongSearch-QR: Reinforced Query Reasoning for Retrieval

Xubo Qin[1] **Jun Bai**[1] **Jiaqi Li**[1] **Zixia Jia**[1] **Zilong Zheng**[1]

[1] NLCo Lab, Beijing Institute for General Artificial Intelligence (BIGAI)

Traditional information retrieval (IR) methods excel at textual and semantic matching but struggle in reasoning-intensive retrieval tasks that require multi-hop inference or complex semantic understanding between queries and documents. One promising solution is to explicitly rewrite or augment queries using large language models (LLMs) to elicit reasoning-relevant content prior to retrieval. However, the widespread use of large-scale language models like GPT-4 or LLaMA3-70B remains impractical due to their high inference cost and limited deployability in real-world systems. In this work, we introduce **TongSearch-QR**, a family of small-scale language models for query reasoning and rewriting in reasoning-intensive retrieval. With a novel semi-rule-based reward function, we employ reinforcement learning approaches enabling smaller language models, e.g., `Qwen2.5-7B-Instruct` and `Qwen2.5-1.5B-Instruct`, to achieve query reasoning performance rivaling large-scale language models without their prohibitive inference costs. Experiment results on BRIGHT (Su et al., 2024) benchmark show that with BM25 as retrievers, both `TongSearch-QR-7B` and `TongSearch-QR-1.5B` models significantly outperform existing baselines, including prompt-based query reasoners and some latest dense retrievers trained for reasoning-intensive retrieval tasks, offering superior adaptability for real-world deployment.

Code `https://github.com/bigai-nlco/TongSearch-QR`

1. Introduction

The Information retrieval system (IR) (Zhu et al., 2023) plays a critical role in enabling users to locate relevant materials from vast repositories of documents, Web pages, and structured records (Bajaj et al., 2018; Karpukhin et al., 2020; Datta et al., 2008). Existing retrieval methods mainly focus on measuring the relevance between queries and documents via text matching or semantic representation techniques, e.g., BM25 algorighm (Robertson & Zaragoza, 2009) or document embedding models (Devlin, 2018; Liu, 2019; Chen et al., 2024a; Ma et al., 2024), achieving considerable success (Liu et al., 2021; Tang et al., 2025; Ye et al., 2024). However, in real-world applications, users may issue questions with high complexity, and finding the relevant documents requires intensive reasoning (Su et al., 2024). For example, a programmer may ask a question to find a function (denoted as $Func_b$) that can be an alternative option to the given function $Func_a$. The ground truth document may be an introduction of $Func_b$, while it may be completely disconnected from the question, both lexically and semantically. The document might not mention $Func_a$ at all, nor contain any explicit cues suggesting its substitutability. In such cases, traditional retrieval methods that rely on lexical overlap or shallow semantic similarity often fail, as they cannot capture the implicit reasoning chain required to connect the user's intent (i.e., "find an alternative to $Func_a$") to the actual answer (i.e., "The description of $Func_b$ ").

[1] Correspondence to: Zilong Zheng <zlzheng@bigai.ai>.

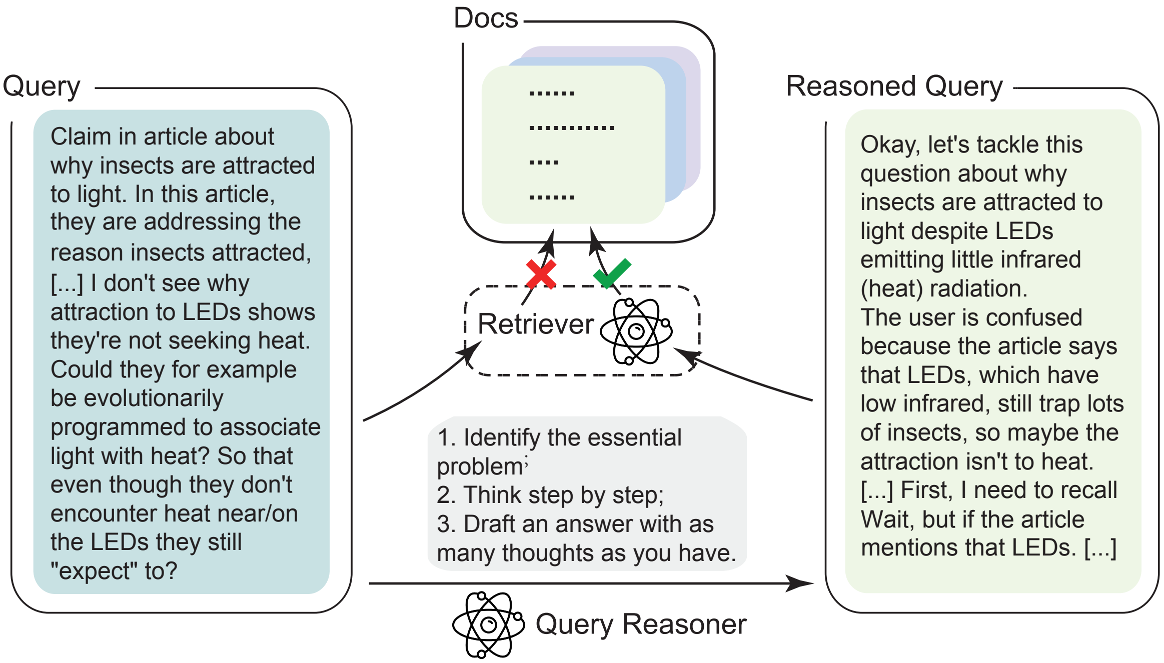

Docs
......
....
....
......
Query
Claim in article about why insects are attracted to light. In this article, they are addressing the reason insects attracted, [...] I don't see why attraction to LEDs shows they're not seeking heat. Could they for example be evolutionarily programmed to associate light with heat? So that even though they don't encounter heat near/on the LEDs they still "expect" to?
Retriever
1. Identify the essential problem;
2. Think step by step;
3. Draft an answer with as many thoughts as you have.
Query Reasoner
Reasoned Query
Okay, let's tackle this question about why insects are attracted to light despite LEDs emitting little infrared (heat) radiation.
The user is confused because the article says that LEDs, which have low infrared, still trap lots of insects, so maybe the attraction isn't to heat.
[...] First, I need to recall Wait, but if the article mentions that LEDs. [...]

Figure 1. An example query reasoning with LLM. The query is sampled from the Biology Subtask of BRIGHT (Su et al., 2024) benchmark.

This highlights a fundamental challenge in information retrieval for complex queries: the need to bridge a reasoning gap between the user's implicit intent and the relevant knowledge. An effective retrieval system must go beyond matching surface-level expressions—it needs to infer what the user is fundamentally trying to achieve, and then identify which pieces of information in the corpus instantiate or fulfill that abstract goal. This requires not only modeling the latent intent behind a query, but also mapping that intent to the appropriate segments of knowledge, which may be distributed, implicit, or expressed in entirely different terms. These kinds of retrieval tasks demand models capable of aligning abstract user goals with semantically distant but conceptually relevant content. We refer to such tasks as **reasoning-intensive retrieval** (Su et al., 2024; Shao et al., 2025), which has been proved to be challenging for most of the existing retrieval approaches with poor performance.

