DeepRetrieval: Hacking Real Search Engines and Retrievers with Large Language Models via Reinforcement Learning

Pengcheng Jiang*, Jiacheng Lin*, Lang Cao*, Runchu Tian, SeongKu Kang, Zifeng Wang, Jimeng Sun, and Jiawei Han

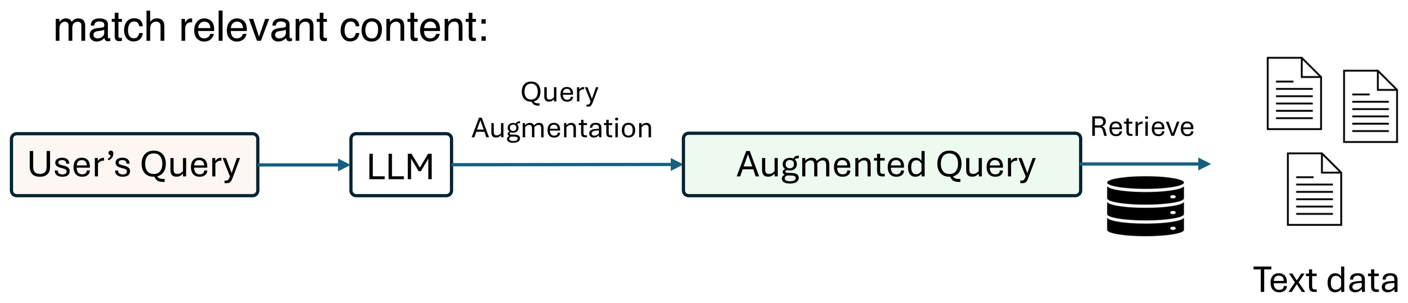
UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

University of Illinois Urbana-Champaign

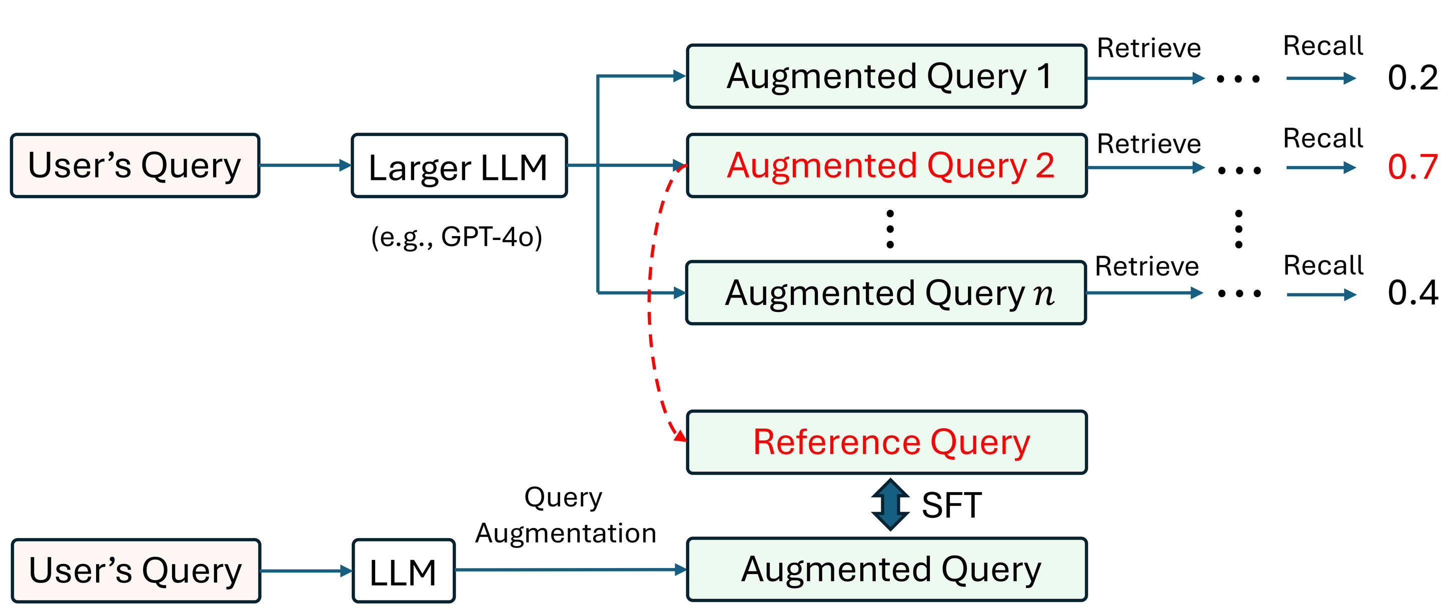
Paper   Code

Background

- Information retrieval systems often struggle with the semantic gap between user queries and relevant documents.
- **Query Augmentation** bridges this gap by reformulating queries to better match relevant content:



