AutoBool: Reinforcement-Learned LLM for Effective Automatic Systematic Reviews Boolean Query Generation

Anonymous ACL submission

Abstract

We present **AutoBool**, a reinforcement learning (RL) framework that trains large language models (LLMs) to generate effective Boolean queries for medical systematic reviews. Boolean queries are the primary mechanism for literature retrieval in this domain and must achieve high recall while maintaining reasonable precision—a challenging balance that existing prompt-based LLM approaches often struggle to achieve. A major limitation in this space is the lack of ground-truth *best* Boolean queries for each topic, which makes supervised fine-tuning impractical. **AutoBool** addresses this challenge by leveraging RL to directly optimize query generation against retrieval performance metrics, without requiring ideal target queries. To support this effort, we create and release the largest dataset of its kind: 65 588 topics in total for training and evaluating the task of automatic Boolean query formulation.

Experiments on our new dataset and two established datasets (CLEF TAR and Seed Collection) show that AutoBool significantly outperforms zero-shot prompting and matches or exceeds the effectiveness of much larger GPT-based models (e.g., GPT-4, O3) using smaller backbones. It also approaches effectiveness of expert-authored queries while retrieving 10–16 times fewer documents. Ablation studies reveal the critical roles of model backbone, size, decoding temperature, and prompt design. Code and data are available at `https://anonymous.4open.science/r/AutoBool-B3E5/`.

1 Introduction

Systematic reviews are essential tools in evidence-based medicine, providing comprehensive and unbiased summaries of scientific knowledge across medicine and social sciences. At the heart of these reviews lies a deceptively complex task: the formulation of Boolean search queries capable of retrieving all relevant literature (high recall) without overburdening researchers with non-relevant results (high precision). A well-formulated Boolean query directly affects the cost, efficiency, and reproducibility of the review process.

Large language models (LLMs) have been explored as tools for automatically generating Boolean queries from an initial research question or topic (Wang et al., 2023, 2025; Staudinger et al., 2024). While conceptually easy to use, prompt-based LLM methods have shown major limitations—often retrieving far fewer relevant studies than expert-crafted queries (low recall), well below the thresholds typically required for systematic reviews (e.g., 10–40% recall instead of 80–90%) (Wang et al., 2025). These limitations highlight the need for new methods that go beyond zero-shot prompting and instead are optimized to generate queries based on retrieval effectiveness.

A natural alternative to prompting is supervised fine-tuning on example queries. However, this is impractical for Boolean query generation, where no single ground-truth "best" query exists. Expert-crafted queries are often inconsistent, and suboptimal (Scells and Zuccon, 2018). Further, existing datasets are too small: overall fewer than 200 training pairs, many sourced from Cochrane (Kanoulas et al., 2018; Wang et al., 2022c).[1]

We propose a reinforcement learning (RL) framework called **AutoBool** to train LLMs for Boolean query generation, enabling direct optimization for retrieval effectiveness. Rather than relying on handcrafted prompts or static templates, our model learns to balance recall and precision through feedback from real document retrieval. To support this, we construct a large-scale training dataset of 32 794 systematic reviews mined from the PubMed Central (PMC) Open Access corpus, and introduce a retrieval-grounded reward func-

[1] Cochrane does not support direct API access; prior work translates these queries into MEDLINE format for PubMed execution (Wang et al., 2025).

tion aligned with the screening goals of systematic review creation. We further release a large-scale evaluation dataset comprising 32 794 topics with high-quality relevance labels—an order of magnitude larger than prior benchmarks such as CLEF TAR and the Seed Collection,[2] along with a smaller PubTemp set designed to be free from data leakage.

Our experiments show that **AutoBool**-trained models substantially outperform prompt-based zero-shot baselines, narrow the gap with expert-crafted queries, and even exceed the performance of much larger commercial LLMs such as GPT-4o and O3 in high-recall retrieval scenarios, despite relying on significantly smaller backbones.

2 Related Work

Previous work on automated Boolean query formulation mainly followed two paradigms: The *objective method* and the *conceptual method* (Scells et al., 2020b,a). The objective method aim to replicate known included studies by expanding on initial seed set of studies using techniques such as keyword co-occurrence, term frequency analysis, or relevance feedback (Hausner et al., 2012). These approaches typically prioritize recall but often sacrifice interpretability and precision. In contrast, conceptual methods involve manually identifying key elements of the review question, such as population, intervention, and outcome, and converting them into structured Boolean expressions. While more interpretable and aligned with expert workflows, conceptual methods require substantial manual effort and domain expertise (Clark, 2013).

LLMs have recently been explored for automating Boolean query generation from topic descriptions. While they can produce syntactically valid queries, they often require multiple attempts, and their recall remains well below that of expert-authored queries (Wang et al., 2025).

These limitations highlight the need for trainable methods that move beyond static prompting. Supervised fine-tuning is not well-suited to Boolean query generation, as there is no single ground-truth “best” query per topic: expert-authored queries are manually refined, inconsistent, and often suboptimal. RL offers a more flexible alternative by optimizing models based on task-level feedback without requiring gold-standard targets. It has emerged as a promising approach for optimizing LLMs on non-differentiable objectives such as factual accuracy, preference alignment, and retrieval quality (Zhang et al., 2020; Zhuang et al., 2025; Nguyen et al., 2024). Notably, Group Relative Policy Optimization (**GRPO**), a recently introduced RL method, has shown strong performance in LLM fine-tuning scenarios, including DeepSeek-R1 (DeepSeek-AI et al., 2025; Guo et al., 2025).

3 Dataset Creation

To support both training and large-scale evaluation of systematic review Boolean query generation, we construct a new dataset based on full-text systematic reviews from the PubMed Central (PMC) Open Access (OA) subset (U.S. National Library of Medicine, 2003). This subset is license-compatible with commercial use and substantially larger than existing public benchmarks.[3]

3.1 Data Source and Extraction

We begin by extracting all articles labeled with the publication type `systematic review` from PubMed Central (PMC) Open Access (OA) subset, resulting in a total of 75 676 systematic review topics. For each topic, we parse the full PMC XML and extract the PMIDs cited in the results section. These cited references are treated as the included studies (i.e., the gold-standard relevant set) for that review. After filtering for availability of cited PMIDs, we retain 65 600 usable topics.