To address this issue, two research directions have been proposed. One is to train novel retriever or reranker models (Shao et al., 2025; Weller et al., 2025) with task-specific reasoning data. The other is to apply **query reasoning and rewriting** to the given query (Su et al., 2024; Niu et al., 2024; Jagerman et al., 2023), leveraging the frontier reasoning capabilities of large language models (LLMs) (Guo et al., 2025; Face, 2025) with chain-of-thought reasoning (Wei et al., 2022) to generate an intermediate reasoning result as **reasoned query**, which instead will be used to retrieve the relevant documents. Figure 1 shows an example of query reasoning. **These two directions are orthogonal**: a retriever designed for reasoning-intensive tasks can take the reasoned queries as input, leading to further improvements of retrieval performance. Existing query reasoning approaches mainly rely on large-scale LLMs (e.g., GPT-4o (OpenAI, 2024) or LLama3-70B (Team, 2024)) with chain-of-thought prompts. This limits the applicability of such methods, as in many real-world RAG scenarios, high-performance commercial models like GPT-4o are not accessible due to inference cost or information security concerns. Moreover, the query rewriting model deployed is generally not expected to outperform the main model (Shao et al., 2025) in RAG pipelines. For example, suppose that an RAG system is primarily built around a small-scale model instance such as Qwen2.5-7B-Instruct. In that case, the query reasoning module is typically constrained to use a model of equal or smaller capacity, which cannot outperform the main model of Qwen2.5-7B mentioned above. As a result, query reasoning methods that rely on high-performance models and Chain-of-Thought prompts are not feasible in such realistic RAG settings.

In this paper, we introduce TongSearch-QR, a family of small-scale language models for query reasoning and rewriting (QR). To the best of our knowledge, this is the first model family specifically trained for query reasoning in reasoning-intensive retrieval tasks. Inspired by previous works using Reinforcement Learning with Verifiable Rewards (RLVR) to enhance LLMs' reasoning (Guo et al., 2025; Qwen et al., 2025), we developed a novel semi-rule-based reward function for GRPO (Group Relative Policy Optimization) (Shao et al., 2024; Guo et al., 2025), enabling RL on the query reasoning of smaller language models. Beyond that, we propose an automatic data curation pipeline for training reasoning-based rewriting with publicly available dataset (Lambert et al., 2023). Experiment results on the BRIGHT (Su et al., 2024) benchmark show that our model achieves an NDCG@10 metric of 27.9, outperforming GPT-4o's metric of 26.5. This metric is comparable to some large-scale reasoning models, e.g., o1-preview[1], DeepSeek R1(Guo et al., 2025), and QwQ-32B (Qwen et al., 2025), with significantly lower cost of inference shows in Figure 2. Besides, our proposed models can also work with reasoning-intensive retrievers (Shao et al., 2025) to achieve the best performance, as shown in Figure 3. This demonstrates that our models possess strong flexibility to adapt to different retrieval pipelines.

In summary, our **main contributions** are listed as follows:

- **Query reasoning models for retrieval tasks:** We propose TongSearch-QRfamily (7B and 1.5B) specifically trained for query reasoning and rewriting in reasoning-intensive retrieval tasks. Our small-scale language models are comparable to state-of-art large-scale language models such as GPT-4o on specific tasks. These results make it possible to apply query reasoning in many real-world RAG system settings. Besides, our query reasoning models can be jointly applied to different existing retrievers to achieve better performance.
- **Semi-rule-based reward function for RL:** The reward function inherits the advantage of existing functions based on semantic similarity, which evaluates the relevance enhancement between queries and retrieval documents. It offers a range of advantages, including strong robustness, high computational efficiency, and the avoidance of reward hacking.
- **Automatic data curation pipeline:** The data curation pipeline proposed in this paper is specifically designed to build training data for query rewriting tasks. It optimizes training without the need for large-scale supervised query reasoning data, which is often unavailable in real applications and scenarios.

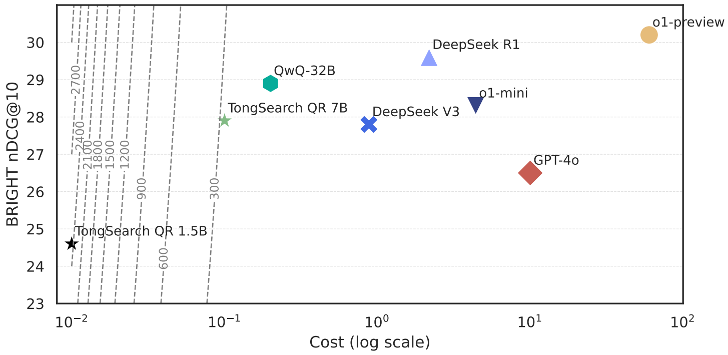


Figure 2. Cost vs. Performance comparison of different models. Details about the cost and performance can be found in Table 2.

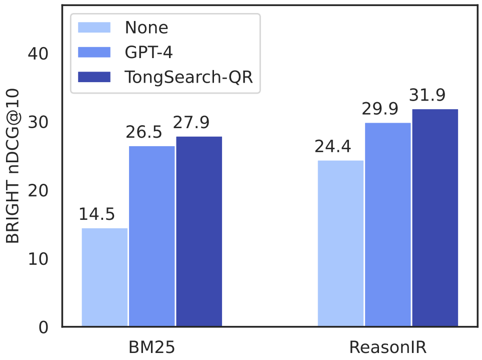


Figure 3. Performance on different reasoning models. Details can be found in Table 1

2. Related Works

Reasoning-intensive Retrieval In recent years, dense retrieval has achieved remarkable progress in retrieval accuracy, propelled by the rapid evolution of foundation models and innovative training methodologies (Luo et al., 2024; Gao et al., 2022; Lee et al., 2024; Wei et al., 2024). Nowadays, BERT (Devlin, 2018)-based and LLM-based (Wang et al., 2023; Luo et al., 2024) embedding models have been widely used in multiple retrieval tasks, achieving great success as general-purpose retrievers (Wang et al., 2022; Li et al., 2023; Chen et al., 2024a; Khattab & Zaharia, 2020). However, recent work of BRIGHT Benchmark (Su et al., 2024) has demonstrated

[1] https://openai.com/index/introducing-openai-o1-preview/

that most of the existing BERT-based or LLM-based retrievers and re-rankers perform poorly on the task of reasoning-intensive retrieval (Shao et al., 2025). These results suggest that existing retrieval methods, which assess relevance based on textual or semantic similarity, fall short in capturing the deep-dived, reasoning-driven relevance that arises in complex knowledge-seeking scenarios. To address this issue, some researchers tried to train reasoning-enhanced retrievers (Shao et al., 2025) or rerankers (Weller et al., 2025) with public or LLM-Synthesized datasets. Another way is to apply LLMs for query reasoning and rewriting. The LLMs take the original queries as input to generate Chain-of-Thought reasoning steps as pseudo queries. These pseudo queries contain richer knowledge and contextual information related to the original question, they are more likely to retrieve relevant documents than the original queries when issued to a text-based or semantic-based retriever. The two approaches mentioned above are orthogonal and can be combined synergistically. Most of those existing query reasoning approaches (Su et al., 2024; Niu et al., 2024) are based on prompting large-scale LLMs, e.g., GPT-4o (OpenAI, 2024) or LLama3-70B (Team, 2024), which are not available in many real-world settings of RAG systems. To the best of our knowledge, none of those previous works focuses on training a small-scale language model for query reasoning and rewriting tasks.