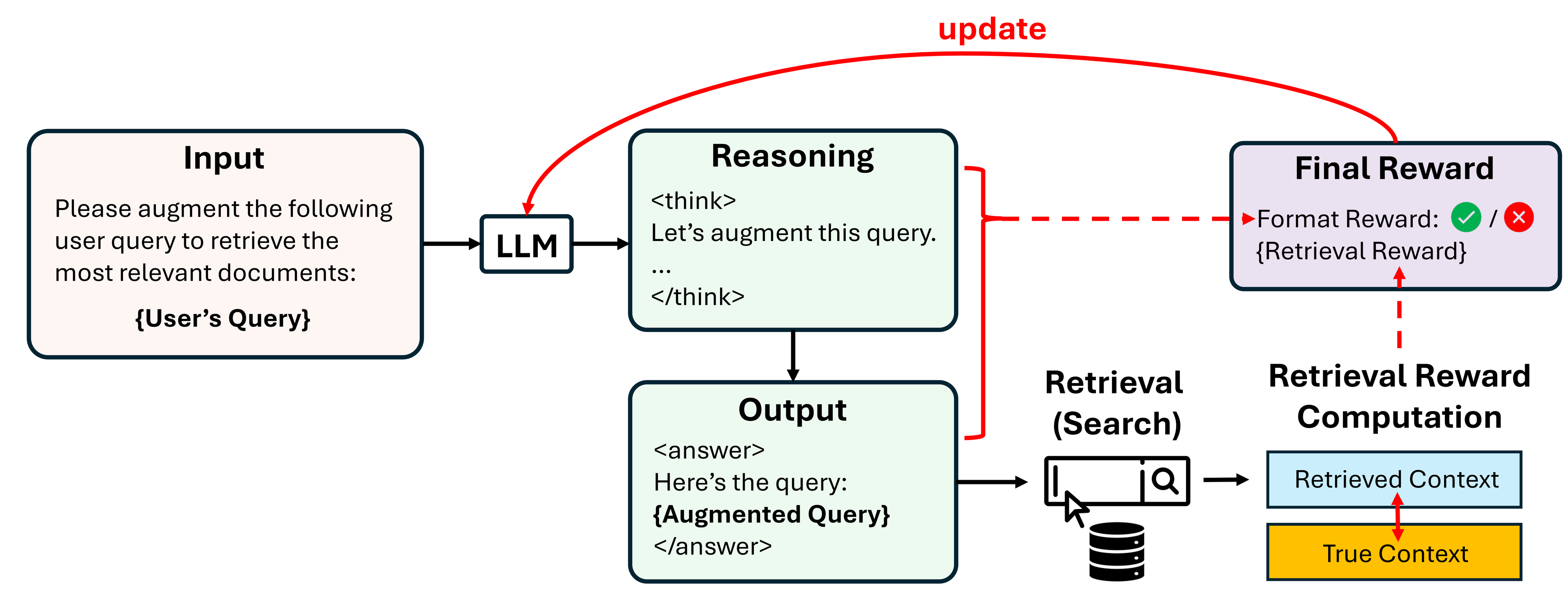
Previous Approaches (Distillation from Larger LLMs):



- Costly and highly rely on the quality of reference query (often suboptimal)