3.2 Benchmark Integrity

To prevent data leakage from prior evaluation sets, we manually remove any topics overlapping with the CLEF TAR or Seed Collection (Kanoulas et al., 2018; Wang et al., 2022a). This resulted in the exclusion of 12 topics, yielding a final dataset of 65 588 unique systematic reviews topics.

3.3 Temporal Train-Test Split

To prevent temporal leakage and better simulate real-world deployment, we split the dataset chronologically based on publication date:

- **Training set:** 32.794 topics published between `2000-07-06` and `2021-10-30`.
- **Test set:** 32.794 topics published between `2021-10-31` and `2025-03-01`.
- **PubTemp (PubMed Temporal) Set:** 1,000 randomly sampled topics published after `2024-11-01`.

[2] Which contain 118 and 40 topics overall respectively.

[3] Dataset will be made available on Huggingface.

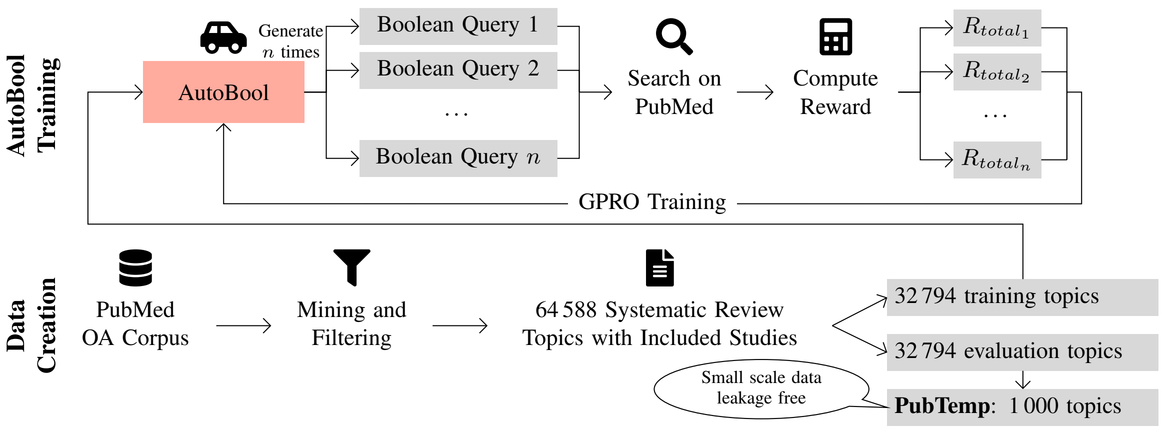

Figure 1: Overview architecture of dataset creation and AutoBool training.

This temporal split reflects real-world usage: an information specialist formulates a Boolean query to retrieve studies published up to that point. The model, trained only on earlier topics, must generalize to future, unseen ones. Chronological partitioning also supports continual evaluation, enabling future research on adapting to evolving terminology, intervention types, and publication trends.

We introduce the **PubTemp** set to enable fair, out-of-distribution evaluation by ensuring topics were unseen during LLM pretraining. Since Qwen3 (the primary model used) has an October 2024 knowledge cutoff,[4] we select topics published after November 1, 2024, to minimize overlap. To keep evaluation feasible, we randomly sample 1000 such topics to avoid the API time cost of issuing too many PubMed API queries and reducing the expense of evaluating commercial models (e.g., generating queries for 1000 topics using O3 costs around $50). The PubTemp split will be released with our dataset to support reproducibility.

4 Method

We train an LLM to generate Boolean queries using GPRO to directly optimize retrieval effectiveness as detailed in Section 2. The central challenge in applying GRPO lies in designing an appropriate reward function. In the context of systematic reviews, the effectiveness of a Boolean query can be objectively evaluated by executing the query on a document collection (e.g., PubMed) and comparing the retrieved documents against a gold-standard set of included studies. This enables computation of retrieval metrics such as recall and precision which form the basis of the reward.

4.1 Reward Design

The total reward consists of three components: formatting correctness, syntactic validity, and retrieval effectiveness.

Formatting Reward. The formatting reward R_{format} checks whether the output follows expected structural conventions (e.g., quoted terms, capitalized Boolean operators):

$$R_{\text{format}} = \begin{cases} 10 & \text{if format is correct} \\ -10 & \text{otherwise} \end{cases}$$

Validity Reward. The validity reward R_{validity} ensures that the generated Boolean query can be both syntactically parsed and successfully executed in a retrieval system:

$$R_{\text{validity}} = \begin{cases} 10 & \text{if query is valid} \\ -10 & \text{if query is invalid} \end{cases}$$

A query is considered valid if it passes two checks: (1) it must be syntactically correct according to a custom Boolean query parser that verifies structural elements such as balanced parentheses and proper use of logical operators; and (2) it must return at least one result when executed via PubMed, and returns fewer than a maximum threshold of 200 000 documents [5]. Queries that fail either check are treated as invalid and receive a penalty.

Retrieval Reward. The retrieval reward is designed to support recall-oriented Boolean query generation, reflecting the priorities of systematic reviews: retrieve as many relevant studies as possible while minimizing screening burden. It balances a direct reward for recall and a recall-modulated reward for precision.

[4] While there is no officially published knowledge cutoff for Qwen3, we confirmed via direct model queries that its knowledge extends up to October 2024.

[5] This threshold is enforced to ensure query efficiency and system responsiveness.

The reward function satisfies three core properties: (1) **recall must be prioritized**, as high recall is essential in systematic reviews (Straube et al., 2021); (2) **precision should matter more as recall increases**, since reducing irrelevant results becomes valuable once coverage improves; and (3) **early precision gains should be emphasized**, as small improvements from very low precision levels bring significant practical benefit.

To implement this, we apply two mechanisms: (1) weighting precision by r^{α} to reduce its impact at low recall; and (2) applying a logarithmic transformation to increase sensitivity when precision is low. The resulting reward is:

$$F(r,p) = \underbrace{M \cdot r}_{\text{recall reward}} + \underbrace{M \cdot r^{\alpha} \cdot \log_{1+s}(1 + s \cdot p)}_{\text{precision reward}}$$

where r is recall, p is precision, $s = 100$ is a smoothing constant, $\alpha \geq 0$ controls precision weighting, and M is a global scaling factor.