Reinforcement Learning with Verifiable Reward Large reasoning models, e.g., `OpenAI o1`, `Gemini Flash-Thinking`[2], `DeepSeek-R1` (Guo et al., 2025) and `QwQ-32B` (Qwen et al., 2025), have achieved great success in reasoning-intensive areas like coding and mathematical proofs. These models adopt a "slow-thinking" (Wu et al., 2024; Chen et al., 2024b) approach for the models to first output a sequence of thinking processes with the tags of "<think></think>" before providing the actual answer. This method has allowed LLMs to enhance their reasoning capabilities to achieve further performance improvements on math or coding tasks (Face, 2025). Inspired by the technical report released by DeepSeek (Guo et al., 2025), researchers (Face, 2025; Xie et al., 2025) have tried to reproduce the slow-thinking ability on smaller-scaled LLMs via reinforcement learning based on GRPO (Group Relative Policy Optimization) (Shao et al., 2024) and rule-based reward functions. Compared with model-based reward functions (e.g., reward functions based on process reward model (Zhang et al., 2025)), verifiable reward functions have the advantages of being simple and effective, making the model training process easier to scale up. Besides, rule-based reward functions focus solely on the correctness of output results, ignoring the intermediate process, which makes them immune to reward hacking and increases the robustness of model training. Moreover, unlike supervised fine-tuning (SFT), reinforcement learning based methods do not force the model to fit every generated token, thereby yielding superior generalization capabilities.

3. TongSearch QR

3.1. Task Formulation

Given a query q and a set of candidate documents $D = \{d_1, ..., d_n\}$, the objective of information retrieval task is to identify and retrieve a subset of relevant documents from D: $D^+ = \{d_1^+, ..., d_i^+, ..., d_m^+\}$, where $m \ll n$ leveraging a retriever $\mathcal{RT}$. In reasoning-intensive retrieval, instead of text or semantic similarity, Q is relevant to D^+ through a specific reasoning path or explanation (e.g., underlying principles, algorithms, or theorems) associated with the query. For instance, typical reasoning paths may involve identifying the query intent, analyzing and modeling the problem, and deriving sub-conclusions based on the provided descriptions. Such reasoning paths usually do not exist explicitly in the query itself, making direct retrieval based solely on the query highly challenging. In this paper, we refer to the process of generating reasoning paths or explanations for query Q as **query reasoning**.

In this paper, we denote $\mathcal{LLM}$ as a large language model for query reasoning and generating the rewritten query $q^{'}$ based on q. $\mathcal{RT}$ will later use $q^{'}$ to retrieve the documents relevant to q. The processes mentioned above can be described with the following equations:

$$q^{'} = \mathcal{LLM}(\text{Inst}; q), \qquad D^+ = \mathcal{RT}(q^{'}),$$

where Inst denotes the instructions for query reasoning and rewriting.

[2] https://deepmind.google/technologies/gemini/flash-thinking/

3.2. Reinforcement Learning with Semi-Rule-Based Reward

Preliminary Inspired by previous works of large reasoning models e.g., DeepSeek R1 (Guo et al., 2025), we employ the GRPO-based reinforcement learning algorithm to train the LLMs for query reasoning, where the model takes the given query q as input and generates a reasoned query $q^{'}$. The GRPO objective is defined as:

$$\mathcal{L}_{\text{GRPO}}(\theta) = \mathbb{E}_{(q,a)\sim\pi_\theta}\big[w_g \cdot \min\big(r_\theta(q,q^{'}) \cdot \hat{A}(q,q^{'}), \\ \text{clip}(r_\theta(q,q^{'}), 1-\epsilon, 1+\epsilon) \cdot \hat{A}(q,q^{'})\big)\big]$$

Here, $r_\theta(q,q^{'}) = \frac{\pi_\theta(q^{'}|q)}{\pi_{\theta_{\text{old}}}(q^{'}|q)}$ is the importance ratio between the current and reference policy. The advantage function $\hat{A}(q,q^{'})$ is computed based on the group-normalized reward:

$$\hat{A}(q,q^{'}) = \frac{R(q,q^{'}) - \mu_g}{\sigma_g + \delta},$$

where $R(q,q^{'})$ is the reward assigned to the reasoned query $q^{'}$, μ_g and σ_g denote the mean and standard deviation of rewards within the group g, and δ is a small constant to avoid division by zero. The weight w_g optionally rescales the advantage based on group-level reward variance. This formulation stabilizes training when rewards are sparse or highly variable across different query groups.

Limitations for Previous Rule-based Reward Function Previous approaches (Jiang et al., 2025) of rule-based reward for retrieval tasks are usually calculated based on retrieval evaluation metrics like Recall@K. The metric-based reward function requires both annotated training data and an existing large-scale document collection to serve as the retrieval source, which is difficult to access in reasoning-intensive retrieval tasks.

Semi-Rule-Based Reward for Query Reasoning In this work, we introduce a reward function to evaluate the incremental relevance score from $< q, D^+ >$ to $< q^{'}, D^+ >$. For a reasoning-intensive task, the goal of query reasoning and rewriting is to improve the retrieval performance using a reasoned query $q^{'}$ with a higher relevance score compared to q. Since the relevance score is computed via an existing relevance model, the reward function is defined as a "semi-rule-based reward function".

Each training sample consists of $< q, D^+ >$, where D^+ indicates single or multiple positive documents for q. We define $score_q$ as the sum of the relevance scores between q and each positive document in D^+:

$$score_q = \Sigma_{i\in D^+}\text{Rel}(q, d_i^+)$$

where $Rel(q, d_i^+)$ denotes the relevance score between q and d_i^+ computed via a relevance model. Here we use a pretrained embedding model to encode queries and documents into embeddings, with the cosine similarities as relevance scores. The parameters of the relevance model will not be updated during the model training process. Similarly, the score of the reasoned query $score_{q^{'}}$ is also computed as:

$$score_{q^{'}} = \Sigma_{i\in D^+}\text{Rel}(q^{'}, d_i^+)$$

The overall reward is defined as the average relevance score increment from q to $q^{'}$ of each positive document:

$$R(q,q^{'}) = \frac{score_{q^{'}} - score_q}{|D^+|}$$

Our semi-rule-based reward function inherits a few advantages from the existing rule-based rewards as follows: Firstly, the function depicts the semantic relevance based on the existing embedding model like bge-base-en (Chen et al., 2024a), which has been proved to exhibit good performance with robustness and low computational cost. Secondly, unlike the process reward models (PRMs), our method does not rely on intermediate processes supervision, and is therefore inherently immune to reward hacking. These properties collectively contribute to the high computational efficiency and robustness of our method, enhancing its tolerance to noise present in the training data.

3.3. Training Data Curation

Existing training datasets like e.g., MSMACRO (Bajaj et al., 2018) are helpful for semantic-based retrieval tasks, which are not specifically designed for reasoning-intensive retrieval. Inspired by the data construction process in benchmark BRIGHT (Su et al., 2024), we use the publicly available H4 Stack Exchange Preferences (Lambert et al., 2023) dataset to construct our training data. The dataset contains questions and answers from the Stack Overflow Data Dump for the purpose of preference model training. Each question in the dataset includes at least two answers, and each answer is labeled "is_selected" or not, indicating if the answer is selected and marked as useful by the real users who issued the question. We select QAs with texts only for data curation.