Inspired by DeepSeek-R1, we introduce **DeepRetrieval**

DeepRetrieval Framework



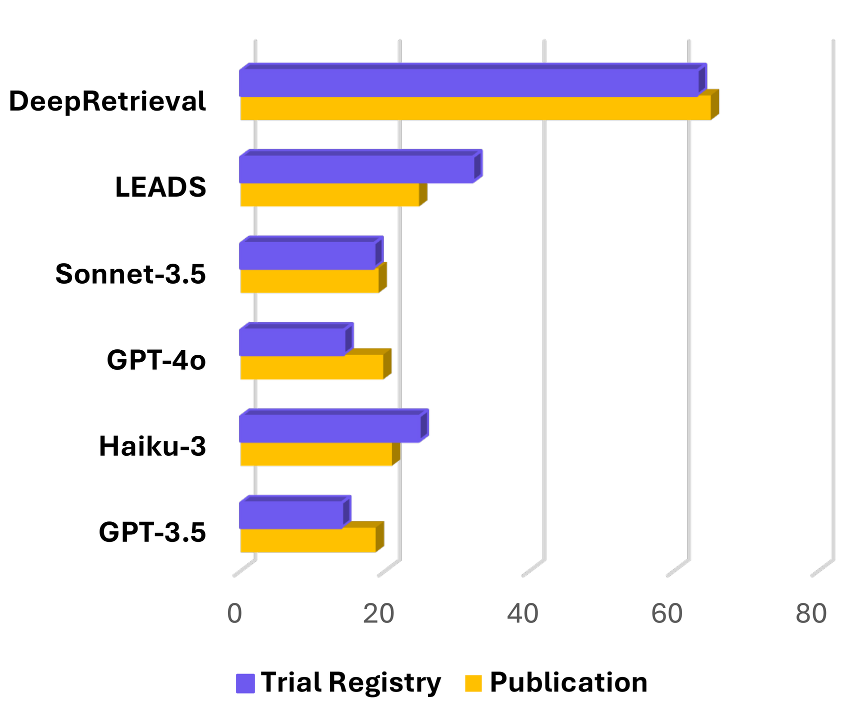
DeepRetrieval discovers optimal query patterns through direct interaction with retrieval systems

Query Generation & Retrieval:
- Input: User query enters the system
- Reasoning: Model first thinks through augmentation strategy in <think> tags
- Output: Model provides final augmented query in <answer> tags
- Retrieval: Search system executes query and retrieves documents

Reward Optimization:
- Format reward ensures adherence to required output structure
- Retrieval reward directly measures search effectiveness (recall, NDCG, etc.)

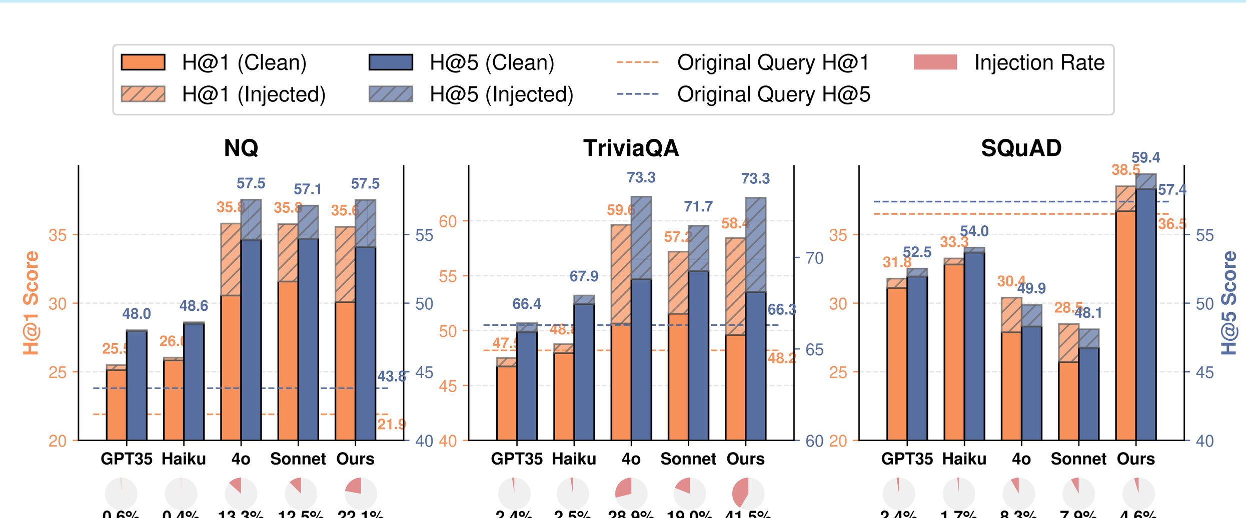
Main Results (find full tables in our paper)

Task 1: Real Search Engines



DeepRetrieval-3B's 65.07% vs. Previous SOTA (SFT)'s 24.68%
on PubMed Search API
(Measured by Recall@3K)