The complete retrieval reward is defined as:

$$R_{\text{retrieval}}(r,p,|D|) = \begin{cases} -20 & \text{if } |D| = 0 \\ -5 & \text{if } r = 0 \wedge p = 0 \\ F(r,p) & \text{otherwise} \end{cases}$$

The choice of α significantly influences retrieval behavior. When $\alpha = 0.5$, the reward is *weakly recall-oriented*, allowing precision to contribute earlier in the learning process. At $\alpha = 1$, the behavior becomes *moderately recall-oriented*, offering a balanced trade-off between recall and precision. Setting $\alpha = 2$ results in a *strongly recall-oriented* reward, where precision only meaningfully contributes once high recall has been achieved.

Total Reward. The final reward combines all components: $R_{\text{total}} = R_{\text{format}} + R_{\text{validity}} + R_{\text{retrieval}}$

4.2 Training Procedure

We fine-tune a pretrained LLM using policy optimization to maximize R_{total}. At each training step, the model is prompted with a systematic review topic and generates a Boolean query. This query is executed on a document collection, and its reward is computed. Gradients from GRPO are used to update the model, encouraging generation of valid, well-structured, and high-recall queries over time.

4.3 Prompting Strategies

We investigate four prompting strategies, each designed to elicit different forms of reasoning. Two prompts—No Reasoning **(N.R)** and Free-text Reasoning **(R)**—provide essential instructions about what a Boolean query is, its components, requirements and how to use different search fields.

Table 1: Prompting strategies for Boolean query generation. Full templates are in Appendix A.1.

Prompt Type and Description
No Reasoning (N.R): Direct query generation with minimal explanation or structure.
Free-text Reasoning (R): Includes a natural language explanation before query generation.
Conceptual Reasoning (R-con): Uses structured decomposition (Population, Intervention, Outcome) to scaffold the query.
Objective Reasoning (R-obj): Simulates a relevant abstract and extracts key terms empirically.

The other two—Conceptual Reasoning **(R-con)** and Objective Reasoning **(R-obj)**—are inspired by established paradigms in query formulation. These prompts offer more structured, step-by-step guidance on how to decompose the topic and construct the query (Scells et al., 2020a,b).

Table 1 summarizes these strategies. Full prompt templates are provided in Appendix A.1.

5 Experimental Setup

5.1 Retrieval and Evaluation Protocol

For all datasets, we use the PubMed Entrez API to execute the generated Boolean queries and retrieve candidate documents by matching on PMIDs (Sayers, 2010). This simulates a realistic literature search workflow, allows us to evaluate query effectiveness in a practical retrieval setting, and follows standard methodology from previous work (Wang et al., 2023, 2025). We apply the same query validity check protocol described in Section 4.1 to detect and reject malformed queries. Queries that fail this check are considered invalid, and the model is prompted to regenerate until a valid query is produced, with a maximum of 10 attempts [6]. This validation process aligns with established evaluation practices (Wang et al., 2025).

We evaluate effectiveness against gold-standard included studies using metrics adopted in earlier research (Wang et al., 2025). The **primary evaluation metrics** are recall, F_3, and the percentage of queries achieving recall above 80% and 90%. These reflect the high-recall requirements of sys-

[6] We cap regenerations at 10 to avoid excessive API usage and model inference.

tematic reviews, where omissions of relevant studies can critically undermine review quality. As **secondary metrics**, we report precision, the average number of documents retrieved (to measure screening effort), the average number of regenerations per query, and the success rate under 10 attempts—to capture robustness and generation stability beyond recall-focused evaluation.

5.2 Model Variants

Our primary experiments are conducted using models from the Qwen3 family (Yang et al., 2025). Unless otherwise specified, all trained AutoBool Models are based on `Qwen3-4B`, which serves as the default backbone throughout our main results. These models are fine-tuned using our reward-driven training framework (Section 4.1) to optimize for systematic review retrieval effectiveness. GRPO is applied to guide query generation behavior based on retrieval performance signals. At inference time, we use the same prompt as during training and fix the decoding temperature to 0.6, following the Qwen3 recommendations (Yang et al., 2025).

5.3 Evaluation Datasets

We evaluate our models on three datasets: the **PubTemp set**, and two established benchmarks: the **CLEF TAR** collection (Kanoulas et al., 2018) and the **Seed Collection** (Wang et al., 2022a).

We follow prior work (Wang et al., 2025) in using the CLEF TAR 2017 and 2018 subsets (72 topics total) and the Seed Collection (40 topics). Each topic is defined by experts and paired with a manually curated set of relevant studies. These benchmarks have been widely adopted for evaluating automatic Boolean query generation in the context of systematic review automation (MacFarlane et al., 2022; Kusa et al., 2023; Wang et al., 2022b; Stevenson and Bin-Hezam, 2023; Lee and Sun, 2018).

5.4 Training Parameters

All models are trained using the GRPO RL algorithm with a retrieval-based reward. We adopt LoRA-based parameter-efficient fine-tuning and use the vLLM backend in colocated mode. For each prompt, the model generates 4 completions, which are evaluated to compute reward scores. We use an effective batch size of 16 via gradient accumulation. Prompt and completion length limits are 768/1024 tokens for non-reasoning prompts and 1024/3072 for reasoning-based prompts. Unless otherwise noted, we set the training temperature to 1.2; its effect on performance is analyzed in Section 7. Full training hyperparameters are listed in Appendix 4.

6 Results

Table 2 summarizes the evaluation results on the PubTemp set. We observe that reinforcement learning substantially improves Boolean query generation performance, especially in recall-critical settings like systematic reviews.

6.1 Effectiveness of Reinforcement Learning

Compared to zero-shot prompting baselines using the same base model (Qwen3-4B), all RL trained AutoBool models achieve substantially higher effectiveness across all prompt types and primary evaluation metrics. In particular, recall improves dramatically: top-performing models achieve an average recall of 0.70, with approximatly 35% of queries retrieving more than 90% of relevant documents. In contrast, zero-shot models achieve at most 0.35 average recall, with only 6.5% of queries exceeding the 90% recall threshold.

On secondary metrics, AutoBool models improve precision under the `N.R` and `R-obj` prompts, but slightly reduce precision under `R` and `R-con`. All trained models retrieve more documents than their zero-shot counterparts: an expected tradeoff when optimizing for recall. However, this increase in retrieved set size remains reasonable (well under 1000 on average), and does not significantly increase screening burden compared to the gains in comprehensiveness.