Here are two ways we further obtain the rewritten queries as the "supervision" for query reasoning training:

(1) Given a query for reasoning, a large reasoning model, e.g., QwQ-32B or DeepSeek-R1, is asked to generate the rewritten query based on Chain-of-Thought(CoT) reasoning. The curated data is denoted as **V1-R1** and **V1-QwQ**.

(2) For each question, we use the answer with the "selected" tag as the reasoned query from StackExchange by real users, which is denoted as **V2**. Notice that not every question includes a selected answer.

Details about the actual dataset in use will be further described in Section 4.1.1 and Appendix B.

	StackExchange							**Coding**		**Theorem-based**			**Avg**
	Bio.	Earth.	Econ.	Psy.	Rob.	Stack.	Sus.	Leet.	Pony	AoPS	TheoQ.	TheoT.	
						Retrievers with Original Queries							
BM25	18.9	27.2	14.9	12.5	13.6	18.4	15.0	24.4	7.9	6.2	10.4	4.9	14.5
BGE	11.7	24.6	16.6	17.5	11.7	10.8	13.3	26.7	5.7	6.0	13.0	6.9	13.7
ReasonIR	26.2	31.4	23.3	30.0	18.0	23.9	20.5	35.0	10.5	14.7	31.9	27.2	24.4
Seed1.5-Embedding	34.8	46.9	23.4	31.6	19.1	25.4	21.0	43.2	4.9	12.2	33.3	30.5	27.2
						Query Reasoner with BM25							
GPT-4o	53.6	53.6	24.3	38.6	18.8	22.7	25.9	19.3	17.7	3.9	18.9	20.2	26.5
Doubao	54.8	53.3	23.7	37.2	22.2	28.1	25.0	21.2	16.4	7.8	21.8	22.7	27.8
Deepseek-V3	56.6	54.2	25.8	38.8	19.9	26.7	26.4	19.8	15.1	6.7	22.5	20.7	27.8
o1-mini	60.2	57.4	24.7	39.3	23.3	26.4	25.4	23.5	13.4	6.9	22.8	16.5	28.3
o1-preview	**64.2**	57.9	27.6	**43.1**	**25.6**	29.1	28.0	21.2	15.9	5.6	24.0	20.5	30.2
Deepseek-R1	62.7	**58.3**	26.0	42.9	21.8	28.1	**30.3**	19.6	10.7	6.0	25.8	22.4	29.6
R1-distill-qwen-7B	33.9	41.6	19.9	31.8	15.1	18.8	16.4	19.7	10.7	6.8	24.5	22.2	21.8
R1-distill-qwen-32B	50.6	49.9	22.9	38.1	20.3	24.6	19.2	19.5	11.3	5.6	24.2	20.2	25.5
QwQ-32B	57.5	56.3	29.9	41.8	19.2	25.7	27.2	21.5	12.8	6.5	25.4	22.8	28.9
TongSearch-QR-1.5B	46.0	47.1	21.1	31.2	19.8	21.7	24.3	22.5	**21.7**	4.3	19.7	15.9	24.6
TongSearch-QR-7B	57.9	50.9	21.9	37.0	21.3	27.0	25.6	23.6	14.4	7.0	26.1	22.0	27.9
						Query Reasoner with ReasonIR							
LLama3.1-8B-Instruct	37.8	39.6	29.6	35.3	24.1	**31.1**	27.4	28.8	14.5	9.2	26.6	32.3	28.0
GPT-4	43.6	42.9	**32.7**	38.8	20.9	25.8	27.5	31.5	19.6	7.4	33.1	35.7	29.9
TongSearch-QR-1.5B	36.4	41.1	29.9	34.0	25.2	30.7	25.6	**33.3**	16.8	9.7	35.7	32.7	29.3
TongSearch-QR-7B	46.2	45.1	31.2	39.6	25.3	28.7	28.4	31.2	16.3	**10.8**	**40.0**	**39.3**	**31.9**

Table 1. Performance comparison on BRIGHT. The best score is shown in bold and the second best is underlined.

4. Experiment

4.1. Experimental Setup

4.1.1. Dataset and Metrics

Training We employ two types of the constructed data mentioned in Section 3.3 for training: **V1-R1**, **V1-QwQ** and **V2**. For V2, we use the user-selected answers since the size of V2 is too large to afford the inference cost of large reasoning models. More details can be found in Appendix B.

Evaluation We use BRIGHT (Su et al., 2024), a novel benchmark for reasoning-intensive retrieval that aims to evaluate the ability of retrieval models to handle complex queries that require deep reasoning. It consists of 1,384 real-world queries from diverse domains with 12 sub-tasks. We adopt the metric **nDCG@10** for the following evaluations.

4.1.2. Baselines

The baselines in our experiments can be divided into these three categories:

Retrievers with Original Queries There are two types of baselines: 1) Traditional baselines in IR systems like BM25 (Robertson & Zaragoza, 2009) for sparse retrieval and bge-large-en (Chen et al., 2024a)for dense retrieval; 2) Reasoning-intensive retrievers like ReasonIR (Shao et al., 2025) and Seed 1.5-Embedding[3]. We keep the same with the experiments reported in (Su et al., 2024) for fair comparison and all the retrievers use the original queries in BRIGHT to retrieve documents. Since Seed1.5-Embedding is not public available when this work is done, we directly use the experiment results reported on their model card.

Query Reasoner with BM25 We include two types of baselines using state-of-the-art large language models: 1) Non-reasoning models including GPT-4o, doubao-1.5-pro [4], DeepSeek-V3 (DeepSeek-AI et al., 2025); 2) Reasoning models including DeepSeek R1 (Guo et al., 2025), o1-mini[5], o1-preview[6], DeepSeek-R1-Distill-Qwen-7B[7], DeepSeek-R1-Distill-Qwen-32B[8] and QwQ-32B (Qwen et al., 2025). All the models use the prompt in Appendix A for reasoning. For each baseline, we only retain the prediction result after reasoning and use BM25 for further retrieval.

Query Reasoner with Reasoning-Intensive Retrievers (ReasonIR) ReasonIR (Shao et al., 2025) is the most recently acknowledged retriever specifically trained for reasoning-intensive retrieval tasks. We further combine TongSearch-QRwith ReasonIR for comparison to explore further improvements with the specialized reasoner and retriever in this task.

