DeepRetrieval: Hacking Real Search Engines and Retrievers with LLMs via RL

The First Search Agent Trained with On-Policy RL

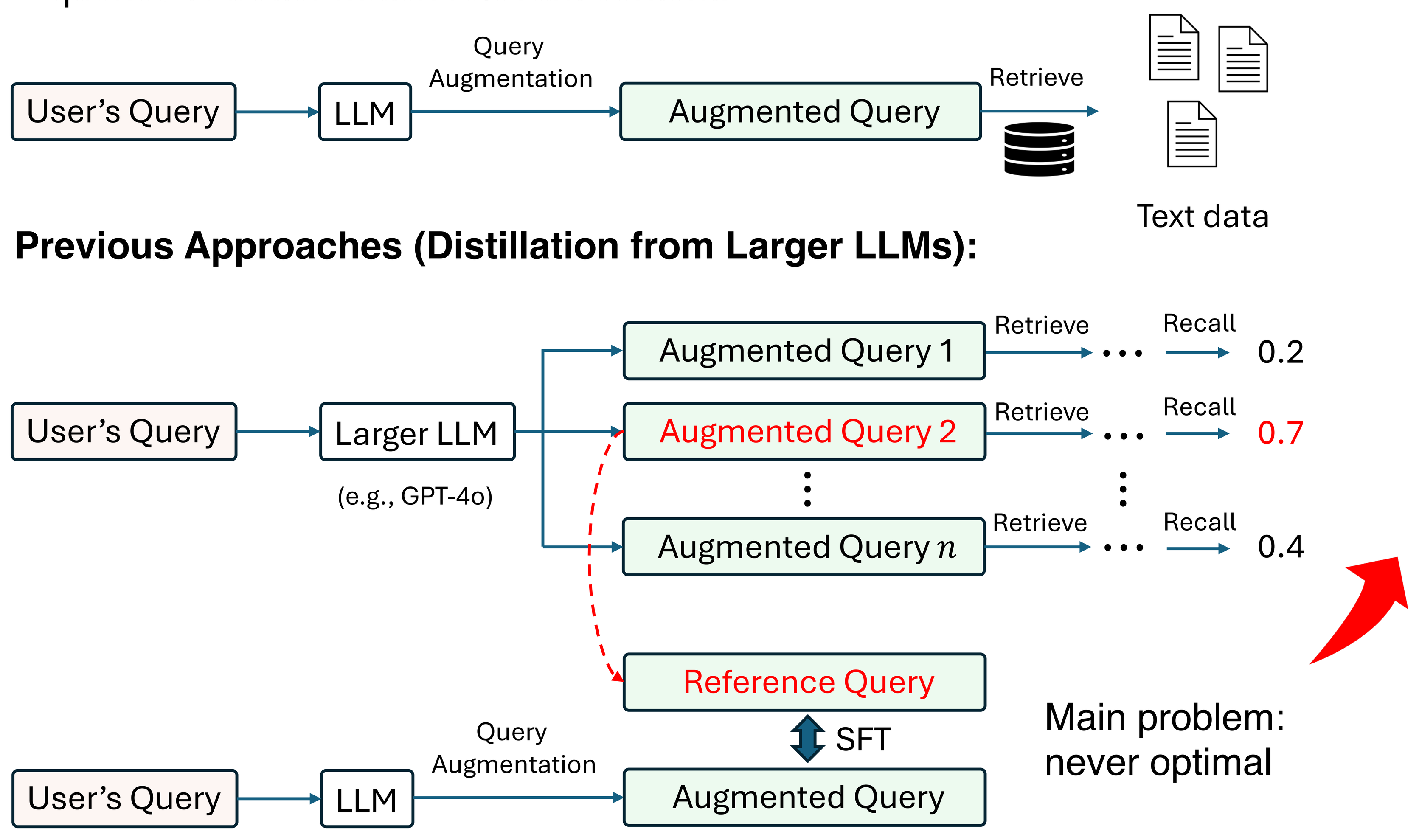
UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