Training also improves generation stability. AutoBool models exhibit consistently higher success rates (near 100%) and require fewer regenerations than zero-shot baselines, indicating more reliable formatting and syntactic validity—driven by our structured reward components.

Comparison with Larger GPT-Based Models. Compared to significantly larger GPT-based models (GPT-4O and O3) [7], AutoBool—despite using a much smaller model—achieves higher recall and a higher percentage of queries exceeding high-recall thresholds (e.g., 80% and 90%). Regeneration and success rates are also comparable across models. While AutoBool slightly lags behind in precision and F_3, the average number of retrieved

[7] Number of Parameters In Trillions

Table 2: Effectiveness of LLM-generated Boolean queries on the PubTemp set. **Bold** indicates the best result for each model within a setting; Underlined indicates the overall best across all models. (O3 model has no N.R capability as reasoning was enabled by default using API.)

Setting	Model	Prompt	Recall	F3	Recall >80%	Recall >90%	Precision	Avg Retrieved	Avg Regen	%Success
Zero-shot	GPT-4O	N.R	0.3591	**0.1758**	12.40	7.10	**0.1074**	**251.32**	1.15	99.60
	GPT-4O	R	0.3937	0.1653	14.30	9.00	0.0913	326.20	1.09	99.80
	GPT-4O	R-con	**0.4387**	0.1491	**15.90**	**9.40**	0.0642	440.45	**1.04**	**99.90**
	GPT-4O	R-obj	0.2530	0.0988	5.30	3.10	0.0742	359.00	1.20	99.80
	O3	R	**0.6868**	0.2039	**44.80**	**31.80**	0.0611	551.54	1.69	98.20
	O3	R-con	0.6483	**0.2106**	38.60	26.20	**0.0690**	**499.48**	**1.19**	99.70
	O3	R-obj	0.5153	0.1293	22.70	14.70	0.0510	603.10	1.21	**100.00**
	Qwen3-4B	N.R	0.0098	0.0074	0.00	0.00	0.0175	**64.72**	7.93	25.60
	Qwen3-4B	R	0.0681	0.0429	0.70	0.60	**0.0640**	105.13	3.86	84.40
	Qwen3-4B	R-con	**0.3458**	**0.0824**	**11.60**	**6.50**	0.0402	514.11	**1.29**	**99.60**
	Qwen3-4B	R-obj	0.0676	0.0212	1.40	1.10	0.0233	306.53	2.89	93.80
AutoBool ($\alpha = 0.5$) Weak R.O	Qwen3-4B	N.R	**0.6791**	**0.1386**	**42.70**	**30.80**	**0.0392**	677.31	1.12	98.80
	Qwen3-4B	R	0.6112	0.1223	33.60	21.70	0.0345	678.80	**1.04**	99.60
	Qwen3-4B	R-con	0.5202	0.1052	22.80	14.30	0.0388	**654.57**	1.11	**99.80**
	Qwen3-4B	R-obj	0.6495	0.0987	39.10	25.50	0.0263	743.89	1.10	99.40
AutoBool ($\alpha = 1$) Mod R.O	Qwen3-4B	N.R	**0.7036**	0.1195	**47.10**	**32.30**	0.0300	732.49	1.17	98.40
	Qwen3-4B	R	0.5453	**0.1346**	27.90	16.50	**0.0496**	**586.01**	1.06	99.70
	Qwen3-4B	R-con	0.5262	0.1066	26.30	16.80	0.0372	636.19	1.06	**100.00**
	Qwen3-4B	R-obj	0.6540	0.1094	39.70	26.70	0.0293	738.34	**1.04**	99.90
AutoBool ($\alpha = 2$) Heavy R.O	Qwen3-4B	N.R	**0.6948**	0.1209	**45.30**	30.80	0.0306	724.15	1.11	98.90
	Qwen3-4B	R	0.5602	**0.1344**	30.00	18.40	**0.0472**	**588.21**	1.07	99.60
	Qwen3-4B	R-con	0.5911	0.0887	31.40	20.40	0.0281	739.69	1.09	**99.80**
	Qwen3-4B	R-obj	0.6878	0.1024	44.80	**31.30**	0.0245	773.01	**1.06**	99.70

documents remains within a practical range (typically 586–773, compared to 500–603 for O3 and 251–440 for GPT-4O) indicating only a modest increase in screening effort. These results highlight the effectiveness of retrieval-driven RL training in generating high-recall Boolean queries, even under constrained model capacity.

Effect of Prompt Type. Under zero-shot settings, the `R-con` prompt generally yields the highest recall and F_3 scores (except for F_3 in GPT-4o and recall in O3), this suggests that structured reasoning aids query formulation when no retrieval feedback is available. However, this advantage diminishes after training. For trained AutoBool models, the `No Reasoning` prompt consistently achieves the best recall and high-recall threshold performance across all settings, while reasoning-based prompts tend to yield higher precision.

We hypothesize two complementary explanations for these trends. First, RL enables the model to internalize task-specific structure and discover its own optimized generation strategy, which may diverge from human-designed decomposition frameworks. Second, Boolean queries are inherently interpretable and self-contained: their logic is fully encoded through syntax and operators. As such, requiring the model to verbalize intermediate reasoning may introduce unnecessary constraints or verbosity once it has learned to generate effective queries end-to-end.

Nonetheless, reasoning-based prompts consistently yield higher success rates and require fewer regenerations than `No Reasoning`, even after training. This suggests that intermediate reasoning may still offer robustness benefits in more difficult systematic review topics, where generating a valid and well-formed Boolean query is especially challenging. In these cases, explicit reasoning may help the model maintain syntactic and semantic integrity under ambiguity or complexity.

Which α Value Should Be Used? We analyze the effect of the α parameter in the retrieval reward, which adjusts the emphasis on recall over precision. In structured reasoning-based prompts (`R-con`, `R-obj`), increasing α consistently improves recall, as intended. In contrast, results are more mixed for less-structured prompts: $\alpha = 1$ yields the highest recall with `N.R`, while $\alpha = 0.5$ performs best with `R`. For F_3, $\alpha = 1$ generally achieves the best performance across prompts, except in `N.R`, where

$\alpha = 0.5$ outperforms. Overall, $\alpha = 1$ offers the most balanced trade-off between recall and precision, making it a strong default. Its effects are also more stable in structured prompts, where recall improves more predictably with higher α.