Task 2: Evidence-Seeking Retrieval



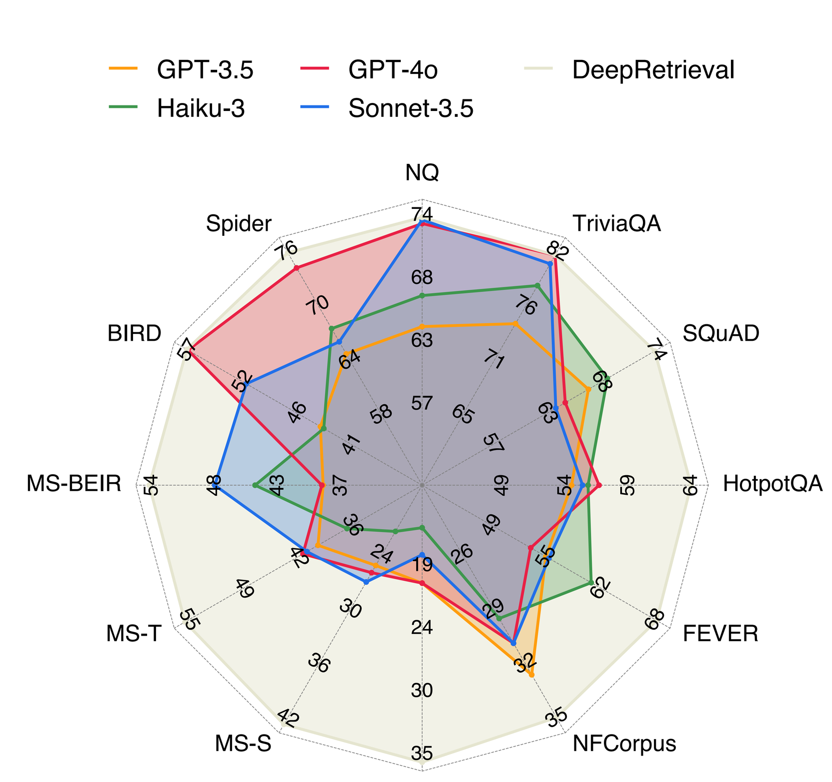
Evidence-Seeking Retrieval: Given a question, looking for the answer span in the retrieved documents. Measured by Hits@N. The shadowed barchart and piechart shows the performance gain by knowledge injection and injection ratio.

Our **DeepRetrieval-3B** achieves comparable performance to GPT-4o/Claude-3.5 on NQ and TriviaQA, and outperfroms them on SQuAD.

Task 3: SQL Search

Methods	BIRD	Spider
Zero-shot (w/ reasoning)		
GPT-3.5	44.07	64.88
GPT-4o	55.93	73.40
Claude-3-Haiku	43.81	67.44
Claude-3.5-Sonnet	50.65	66.05
Qwen2.5$_{3B\text{-}Inst}$	30.83	55.13
Qwen2.5-Coder$_{3B\text{-}Inst}$	33.57	54.45
Qwen2.5-Coder$_{7B\text{-}Inst}$	45.57	67.70
SFT		
Qwen2.5$_{3B\text{-}Inst}$	33.77	56.67
Qwen2.5-Coder$_{3B\text{-}Inst}$	39.77	58.61
Qwen2.5-Coder$_{7B\text{-}Inst}$	44.07	65.96
Ours		
DeepRetrieval$_{3B\text{-}Base}$	41.40	68.79
w/ cold start	44.00	70.33
w/o reasoning	39.57	70.24
DeepRetrieval$_{3B\text{-}Coder}$	49.02	74.85
w/ cold start	50.52	74.34
w/o reasoning	47.00	73.59
DeepRetrieval$_{7B\text{-}Coder}$	**56.00**	**76.01**