4.1.3. Implementation Details

With the initial checkpoint of Qwen2.5-7B-Instruct[9] and Qwen2.5-1.5B-Instruct[10], TongSearch-QR7B and 1.5B are both trained with TRL codebase[11] on a single node with 4 NVIDIA A800-80G GPUs. Following the instructions of Open-R1 (Face, 2025), we use 1 GPU for vLLM (Kwon et al., 2023) serving and the rest 3 GPUs for model training. DeepSpeed (Rasley et al., 2020), ZeRO-3, and Gradient Checkpoint are applied to reduce the cost of VRAM. It takes about 16 hours for 1.5B model training and about 48 hours for 7B model training. We set the learning rate $1e-6$, the batch size per device 16, and the KL coefficient 0.008. For each input prompt, 16 samples

[3] https://huggingface.co/ByteDance-Seed/Seed1.5-Embedding

[4] https://seed.bytedance.com/en/special/doubao_1_5_pro

[5] https://openai.com/index/openai-o1-mini-advancing-cost-efficient-reasoning/

[6] https://openai.com/index/introducing-openai-o1-preview/

[7] https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-7B

[8] https://huggingface.co/deepseek-ai/DeepSeek-R1-Distill-Qwen-32B

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

[10] https://huggingface.co/Qwen/Qwen2.5-1.5B-Instruct

[11] https://github.com/huggingface/trl

are generated to estimate the advantage in GRPO. Since we use bge-base-en-v1.5[12] embedding model to compute relevance, the maximum completion length is set to 500 to avoid exceeding the input length limitation of the embedding model. Experiments on all the above-mentioned baselines are conducted without reranking.

4.2. Main Results

Table 1 shows that our 7B model outperforms all query reasoning baselines of non-reasoning LLMs, including GPT-4o and DeepSeek V3, performing comparable to the large reasoning models, e.g., o1-mini, QwQ-32B and DeepSeek R1. Our 7B model strikes a favorable balance between inference efficiency and reasoning performance, offering a compelling trade-off for query reasoning tasks. Besides, our 1.5B model also achieves performance comparable to that of large-scale language models, making it an effective solution for resource-constrained scenarios.

To quantitatively assess the efficiency of different models, we report both their *Performance* and *Cost* in Table 2. Here, **Performance** is defined as the nDCG@10 score achieved by each model on the BRIGHT benchmark (Su et al., 2024) with BM25 retriever. Meanwhile, **Cost** represents the price of each model (USD per 1M output tokens) when accessed via the OpenRouter platform[13], indicating the actual monetary expense required to obtain outputs from the model[14]. Based on the calculated efficiency (*Eff = Performance / Cost*), TongSearch-QR-1.5B and TongSearch-QR-7B achieve the highest cost-effectiveness among all evaluated models, with efficiency scores of **2460.0** and **279.0**, respectively. This highlights the strong cost-performance advantage of the our method.

Model	Performance	Cost	Efficiency
GPT-4o	26.5	10.0	2.7
DeepSeek V3	27.8	0.9	31.6
DeepSeek R1	29.6	2.2	13.6
QwQ-32B	28.9	0.2	144.5
o1-preview	30.2	60.0	0.5
o1-mini	28.3	4.4	6.4
TongSearch-QR-7B	27.9	0.1	279.0
TongSearch-QR-1.5B	24.6	0.01	**2460.0**

Table 2. Model performance (Perf.), cost, and efficiency (Eff. = Perf. / Cost).

Compared to the retrievers specifically trained for reasoning, both the 7B and 1.5B models can both outperform ReasonIR with original query, and our 7B model can outperform Seed 1.5-Embedding. Since ReasonIR is an embedding model based on the backbone of LLaMA3.1-8B, the computational cost of pre-encoding documents can be prohibitive when the corpus is large. In contrast, TongSearch-QRcan work with BM25 retrievers, incurring significantly lower pre-processing costs than LLM-based embedding models.

As ReasonIR can work with reasoned queries to achieve better performance, we apply the reasoned queries generated by our method for further exploring the effect of combining reasoned queries with reasoning-intensive retrievers. We confirmed that our method can achieve further improvements based on ReasonIR. Our 1.5B model can outperform LLama3.1-8B-Instruct which is more than 5 times larger in parameters, and our 7B model can outperform GPT-4. Comparing with the reasoned queries of GPT-4 and our 7B model , the performance improvement based on ReasonIR (29.9->31.9) is higher than the improvement based on BM25 (26.5->27.9). These results incidate that our method is flexible and can work with different retrievers, the improvement of retriever will further expand the advantage of our method.

Dataset	Bio.	Earth.	Econ.	Psy.	Rob.	SO	SL	LC	Pony	AoPS	TQ	TT	**Avg**
V1-QwQ	42.3	43.6	19.1	**31.9**	18.6	**23.7**	22.8	21.5	17.7	4.0	17.2	10.1	22.7
V1-R1	**48.0**	46.5	19.9	31.4	15.2	23.6	22.3	21.2	18.0	**5.3**	18.8	11.3	23.5
V2	46.0	**47.1**	**21.1**	31.2	**19.8**	21.7	**24.3**	**22.5**	**21.8**	4.3	**19.7**	**15.9**	**24.6**

Table 3. Results on different training data for TongSearch-QR-1.5B with BM25 retriever.

4.3. Ablation Studies

4.3.1. Effect of Data Size and Quality

DeepSeek R1 performs better than QwQ-32B while the performance of V1-R1 and V1-QwQ is close on most BRIGHT subtasks. V1-R1 exhibits a notable advantage only in the Biology and Earth Science subtasks. We hypothesize that this may be attributed to the fact that these two subtasks are more knowledge-intensive compared to others, thereby granting the larger-parameter DeepSeek R1 model with 671B parameters a more pronounced advantage over QwQ-32B. The V2 dataset with more samples leads to the best performance. Instead of using large reasoning models to generate answers for distillation, a better approach may be to use the answers selected by the users in the StackExchange datasets. It can be easily scaled since generating answers with large reasoning models on large-scale question set is too expensive.

4.3.2. Effect of Reinforcement Learning with Semi-Rule-Based Rewards

We further explore the effect of our proposed approaches with semi-rule-based reward functions compared to traditional supervised fine-tuning (SFT). Following the same experimental settings in Section 4.3.1, we use Qwen2.5-1.5B-Instruct, with BM25 as retrievers. With the dataset of V1-QwQ and V2, we separately trained the model with SFT and RL. Results are shown in Table 4c, where "RL" is for our proposed reinforced learning approaches and "SFT" is for supervised fine-tuning. Both RL and SFT are in full parameters.

These results indicate that when using the training data generated by large reasoning models, the performance of RL is slightly higher than SFT. - While using the user-selected answer data for training, the performance of SFT experienced a significant decline. This is likely because the user-selected answers written by actual users may exhibit substantial quality deficiencies (e.g., higher perplexity) compared to data synthesized by large reasoning models. In addition, we did not apply fine-grained data cleaning for the answer. As a result, the answers of the questions may include URL links of pictures which do not include available information. Using such data for supervised fine-tuning may lead to catastrophic forgetting in the model. In contrast, our proposed reinforcement learning approach with semi-rule-based reward functions does not strictly require the model to fit the answers per token exactly. Since the relevance score in reward function is based on the embedding similarity of generated answers and selected answers, the noisy signals in selected answers may not explicitly affect the similarity scores. As a result, our proposed approach demonstrates stronger generalization capabilities and greater tolerance for noisy data.

4.3.3. Effect of Relevance Model in Reward Functions

As we mentioned in Section 3.2, the relevance model is playing an important role in our proposed semi-rule-based reward functions. We further explore the effect of different relevance models in our proposed reward functions. Besides the dense embedding model of bge-base-en-v1.5, we also implement the relevance function via the sparse model of bge-m3 (Chen et al., 2024a). As the bge-m3 model can accept a longer input length, we also explore the effect of extending the maximum completion length to 1000. With Qwen2.5-1.5B-Instruct as base model and BM25 as retriever, we train the model on different reward functions and completion length settings on the training data of V1-R1. Results are shown in Table 4a. Since all experiments are conducted with the sparse retriever of BM25, we initially expected bge-m3, as a sparse relevance model, to offer performance improvements.