6.1.1 Generalization on Existing Datasets

To assess generalization, we evaluate our trained models on two established systematic review benchmarks: CLEF TAR and the Seed Collection (Table 5 and Table 6 in Appendix). Both datasets are relatively small and publicly available, raising potential concerns around bias and data leakage.

CLEF TAR. AutoBool generalizes well, obtaining substantially higher recall and recall-threshold metrics than its zero-shot counterparts. It is also more effective than larger models like GPT-4O and O3 in recall. With $\alpha = 1$ and the `No Reasoning` prompt, AutoBool almost matches the recall of expert-crafted queries (within 1%) while retrieving 17 times fewer documents; yielding improved F_3, higher precision, and reduced screening effort. It is also more effective than the O1 model from Wang et al. (2025) in both recall and F_3, highlighting the benefits of RL optimization.

Seed Collection. A similar pattern holds. While AutoBool underperforms expert-written queries, it significantly outperforms zero-shot baselines and the O1 model from Wang et al. (2025). These results demonstrate that retrieval-aware training enables robust Boolean query generation—even when trained on a single corpus.

7 Ablation Studies

We conduct ablation experiments to assess the impact of model size, training-time temperature, and backbone choice on retrieval performance under our reinforcement learning framework. All models are trained on the same data and reward function, with $\alpha = 1$ unless otherwise specified.

7.1 Effect of Backbone Size

To understand the impact of backbone model size on retrieval performance, we evaluate Qwen3 models at four parameter sizes (1.7B, 4B, 8B, and 14B), each trained with the same reinforcement learning setup. As shown in Figure 2, increasing model size leads to a consistent improvement in F_3, reflecting better overall balance between recall and precision. However, this comes with a trade-off: recall and recall-threshold metrics (Recall > 80%, Recall > 90%) tend to decrease slightly as model size increases. This suggests that larger models may learn to generate more screening-efficient queries, retrieving fewer documents with higher precision, but at the cost of slightly missing additional relevant studies.

Table 3: Impact of Backbone model on the effectiveness of Boolean query generation across prompt types on the PubMed Temporal-Cutoff set.

Model	Prompt	Recall	F3	Recall >80%	Recall >90%
Qwen3-8B	N.R	0.7116	0.1304	47.90	32.80
	R	0.4716	0.1360	20.80	13.20
	R-con	0.4928	0.1167	22.50	14.40
	R-obj	0.5097	**0.1376**	19.50	12.30
Llama3.1-8B	N.R	**0.7380**	0.1035	53.00	38.00
	R	0.7375	0.0999	**54.20**	**39.70**
	R-con	0.7165	0.1006	48.60	34.00
	R-obj	0.7291	0.1075	51.40	36.60

These findings highlight the importance of aligning model scale with task priorities. For high-recall applications like systematic reviews, smaller models (e.g., 4B) may be preferable due to their stronger recall performance. In contrast, larger models (e.g., 8B or 14B) may be more appropriate when minimizing screening effort is critical and slight recall reductions are acceptable.

7.2 Impact of Temperature

At training time, the generation temperature controls sampling diversity: higher values promote more varied outputs, while lower values encourage more deterministic decoding during training for the same topic. To assess its effect on retrieval effectiveness, we evaluate AutoBool models at three generation temperatures: 0.6, 0.9, and 1.2. As shown in Figure 3, increasing temperature consistently improves all primary metrics. This indicates that more diverse generations help the model explore effective query formulations, resulting in broader coverage and higher-quality retrieval.

7.3 Effect of Backbone Model

To examine how the backbone model affects performance, we compare two similarly sized models—Qwen3-8B and LLaMA3.1-8B—trained with identical reinforcement learning procedures across the same prompts. As shown in Table 3, LLaMA3.1-8B consistently outperforms Qwen3-8B across all primary recall metrics. Notably,

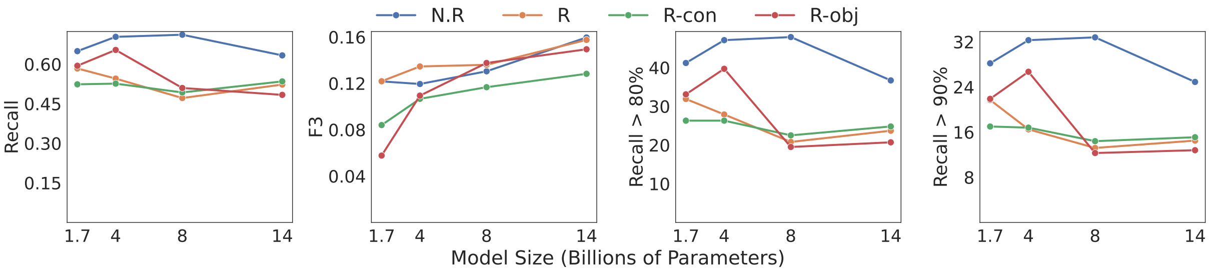

Figure 2: Effect of model size on the effectiveness of Boolean query generation across prompt types on the PubMed Temporal-Cutoff set, all result based on Qwen3 based Models.

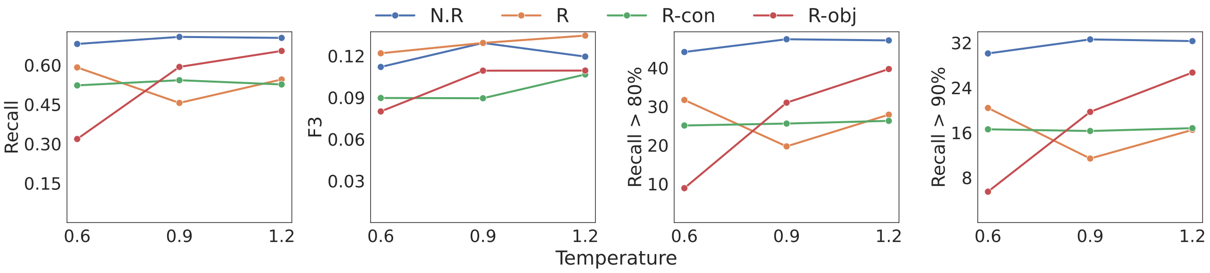

Figure 3: Effect of reinforcement learning training temperature value on the effectiveness of Boolean query generation across prompt types on the PubMed Temporal-Cutoff set, all result based on Qwen3-4B Model.