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

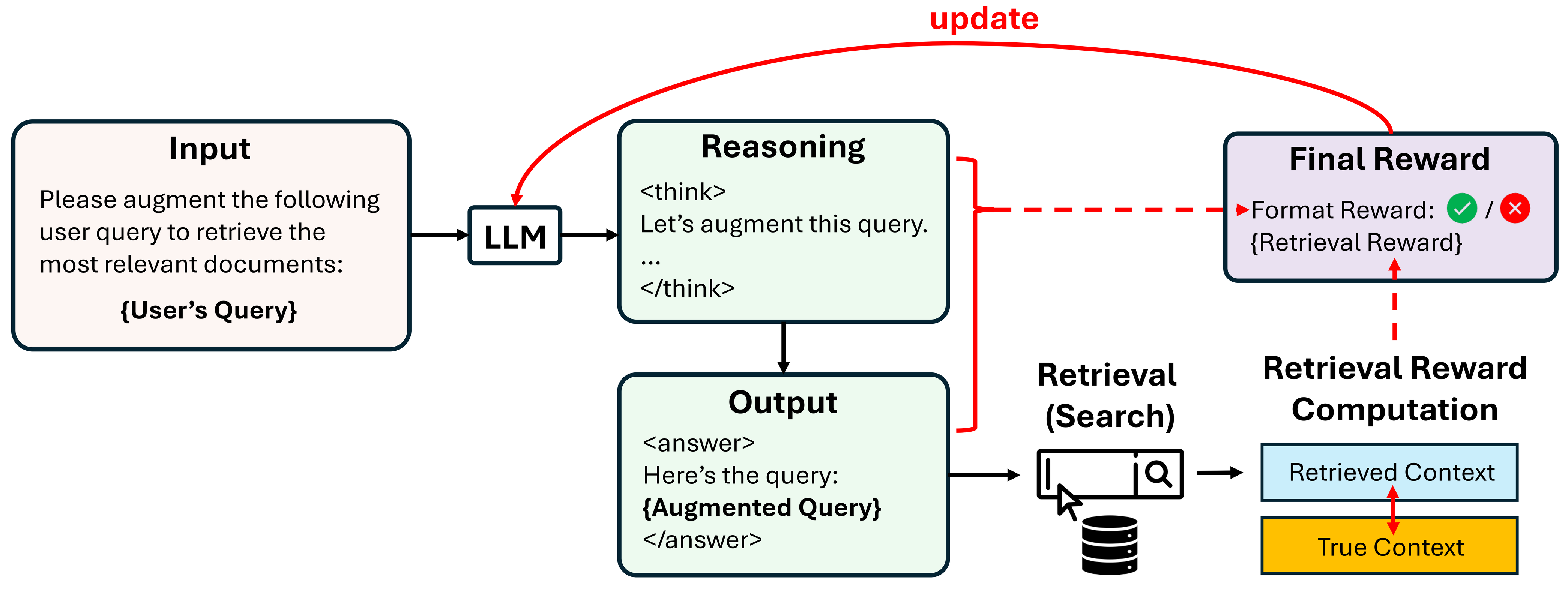
Paper Code

Background

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



DeepRetrieval Framework

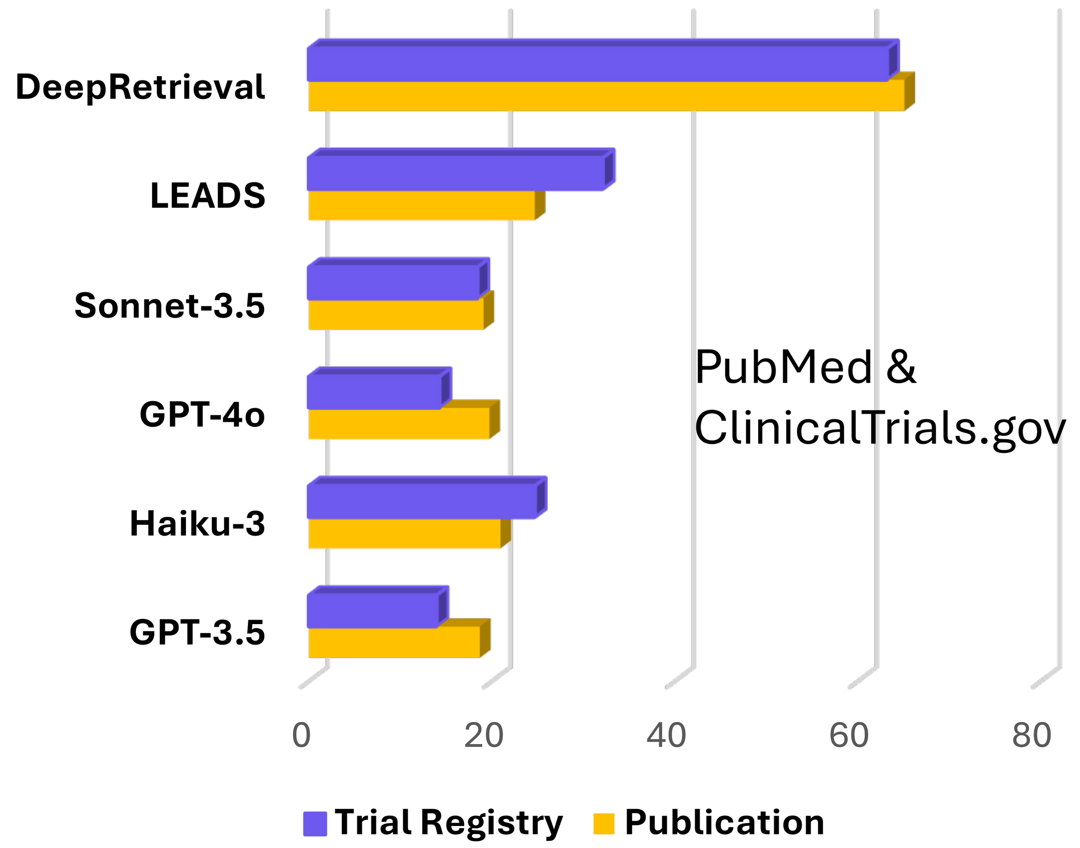


We introduce **DeepRetrieval**