[12] https://huggingface.co/BAAI/bge-base-en-v1.5

[13] https://openrouter.ai

[14] We use the price of Qwen2.5-7B-Instruct as the price of TongSearch-QR-7B, and we define the price of TongSearch-QR-1.5B as 0.01 since the price of Qwen2.5-1.5B is free.

Type	Resp. Length	Avg
Dense	500	23.5
Sparse	500	23.1
Sparse	1000	22.9

(a) Relevance model types and Response length.

Content	Explicit	Avg
Answer	No	22.7
Think+Answer	Yes	21.4
Answer	Yes	20.9

(b) Effect of explicit thinking.

Data	Method	Avg
V1-QwQ	SFT	22.4
V1-QwQ	RL	22.7
V2	SFT	12.8
V2	RL	24.6

(c) Training data and methods.

Table 4. Performance comparison across (a) relevance model types and response length, (b) explicit thinking, and (c) training data and methods.

However, bge-m3 actually underperforms compared to the dense embedding model bge-base-en-v1.5, which has fewer parameters (110M vs 550M)[15]. This result suggests that for our proposed semi-rule-based reward function, overly fine-grained relevance matching signals may harm the model's generalization ability. It is worth noting that our training data is based on V1-R1 rather than V2, these results are unlikely to be primarily attributed to data noise since the answers are generated by DeepSeek R1. Furthermore, we observed that even increasing the output length did not improve performance, indicating that excessively long outputs might dilute the effective relevance signals, thus providing no benefit to final retrieval performance.

4.3.4. Effect of Explicit Thinking

Inspired by DeepSeek R1 (Guo et al., 2025) and some recent works (Weller et al., 2025; Xie et al., 2025), we further investigate the effect of the explicit thinking process. When the explicit thinking process is applied, the model will first think about the reasoning process explicitly and then provide the actual answer. The reasoning process and answer are enclosed with "<think></think>" and "<answer></answer>" tags. With the dataset of V1-QwQ, we train the model on Qwen2.5-1.5B-Instruct, and evaluate the query reasoners with BM25 retriever. Since the thinking process requires external output tokens, the max completion length is set to 1000 when the explicit thinking process is applied. Details about the prompt and reward settings are listed in Appendix C. Results are shown in Table 4b. In the table, "Explicitly Thinking" denotes if the explicitly thinking process is applied for model training, and "Content" denotes if the output query contains the thinking process within the "<think></think>" tags. "Thinking+Answer" means that the contents within the "<think></think>" and "<answer></answer>" tags are concatenated as the reasoned queries, and "Answer" means that only the answer content is returned.

Experimental results indicate that applying an explicit thinking process does not improve performance on query reasoning tasks. Previous studies (Weller et al., 2025) have shown that explicitly generating the reasoning process within the "<think></think>" tags can be beneficial for certain reasoning-intensive tasks such as math or coding, possibly because these tasks require the model to produce answers in specific output formats. For example, in ranking tasks, the model receives a query and a document as input and must output a binary relevance judgment (true or false). In such cases, applying an explicit thinking process can help the model fully leverage its reasoning capabilities through chain-of-thought prompting, thereby enhancing inference performance. However, in the case of query reasoning tasks, the generated reasoned query inherently encapsulates the reasoning process and is not constrained by output format requirements. As a result, explicitly generating the reasoning process does not lead to further performance gains.

5. Conclusion

In this work, we present TongSearch-QR, a family of compact and efficient language models tailored for query reasoning and rewriting in reasoning-intensive retrieval. By leveraging the learning algorithm of GRPO with a novel semi-rule-based reward function, our approach enables effective and robust reinforcement learning without relying on expensive human-annotated datasets and retrieval sources. Our proposed models demonstrate strong performance on the BRIGHT benchmark, rivaling or even surpassing large-scale commercial LLMs,

[15] bge-m3 is based on XLM-RoBERTa-Large

while significantly reducing inference cost and latency. Furthermore, TongSearch-QRmodels exhibit strong compatibility with both traditional and reasoning-intensive retrievers, making them highly versatile for real-world deployment. Our findings highlight a promising direction toward building lightweight, affordable, and high-performing reasoning components for retrieval-augmented generation pipelines and the latest deep research products.

References

Bajaj, P., Campos, D., Craswell, N., Deng, L., Gao, J., Liu, X., Majumder, R., McNamara, A., Mitra, B., Nguyen, T., Rosenberg, M., Song, X., Stoica, A., Tiwary, S., and Wang, T. Ms marco: A human generated machine reading comprehension dataset, 2018. URL https://arxiv.org/abs/1611.09268.

Chen, J., Xiao, S., Zhang, P., Luo, K., Lian, D., and Liu, Z. BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation, June 2024a. URL http://arxiv.org/abs/2402.03216. arXiv:2402.03216 [cs].

Chen, L., Davis, J. Q., Hanin, B., Bailis, P., Stoica, I., Zaharia, M., and Zou, J. Are more llm calls all you need? towards scaling laws of compound inference systems, 2024b. URL https://arxiv.org/abs/2403.02419.

Datta, R., Joshi, D., Li, J., and Wang, J. Z. Image retrieval: Ideas, influences, and trends of the new age. *ACM Comput. Surv.*, 40(2), May 2008. ISSN 0360-0300. doi: 10.1145/1348246.1348248. URL https://doi.org/10.1145/1348246.1348248.