LLaMA3.1-8B achieves the highest recall across all prompt types, with especially strong results under the `N.R` prompt (0.7380 recall, 53.00% Recall > 80%) and `R` prompt (0.7375 recall, 54.20% Recall > 80%). In contrast, Qwen3-8B yields generally higher F_3, suggesting it may be more efficient in reducing screening effort.

Consistent with findings on Qwen models, LLaMA3.1 also performs best under the `No Reasoning` prompt, indicating that post-training, simpler prompting leads to more effective generation. Reasoning-based prompts tend to degrade recall performance, though they may still offer benefits in terms of robustness or interpretability.

Overall, these results highlight that backbone differences among decoder-only LLMs can significantly influence learning dynamics under reinforcement optimization, particularly in how recall and efficiency are balanced during query generation.

8 Conclusion

We present **AutoBool**, a reinforcement learning framework for training language models to generate high-quality Boolean queries for systematic reviews. We also release PubTemp: a real-world dataset of reviews significantly larger than existing training and evaluation resources.

AutoBool provides scalable training by optimising directly for retrieval effectiveness using generation preference. This overcomes a major limitation of supervised fine-tuning: the need for ground-truth Boolean queries, which not available in large quantities for the task of systematic reviews.

AutoBool substantially improves recall and robustness over zero-shot method with the same backbone, and matches or exceeds much larger models (GPT-4o and O3). Reinforcement learning enhances both retrieval effectiveness and generation stability: trained models consistently achieve higher recall, broader recall-threshold coverage, and stronger success rates. These improvements generalize to external benchmarks like CLEF and the Seed Collection, highlighting the transferability of retrieval-aware optimization.

Ablation studies reveal several key insights. Larger models improve F_3 but can slightly reduce recall. Backbone choice matters: LLaMA outperforms Qwen at the same scale. Higher decoding temperatures improve retrieval, likely by increasing query diversity. Notably, `No Reasoning` prompts consistently yield the best performance after training, though reasoning-based prompts still help in more challenging cases.

Together, these findings establish AutoBool as a scalable, flexible and high-performing solution for automated Boolean query generation, balancing comprehensiveness, efficiency, and reliability for evidence-based search.

9 Limitations

While our findings demonstrate the effectiveness of AutoBool in generating high-recall Boolean queries with relatively small language models, several limitations remain.

First, we are currently limited to fine-tuning open-source models with moderate sizes (e.g., up to 14B parameters). Although AutoBool outperforms much larger commercial LLMs (e.g., GPT-4o, O3) in recall, prior work suggests that larger models may better support reasoning-based prompts and produce higher-quality queries. Due to GPU memory constraints, we are unable to fine-tune such models, limiting our ability to explore how retrieval-aware training scales with model size—particularly in prompt-sensitive settings. Second, while LLaMA3.1 models achieved higher retrieval effectiveness in our experiments, their RL training was significantly less stable than Qwen3. We observed abrupt collapses in average reward during training that often did not recover, preventing reliable replication and leading us to focus on Qwen-based models for core experiments. Third, as with most RL-fine-tuned LLMs, AutoBool exhibits stochastic behavior during training due to non-deterministic generation and moderate-temperature decoding. This can introduce minor variance across runs and affect reproducibility at the query level, though overall performance trends remain consistent.

Future work could improve training stability for architectures like LLaMA and investigate the root causes of instability. Access to larger and more robust open-source models may further enhance AutoBool's effectiveness and generalizability.

A Appendix

A.1 Prompt Template

We design four prompt templates to investigate how different reasoning styles influence Boolean query generation performance:

- **No Reasoning (N.R):** A simple prompt that directly asks the model to generate a Boolean query from a review topic without explanation. (See Table 4)
- **Free-text Reasoning (R):** Allows the model to reason freely before producing a final query, enabling unstructured decomposition. (See Table 5)
- **Conceptual Reasoning (R-con):** Guides the model to map the topic into structured elements like Population, Intervention, and Outcome before forming the query. (See Table 6)
- **Objective Reasoning (R-obj):** Encourages the model to extract explicit inclusion criteria and convert them into Boolean syntax. (See Table 7)

These templates serve as both zero-shot prompting strategies and scaffolds for reinforcement learning. Their comparative impact is analyzed throughout our experiments.

A.2 Model Tunning

Parameter	Value
Adapter type	LoRA
LoRA rank (r)	16
LoRA alpha	32
LoRA dropout	0.05
Quantization	bf16
Attn implementation	flash-attention-2
Effective batch size	16
Learning rate	1e-5
Generation temperature	0.6
# Generations / prompt	4
Prompt length (non-reasoning)	768 / 1024 tokens
Prompt length (reasoning-based)	1024 / 3072 tokens
Reward functions	Format, Validity, Retrieval
Inference engine	vLLM (colocate mode)
Optimizer backend	DeepSpeed
# Epochs	1

Table 4: Training hyperparameters used for GRPO-based Boolean query generation.

A.2.1 Result on CLEF and Seed

We report detailed evaluation results for AutoBool on two widely used external benchmarks for Boolean query generation: CLEF TAR (Table 5) and the Seed Collection (Table 6). Both tables compare AutoBool against zero-shot prompting baselines using the same underlying model (Qwen3-4B), commercial LLMs (GPT-4O, O3), and manually written expert queries when available. These benchmarks provide insights into AutoBool's generalization ability when deployed beyond its training distribution.

CLEF TAR. AutoBool demonstrates strong generalization across all primary metrics. It substantially improves recall over zero-shot baselines and also outperforms commercial LLMs such as GPT-4O and O3 on recall and high-recall coverage (Recall > 80%, Recall > 90%). Notably, when using the `No Reasoning` prompt and $\alpha = 1$, AutoBool

System Message

You are an expert systematic review information specialist.
You are tasked to formulate a systematic review Boolean query in response to a research topic. The final Boolean query must be enclosed within <answer> </answer> tags. Do not include any explanation or reasoning.