Task 4: Classic IR

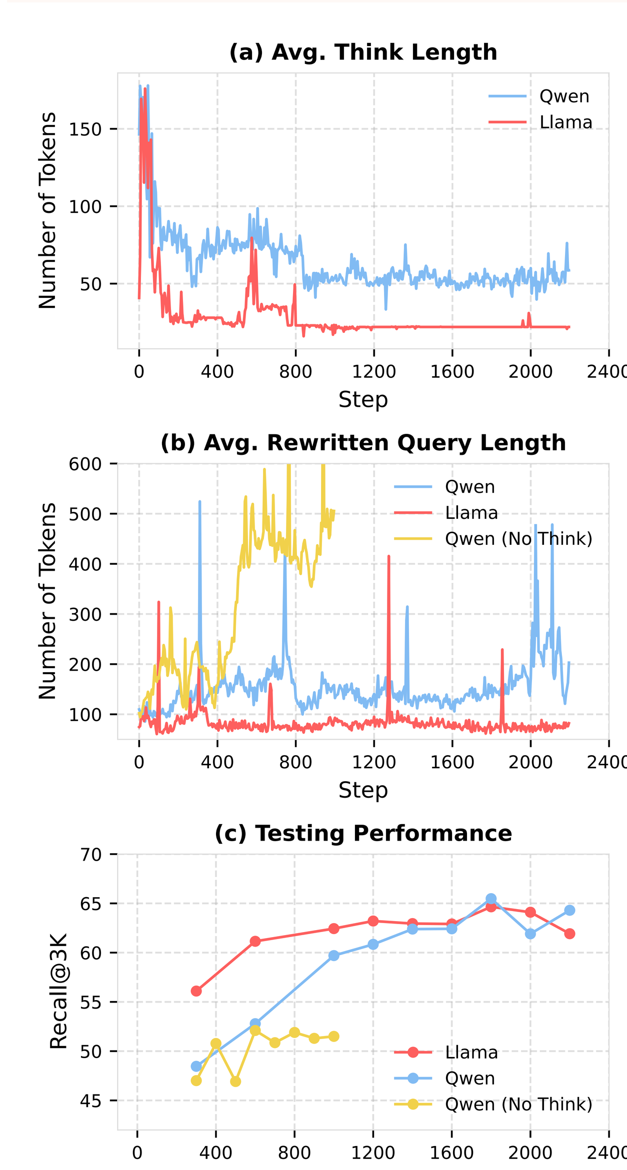


DeepRetrieval outperforms leading industry models GPT-4o and Claude-3.5-Sonnet on

1. **SQL Search (BIRD and Spider)**: Given a user query in text, do text-to-SQL generation, and execute the SQL to search DB. Measured by execution accuracy (answer exact match).
2. **Classic Sparse/Dense Text Retrieval:** Query rewriting and retrieve text from corpus using BM25 / dense retriever. Measured by NDCG@10.

Discussions & Takeaways

Think/Query Length Study



Reasoning Evolution: Unlike tasks requiring long reasoning chains, reasoning length decreases over time as models internalize effective strategies

Different Strategies leading to similar performance: Models discover distinct approaches (Qwen favors longer queries, LLaMA produces shorter ones), yet achieve comparable recall (~65%) - demonstrating multiple valid paths to high performance

Without Reasoning: Models fall into local minima of query verbosity (yellow line) with lower performance (~52% vs ~65% recall)

Key Finding: Thinking phase is crucial for exploration during training but becomes more efficient as model learns optimal patterns

Why RL >> SFT?

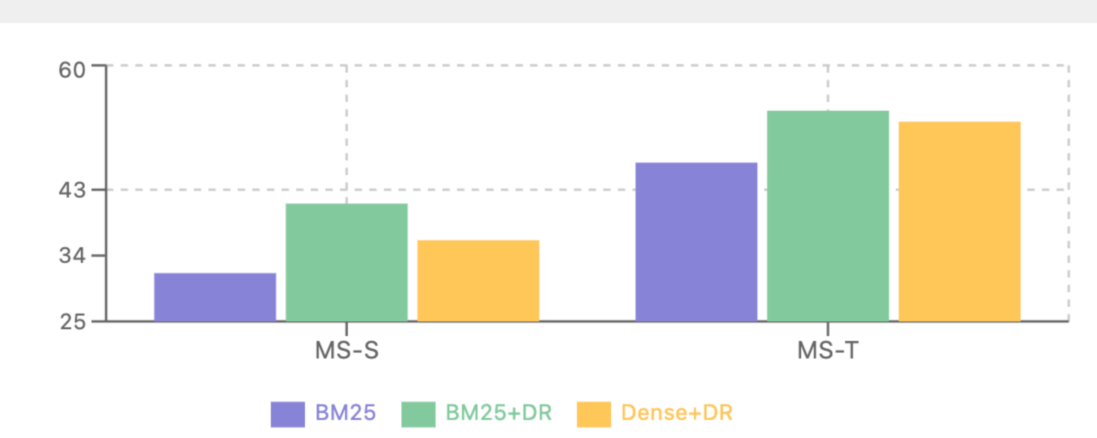
- **Direct Optimization**: RL optimizes retrieval metrics directly rather than mimicking reference queries
- **Exploration Advantage**: RL explores query space through trial-and-error, discovering patterns human experts might miss
  For example:

```
((Total Knee Arthroplasty Trial OR Total Knee
Arthroplasty Surgery) AND (Drainage OR
Antibiotics Trial OR Surgical Drainage Trial
OR Postoperative Drains Trial))
```

- **Task Adaptability**: RL performs consistently well across scenarios with varying levels of ground truth availability

They are also complementary : SFT can provide strong initialization for RL when model lacks domain capabilities (SQL coding)

BM25 Renaissance for Classic IR



34× faster with better performance

- **BM25+DeepRetrieval** combines the efficiency of sparse retrieval with performance that **matches or exceeds dense methods**.
- Our experiments show 34× faster runtime while achieving better accuracy on MS MARCO domain-specific collections.

More Questions?

Feel free to reach out Patrick Jiang (pj20@illinois.edu) if you have further questions & discussions!