DeepSeek-AI, Liu, A., Feng, B., Xue, B., Wang, B., Wu, B., Lu, C., Zhao, C., Deng, C., Zhang, C., Ruan, C., Dai, D., Guo, D., Yang, D., Chen, D., Ji, D., Li, E., Lin, F., Dai, F., Luo, F., Hao, G., Chen, G., Li, G., Zhang, H., Bao, H., Xu, H., Wang, H., Zhang, H., Ding, H., Xin, H., Gao, H., Li, H., Qu, H., Cai, J. L., Liang, J., Guo, J., Ni, J., Li, J., Wang, J., Chen, J., Chen, J., Yuan, J., Qiu, J., Li, J., Song, J., Dong, K., Hu, K., Gao, K., Guan, K., Huang, K., Yu, K., Wang, L., Zhang, L., Xu, L., Xia, L., Zhao, L., Wang, L., Zhang, L., Li, M., Wang, M., Zhang, M., Zhang, M., Tang, M., Li, M., Tian, N., Huang, P., Wang, P., Zhang, P., Wang, Q., Zhu, Q., Chen, Q., Du, Q., Chen, R. J., Jin, R. L., Ge, R., Zhang, R., Pan, R., Wang, R., Xu, R., Zhang, R., Chen, R., Li, S. S., Lu, S., Zhou, S., Chen, S., Wu, S., Ye, S., Ye, S., Ma, S., Wang, S., Zhou, S., Yu, S., Zhou, S., Pan, S., Wang, T., Yun, T., Pei, T., Sun, T., Xiao, W. L., Zeng, W., Zhao, W., An, W., Liu, W., Liang, W., Gao, W., Yu, W., Zhang, W., Li, X. Q., Jin, X., Wang, X., Bi, X., Liu, X., Wang, X., Shen, X., Chen, X., Zhang, X., Chen, X., Nie, X., Sun, X., Wang, X., Cheng, X., Liu, X., Xie, X., Liu, X., Yu, X., Song, X., Shan, X., Zhou, X., Yang, X., Li, X., Su, X., Lin, X., Li, Y. K., Wang, Y. Q., Wei, Y. X., Zhu, Y. X., Zhang, Y., Xu, Y., Xu, Y., Huang, Y., Li, Y., Zhao, Y., Sun, Y., Li, Y., Wang, Y., Yu, Y., Zheng, Y., Zhang, Y., Shi, Y., Xiong, Y., He, Y., Tang, Y., Piao, Y., Wang, Y., Tan, Y., Ma, Y., Liu, Y., Guo, Y., Wu, Y., Ou, Y., Zhu, Y., Wang, Y., Gong, Y., Zou, Y., He, Y., Zha, Y., Xiong, Y., Ma, Y., Yan, Y., Luo, Y., You, Y., Liu, Y., Zhou, Y., Wu, Z. F., Ren, Z. Z., Ren, Z., Sha, Z., Fu, Z., Xu, Z., Huang, Z., Zhang, Z., Xie, Z., Zhang, Z., Hao, Z., Gou, Z., Ma, Z., Yan, Z., Shao, Z., Xu, Z., Wu, Z., Zhang, Z., Li, Z., Gu, Z., Zhu, Z., Liu, Z., Li, Z., Xie, Z., Song, Z., Gao, Z., and Pan, Z. Deepseek-v3 technical report, 2025. URL https://arxiv.org/abs/2412.19437.

Devlin, J. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Face, H. Open r1: A fully open reproduction of deepseek-r1, January 2025. URL https://github.com/huggingface/open-r1.

Gao, L., Ma, X., Lin, J., and Callan, J. Precise zero-shot dense retrieval without relevance labels. *arXiv preprint arXiv:2212.10496*, 2022.

Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Jagerman, R., Zhuang, H., Qin, Z., Wang, X., and Bendersky, M. Query Expansion by Prompting Large Language Models, May 2023. URL http://arxiv.org/abs/2305.03653. arXiv:2305.03653 [cs].

Jiang, P., Lin, J., Cao, L., Tian, R., Kang, S., Wang, Z., Sun, J., and Han, J. Deepretrieval: Hacking real search engines and retrievers with large language models via reinforcement learning, 2025. URL https://arxiv.org/abs/2503.00223.

Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., and tau Yih, W. Dense passage retrieval for open-domain question answering, 2020. URL https://arxiv.org/abs/2004.04906.

Khattab, O. and Zaharia, M. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pp. 39–48, 2020.

Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C. H., Gonzalez, J. E., Zhang, H., and Stoica, I. Efficient memory management for large language model serving with pagedattention, 2023. URL https://arxiv.org/abs/2309.06180.

Lambert, N., Tunstall, L., Rajani, N., and Thrush, T. Huggingface h4 stack exchange preference dataset, 2023. URL https://huggingface.co/datasets/HuggingFaceH4/stack-exchange-preferences.

Lee, C., Roy, R., Xu, M., Raiman, J., Shoeybi, M., Catanzaro, B., and Ping, W. Nv-embed: Improved techniques for training llms as generalist embedding models. *arXiv preprint arXiv:2405.17428*, 2024.

Li, Z., Zhang, X., Zhang, Y., Long, D., Xie, P., and Zhang, M. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*, 2023.

Liu, Y. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 364, 2019.

Liu, Y., Huang, G., Liu, J., Lu, W., Cheng, S., Li, Y., Shi, D., Wang, S., Cheng, Z., and Yin, D. Pre-trained language model for web-scale retrieval in baidu search, 2021. URL https://arxiv.org/abs/2106.03373.

Luo, K., Qin, M., Liu, Z., Xiao, S., Zhao, J., and Liu, K. Large language models as foundations for next-gen dense retrieval: A comprehensive empirical assessment. *arXiv preprint arXiv:2408.12194*, 2024.

Ma, X., Wang, L., Yang, N., Wei, F., and Lin, J. Fine-Tuning LLaMA for Multi-Stage Text Retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, pp. 2421–2425, New York, NY, USA, July 2024. Association for Computing Machinery. ISBN 979-8-4007-0431-4. doi: 10.1145/3626772.3657951. URL https://doi.org/10.1145/3626772.3657951.

Niu, T., Joty, S., Liu, Y., Xiong, C., Zhou, Y., and Yavuz, S. Judgerank: Leveraging large language models for reasoning-intensive reranking, 2024. URL https://arxiv.org/abs/2411.00142.

OpenAI. Gpt-4o system card, 2024. URL https://arxiv.org/abs/2410.21276.

Qwen, :, Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., Lin, H., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Lin, J., Dang, K., Lu, K., Bao, K., Yang, K., Yu, L., Li, M., Xue, M., Zhang, P., Zhu, Q., Men, R., Lin, R., Li, T., Tang, T., Xia, T., Ren, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Wan, Y., Liu, Y., Cui, Z., Zhang, Z., and Qiu, Z. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Rasley, J., Rajbhandari, S., Ruwase, O., and He, Y. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, pp. 3505–3506, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450379984. doi: 10.1145/3394486.3406703. URL https://doi.org/10.1145/3394486.3406703.

Robertson, S. and Zaragoza, H. The probabilistic relevance framework: Bm25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389, April 2009. ISSN 1554-0669. doi: 10.1561/1500000019. URL https://doi.org/10.1561/1500000019.

Shao, R., Qiao, R., Kishore, V., Muennighoff, N., Lin, X. V., Rus, D., Low, B. K. H., Min, S., tau Yih, W., Koh, P. W., and Zettlemoyer, L. Reasonir: Training retrievers for reasoning tasks, 2025. URL https://arxiv.org/abs/2504.20595.

Shao, Z., Wang, P., Zhu, Q., Xu, R., Song, J., Bi, X., Zhang, H., Zhang, M., Li, Y. K., Wu, Y., and Guo, D. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Su, H., Yen, H., Xia, M., Shi, W., Muennighoff, N., Wang, H.-y., Liu, H., Shi, Q., Siegel, Z. S., Tang, M., et al. Bright: A realistic and challenging benchmark for reasoning-intensive retrieval. *arXiv preprint arXiv:2407.12883*, 2024.

Tang, T., Tian, Z., Zhu, Z., Wang, C., Hu, H., Tang, G., Liu, L., and Xu, S. Lref: A novel llm-based relevance framework for e-commerce search. In *Companion Proceedings of the ACM on Web Conference 2025*, WWW '25, pp. 468–475, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400713316. doi: 10.1145/3701716.3715246. URL https://doi.org/10.1145/3701716.3715246.

Team, L. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Wang, L., Yang, N., Huang, X., Jiao, B., Yang, L., Jiang, D., Majumder, R., and Wei, F. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*, 2022.

Wang, L., Yang, N., Huang, X., Yang, L., Majumder, R., and Wei, F. Improving text embeddings with large language models. *arXiv preprint arXiv:2401.00368*, 2023.