User Message

You are given a systematic review research topic, with the topic title "topic".
Your task is to formulate a highly effective Boolean query in MEDLINE format for PubMed.
The query should balance **high recall** (capturing all relevant studies) with **reasonable precision** (avoiding irrelevant results):
- Use both free-text terms and MeSH terms (e.g., chronic pain[tiab], Pain[mh]).
- **Do not wrap terms or phrases in double quotes**, as this disables automatic term mapping (ATM).
- Combine synonyms or related terms within a concept using OR.
- Combine different concepts using AND.
- Use wildcards (*) to capture word variants (e.g., vaccin* -> vaccine, vaccination):
- Terms must have >= 4 characters before the * (e.g., colo*)
- Wildcards work with field tags (e.g., breastfeed*[tiab]).
- Field tags limit the search to specific fields and disable ATM.
- Do not include date limits.
- Tag term using term field (e.g., covid-19[ti] vaccine[ti] children[ti]) when needed.

Only use the following allowed field tags:
Title: [ti], Abstract: [ab], Title/Abstract: [tiab]
MeSH: [mh], Major MeSH: [majr], Supplementary Concept: [nm]
Text Words: [tw], All Fields: [all]
Publication Type: [pt], Language: [la]
Output and only output the formulated Boolean query inside <answer></answer> tags. Do not include any explanation or content outside or inside the <answer> tags.

Figure 4: No-reasoning prompt

System Message

You are an expert systematic review information specialist.
You are tasked to formulate a systematic review Boolean query in response to a research topic. Your reasoning process should be enclosed within <think></think>, and the final Boolean query must be enclosed within <answer></answer> tags. Do not include anything outside of these tags.

User Message

You are given a systematic review research topic, with the topic title "topic".
Your task is to generate a highly effective Boolean query in MEDLINE format for PubMed.
The query should balance **high recall** (capturing all relevant studies) with **reasonable precision** (avoiding irrelevant results):
- Use both free-text terms and MeSH terms (e.g., chronic pain[tiab], Pain[mh]).
- **Do not wrap terms or phrases in double quotes**, as this disables automatic term mapping (ATM).
- Combine synonyms or related terms within a concept using OR.
- Combine different concepts using AND.
- Use wildcards (*) to capture word variants (e.g., vaccin* -> vaccine, vaccination):
- Terms must have >= 4 characters before the * (e.g., colo*)
- Wildcards work with field tags (e.g., breastfeed*[tiab]).
- Field tags limit the search to specific fields and disable ATM.
- Do not include date limits.
- Tag terms using appropriate fields (e.g., covid-19[ti] vaccine[ti] children[ti]) when needed.

Only use the following allowed field tags:
Title: [ti], Abstract: [ab], Title/Abstract: [tiab]
MeSH: [mh], Major MeSH: [majr], Supplementary Concept: [nm]
Text Words: [tw], All Fields: [all]
Publication Type: [pt], Language: [la]
Output your full reasoning inside <think></think>.
Output the final Boolean query inside <answer></answer>.
Do not include any content outside these tags.

Figure 5: Free-text Reasoning prompt with <think> and <answer> outputs

System Message

You are an expert systematic review information specialist.
Formulate a systematic review Boolean query using step-by-step reasoning inside <think> </think>, and output the final query inside <answer> </answer>.

User Message

You are given a systematic review topic titled: "topic".
Construct a Boolean query using the **conceptual method**, based on domain logic and structured thinking.
Step 1: Identify 2–3 key concepts from the topic (e.g., Population, Intervention, Outcome).
Step 2: For each concept: - List related terms: synonyms, variants, relevant MeSH terms. - Prioritise specific, high-precision terms.
Step 3: Create a Boolean block per concept: - Combine terms using OR - Use free-text terms and MeSH terms (e.g., chronic pain[tiab], Pain[mh]) - **Do not wrap terms or phrases in double quotes**, as this disables automatic term mapping (ATM) - Tag terms individually when needed (e.g., covid-19[ti] vaccine[ti] children[ti]) - Field tags limit search scope and disable ATM
Step 4: Use wildcards (*) to capture word variants (e.g., vaccin* -> vaccine, vaccination): - Terms must have >= 4 characters before the * (e.g., colo*) - Wildcards work with field tags (e.g., breastfeed*[tiab]).
Step 5: Combine all Boolean blocks using AND: ((Concept1_term1[tiab] OR Concept1_term2[tiab] OR Concept1_termX[mh]) AND (Concept2_...))
Only use the following allowed field tags: Title: [ti], Abstract: [ab], Title/Abstract: [tiab] MeSH: [mh], Major MeSH: [majr], Supplementary Concept: [nm] Text Words: [tw], All Fields: [all] Publication Type: [pt], Language: [la]
Output your full reasoning inside <think>...</think>
Output only the final Boolean query inside <answer>...</answer>
Do not include any content outside these tags.
Do not include date limits.

Figure 6: Conceptual-method prompt with <think> reasoning and <answer> output

achieves recall that is within 1% of expert-written queries, while retrieving 17× fewer documents. This trade-off leads to improved F_3 and precision, indicating not only strong comprehensiveness but also a meaningful reduction in screening burden. AutoBool also surpasses the performance of the O1 model used in Wang et al. (2025), highlighting the advantage of retrieval-aware training via reinforcement learning over static prompting-based generation.

Seed Collection. Similar results are observed on the Seed Collection benchmark. While AutoBool does not fully match the performance of expert-authored queries in recall or F_3, it consistently outperforms all zero-shot prompting baselines and the O1 model from Wang et al. (2025). The model demonstrates strong recall-oriented behavior while keeping the number of retrieved documents at a practical level, maintaining its advantage in terms of screening efficiency. The success rate remains close to 100%, indicating stable formatting and reliable generation.

These results confirm that AutoBool generalizes effectively across different domains and collections, even though it was trained solely on PubMed data. The performance gains on CLEF and Seed further reinforce the value of reinforcement learning for robust and transferable Boolean query generation.

References

Justin Clark. 2013. Systematic reviewing: Introduction, locating studies and data abstraction. In *Methods of clinical epidemiology*, pages 187–211. Springer.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu, Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun,

System Message

You are an expert systematic review information specialist.
You are tasked to formulate a systematic review Boolean query step by step as a reasoning process within <think> </think>, and provide the Boolean query formulated <answer> </answer>.