DeepRetrieval discovers optimal query patterns through direct interaction with retrieval systems

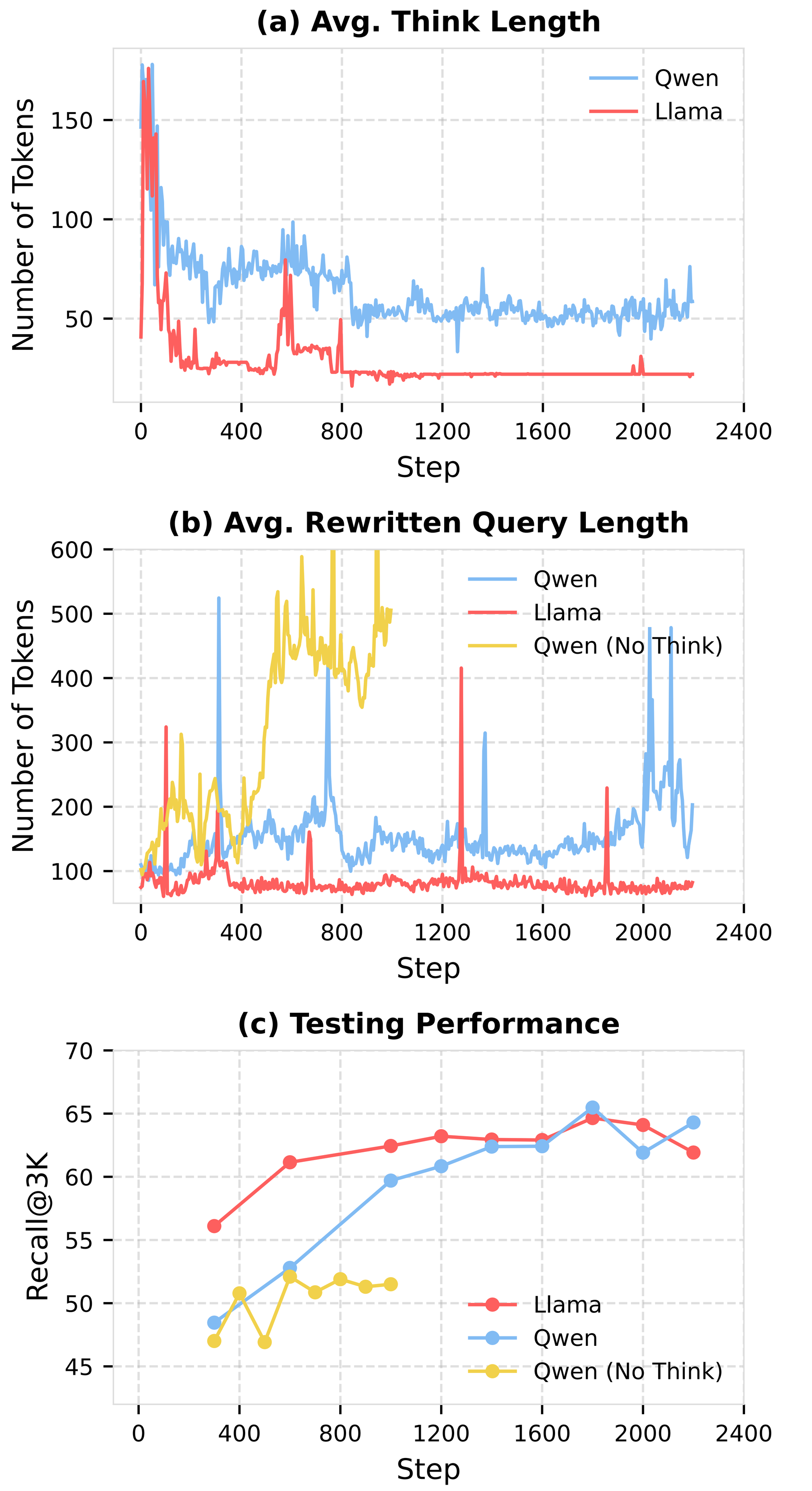
Reward Optimization:
- Format reward ensures adherence to required output structure
- Retrieval reward directly measures search effectiveness (recall, NDCG, etc.)