Wei, C., Chen, Y., Chen, H., Hu, H., Zhang, G., Fu, J., Ritter, A., and Chen, W. Uniir: Training and benchmarking universal multimodal information retrievers. In *European Conference on Computer Vision*, pp. 387–404. Springer, 2024.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., Zhou, D., et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837, 2022.

Weller, O., Ricci, K., Yang, E., Yates, A., Lawrie, D., and Durme, B. V. Rank1: Test-time compute for reranking in information retrieval, 2025. URL https://arxiv.org/abs/2502.18418.

Wu, Y., Sun, Z., Li, S., Welleck, S., and Yang, Y. Inference scaling laws: An empirical analysis of compute-optimal inference for llm problem-solving. In *The 4th Workshop on Mathematical Reasoning and AI at NeurIPS'24*, 2024.

Xie, T., Gao, Z., Ren, Q., Luo, H., Hong, Y., Dai, B., Zhou, J., Qiu, K., Wu, Z., and Luo, C. Logic-rl: Unleashing llm reasoning with rule-based reinforcement learning, 2025. URL https://arxiv.org/abs/2502.14768.

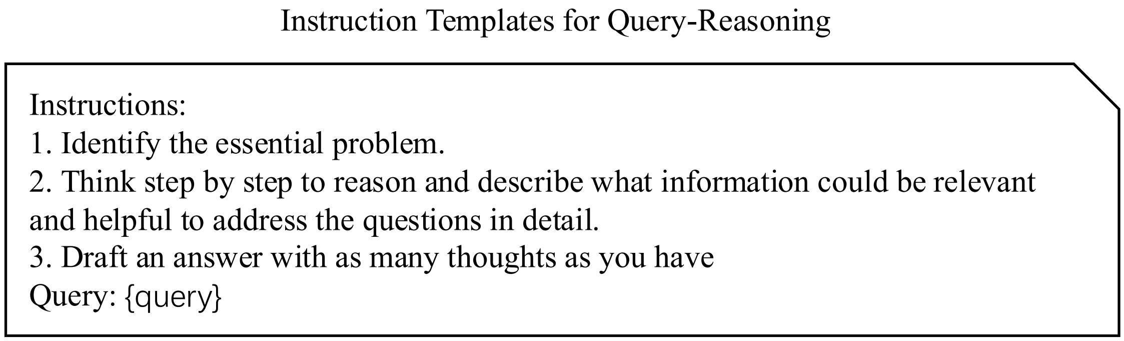
Ye, D., Liu, J., Fan, J., Tian, B., Zhou, T., Chen, X., and Ma, J. Enhancing asymmetric web search through question-answer generation and ranking. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, pp. 6127–6136, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400704901. doi: 10.1145/3637528.3671517. URL https://doi.org/10.1145/3637528.3671517.

Zhang, Z., Zheng, C., Wu, Y., Zhang, B., Lin, R., Yu, B., Liu, D., Zhou, J., and Lin, J. The lessons of developing process reward models in mathematical reasoning. *arXiv preprint arXiv:2501.07301*, 2025.

Zhu, Y., Yuan, H., Wang, S., Liu, J., Liu, W., Deng, C., Chen, H., Liu, Z., Dou, Z., and Wen, J.-R. Large language models for information retrieval: A survey. *arXiv preprint arXiv:2308.07107*, 2023.

A. Prompt Templates

Figure 4 shows the prompt template for the instructions of chain-of-thought query reasoning. The reasoner model takes the instructions and original query as input, and return a "pseudo-answer" with thoughts including as much relevant information as possible. The "pseudo-answer" can be used as the reasoned query, and the retriever can benefit from the external information provided by the reasoned query.



Instruction Templates for Query-Reasoning

Instructions:
1. Identify the essential problem.
2. Think step by step to reason and describe what information could be relevant and helpful to address the questions in detail.
3. Draft an answer with as many thoughts as you have
Query: {query}

Figure 4. The prompt template for the instructions of Chain-of-Thought query reasoning.

B. Training Data

Details about the construction of training data are described as follows:

- Version 1: sampling at most 1200 questions for each selected category to generate answers with large reasoning models. The selected categories include: 'biology', 'chemistry', 'codereview', 'cs', 'earthscience', 'economics', 'math', 'physics', 'robotics'. The Version 1 dataset includes around 10k sampled questions. In this paper, the corresponding datasets are denoted as **V1-R1** and **V1-QwQ**, indicating that the answers are generated by DeepSeek R1 or QwQ-32B.
- Version 2: sampling at most 1500 questions for each selected category with selected answers. Those questions can also be used to generate answers via large reasoning models. The categories include: 'ai', 'biology', 'chemistry','codereview', 'cs', 'earthscience', 'economics', 'computergraphics', 'math', 'mathoverflow', 'philosophy', 'physics', 'robotics', 'stackoverflow', 'sustainability', 'softwareengineering', 'bioinformatics'. The Version 2 dataset includes around 30k sampled questions, nearly three times as many as Version 1, making answer generation with large reasoning models unaffordable since the inference time is too long. In this paper, the dataset is denoted as **V2**.

C. System Prompt and Reward for Explicit Thinking

Inspired by previous works (Xie et al., 2025; Weller et al., 2025), we use the following system prompt to instruct the model to output the thinking process explicitly in the format of "<think>thinking process</think><answer>the answer</answer>".

System Prompt

You are a helpful assistant. The assistant first thinks about the reasoning process in the mind and then provides the user with the answer. The reasoning process and answer are enclosed within `<think> </think>` and `<answer> </answer>` tags, respectively, i.e., `<think>` reasoning process here `</think>` `<answer>` answer here `</answer>`.

When the explicit thinking process is applied, we also design a format reward to force the model returning an output in the correct format. Our format checking strategy is identical to (Xie et al., 2025). If the model's output fails the format checking, the reward function will immediately return a score of -1, and the subsequent computation of the query reasoning reward will be skipped.

D. License

In this section we list the artifacts we used and the corresponding URL and licenses:

Name	Type	URL	License
StackExchange-Preferences	Dataset	https://huggingface.co/datasets/HuggingFaceH4/stack-exchange-preferences	cc-by-sa-4.0
BRIGHT Benchmark	Dataset	https://huggingface.co/datasets/xlangai/BRIGHT	cc-by-4.0
Qwen2.5-1.5B-Instruct	Model	https://huggingface.co/Qwen/Qwen2.5-1.5B-Instruct	apache-2.0
Qwen2.5-7B-Instruct	Model	https://huggingface.co/Qwen/Qwen2.5-7B-Instruct	apache-2.0
bge-base-en-v1.5	Model	https://huggingface.co/BAAI/bge-base-en-v1.5	mit
bge-m3	Model	https://huggingface.co/BAAI/bge-m3	mit
QwQ-32B	Model	https://huggingface.co/Qwen/QwQ-32B	mit
DeepSeek R1	Model	https://huggingface.co/deepseek-ai/DeepSeek-R1	mit
DeepSeek V3	Model	https://huggingface.co/deepseek-ai/DeepSeek-V3-0324	mit
ReasonIR	Model	https://huggingface.co/reasonir/ReasonIR-8B	cc-by-nc-4.0

Table 5. List of datasets and models used, along with their URLs and licenses.