Task: Real Search Engines



DeepRetrieval-3B's 65.07%
vs. Previous SOTA (SFT)'s 24.68%

Think/Query Length Study



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

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

Task: (Text2)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$_{\text{3B-Inst}}$	30.83	55.13
Qwen2.5-Coder$_{\text{3B-Inst}}$	33.57	54.45
Qwen2.5-Coder$_{\text{7B-Inst}}$	45.57	67.70
SFT		
Qwen2.5$_{\text{3B-Inst}}$	33.77	56.67
Qwen2.5-Coder$_{\text{3B-Inst}}$	39.77	58.61
Qwen2.5-Coder$_{\text{7B-Inst}}$	44.07	65.96
Ours		
DeepRetrieval$_{\text{3B-Base}}$	41.40	68.79
w/ cold start	44.00	70.33
w/o reasoning	39.57	70.24
DeepRetrieval$_{\text{3B-Coder}}$	49.02	74.85
w/ cold start	50.52	74.34
w/o reasoning	47.00	73.59
DeepRetrieval$_{\text{7B-Coder}}$	**56.00**	**76.01**

Why RL >> SFT for Retrieval?

- **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 (search PubMed):

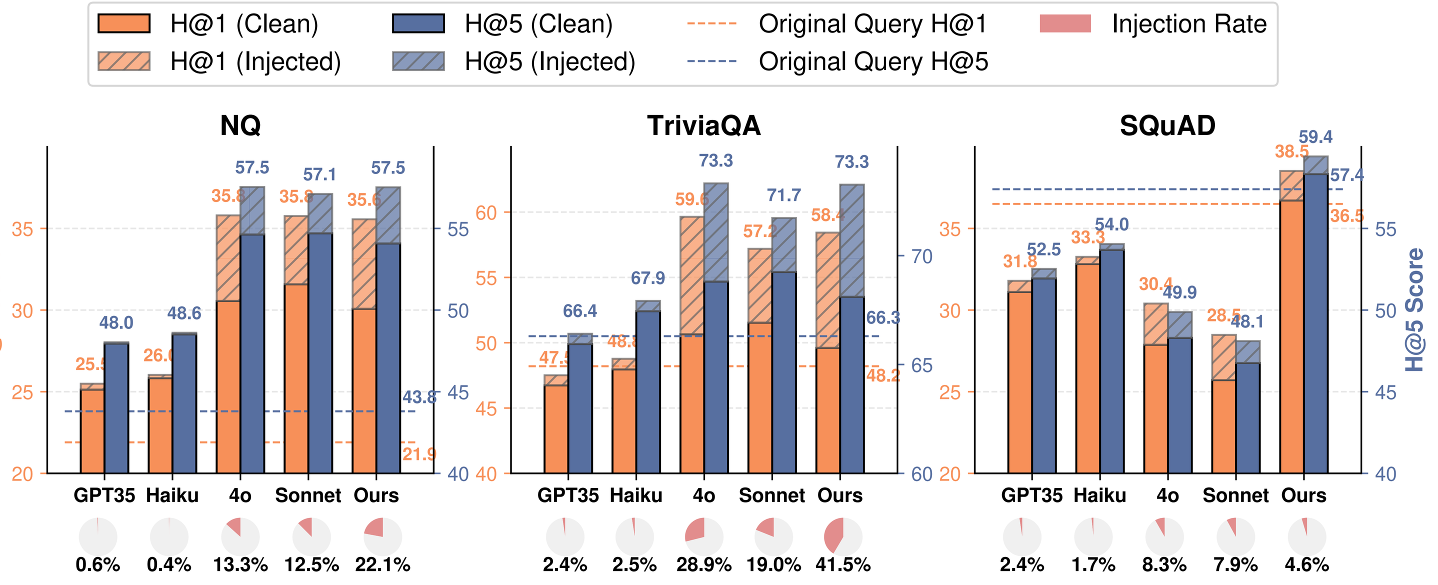
```
((Total Knee Arthroplasty Trial OR Total Knee
Arthroplasty Surgery) AND (Drainage OR
Antibiotics Trial OR Surgical Drainage Trial
OR Postoperative Drains Trial))
```

- **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 reference query is **truly golden** (see "*w/ cold start*" in "Task: SQL Search")

Task: QA Retrieval

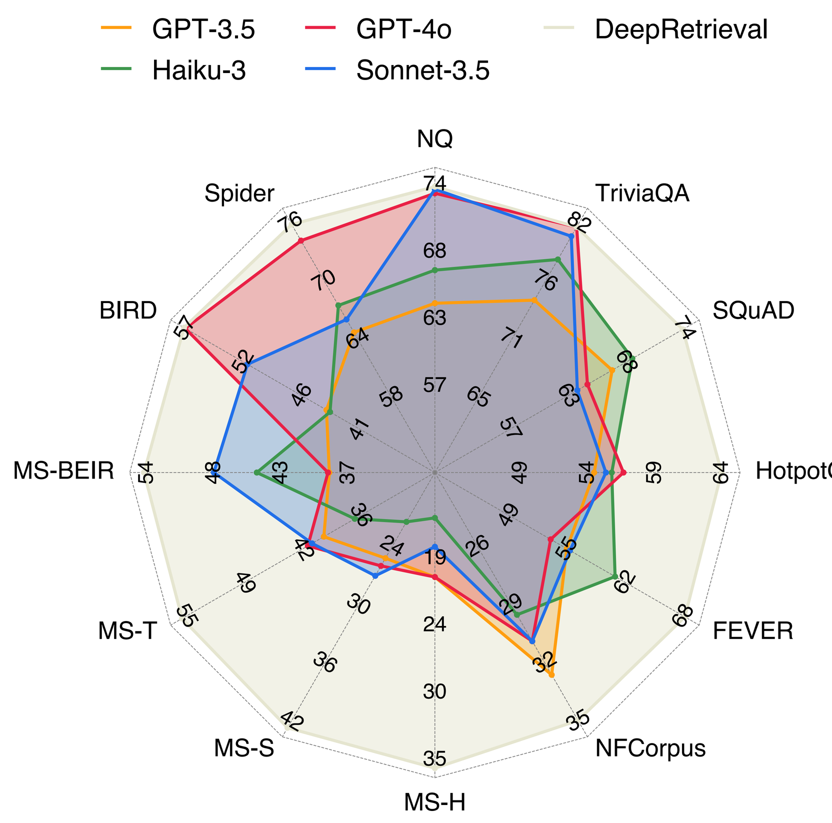
Evidence-Seeking (QA) Retrieval: Given a question, looking for the answer span in the retrieved documents. Measured by Hits@N. The shadowed barchart and piechart shows the performance gain by knowledge injection and injection ratio.



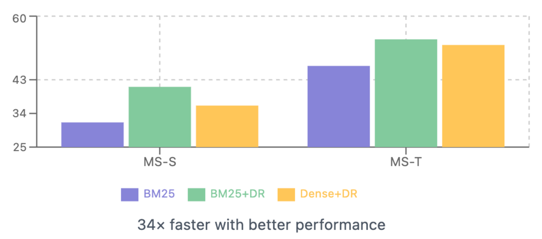
DeepRetrieval-3B achieves comparable performance to GPT-4o/Claude-3.5 on NQ and TriviaQA, and outperfroms them on SQuAD.

Task: Classic IR

Classic Sparse/Dense Text Retrieval: Query rewriting and retrieve text from corpus using BM25 / dense retriever. Metric: NDCG@10.



BM25 Renaissance for IR



- **BM25+DeepRetrieval** combines the efficiency of sparse retrieval with performance that **matches or exceeds dense methods**.