User Message

You are given a systematic review research topic, with the topic title "topic".
You need to simulate a Boolean query construction process using the **objective method**, which is grounded in domain expertise and structured logic.
Step 1: Simulate a concise title and abstract (2–3 sentences) of a *relevant and focused* article clearly aligned with the topic. This is a hypothetical but plausible example.
Step 2: Based on the simulated text, identify *key informative terms or phrases* that best represent the article's core concepts. Prioritise specificity and informativeness. Avoid overly broad or ambiguous terms.
Step 3: Categorise each term into one of the following: - (A) Health conditions or populations (e.g., diabetes, adolescents) - (B) Treatments, interventions, or exposures (e.g., insulin therapy, air pollution) - (C) Study designs or methodologies (e.g., randomized controlled trial, cohort study) - (N/A) Not applicable to any of the above categories
Step 4: Using the categorised terms, build a Boolean query in MEDLINE format for PubMed: - Combine synonyms or related terms within each category using OR - Use both free-text terms and MeSH terms (e.g., chronic pain[tiab], Pain[mh]) - **Do not wrap terms or phrases in double quotes**, as this disables automatic term mapping (ATM) - Tag each term individually when needed (e.g., covid-19[ti] vaccine[ti] children[ti]) - Field tags limit the search to specific fields and disable ATM
Step 5: Use wildcards (*) to capture word variants (e.g., vaccin* -> vaccine, vaccination): - Terms must have >= 4 characters before the * (e.g., colo*) - Wildcards work with field tags (e.g., breastfeed*[tiab]).
Step 6: Combine all category blocks using AND: ((itemA1[tiab] OR itemA2[tiab] OR itemA3[mh]) AND (itemB1[tiab] OR ...) AND (itemC1[tiab] OR ...))
Only use the following allowed field tags: Title: [ti], Abstract: [ab], Title/Abstract: [tiab] MeSH: [mh], Major MeSH: [majr], Supplementary Concept: [nm] Text Words: [tw], All Fields: [all] Publication Type: [pt], Language: [la]
Place your full reasoning (including simulated abstract, term list, classification, and query construction) inside <think></think>.
Output the final Boolean query inside <answer></answer>.
Do not include anything outside the <think> and <answer> tags.
Do not include date restrictions.

Figure 7: Objective-method prompt with simulated article and structured reasoning in <think>

W. L. Xiao, Wangding Zeng, Wanjia Zhao, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaokang Zhang, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xinnan Song, Xinxia Shan, Xinyi Zhou, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. X. Zhu, Yang Zhang, Yanhong Xu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Yu, Yi Zheng, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Ying Tang, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yu Wu, Yuan Ou, Yuchen Zhu, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yukun Zha, Yunfan Xiong, Yunxian Ma, Yuting Yan, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhipeng Xu, Zhiyu Wu, Zhongyu Zhang, Zhuoshu Li, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Ziyi Gao, and Zizheng Pan. 2025. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Elke Hausner, Siw Waffenschmidt, Thomas Kaiser, and Michael Simon. 2012. Routine development of objectively derived search strategies. *Systematic reviews*, 1(1):19.

Evangelos Kanoulas, Rene Spijker, Dan Li, and Leif Azzopardi. 2018. Clef 2018 technology assisted reviews in empirical medicine overview. In *CLEF 2018 Evaluation Labs and Workshop: Online Working Notes, CEUR-WS*.

Wojciech Kusa, Oscar E Mendoza, Matthias Samwald, Petr Knoth, and Allan Hanbury. 2023. Csmed: bridging the dataset gap in automated citation screening for systematic literature reviews. *Advances in Neural Information Processing Systems*, 36:23468–23484.

Grace E. Lee and Aixin Sun. 2018. Seed-driven document ranking for systematic reviews in evidence-based medicine. In *Proceedings of the 41st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 455–464.

Table 5: Effectiveness of LLM-generated Boolean queries on the CLEF TAR set. **Bold** indicates the best result for each model within a setting; Underlined indicates the overall best across all models. **Expert-Crafted** refers to results obtained by issuing the original Boolean queries from the dataset to PubMed. **Best Wang et al. (2025)-O1** denotes the O1 model using the P3 prompt from Wang et al. (2025), which achieved the highest recall in that study.

Setting	Model	Prompt	Recall	F3	Recall >80%	Recall >90%	Precision	Avg Retrieved	Avg Regen	%Success
Zero-shot	Expert Crafted		**0.8458**	**0.0970**	**80.56**	**79.17**	**0.0206**	**14327.07**	/	**100.00**
	Best Wang et al. (2025)-O1		0.6545	0.1966	/	/	0.1078	/	/	/
	GPT-4O	N.R	0.4258	0.2245	19.44	11.11	**0.1275**	**389.74**	1.01	**100.00**
	GPT-4O	R	0.4534	0.2160	22.22	16.67	0.1013	459.18	1.01	**100.00**
	GPT-4O	R-con	**0.5283**	**0.2283**	**26.39**	**19.44**	0.1080	535.54	**1.00**	**100.00**
	GPT-4O	R-obj	0.3498	0.1449	16.67	9.72	0.1139	422.67	1.07	**100.00**
	O3	R	**0.7454**	0.3167	**56.94**	**40.28**	0.0879	663.10	1.58	98.61
	O3	R-con	0.7270	**0.3196**	48.61	34.72	**0.0928**	**627.35**	1.18	**100.00**
	O3	R-obj	0.5811	0.2157	25.00	13.89	0.0779	731.69	**1.15**	**100.00**
	Qwen3-4B	N.R	0.0255	0.0235	0.00	0.00	0.0608	**59.00**	6.39	41.67
	Qwen3-4B	R	0.1756	0.1246	2.78	0.00	**0.1389**	175.38	1.65	98.61
	Qwen3-4B	R-con	**0.5282**	**0.1579**	**33.33**	**22.22**	0.0871	608.08	**1.18**	**100.00**
	Qwen3-4B	R-obj	0.0717	0.0422	1.39	1.39	0.0447	236.60	2.15	94.44
AutoBool ($\alpha = 0.5$) Weak R.O	Qwen3-4B	N.R	**0.7919**	**0.2497**	**65.28**	**41.67**	**0.0728**	772.68	1.12	98.61
	Qwen3-4B	R	0.6677	0.1773	43.06	31.94	0.0709	769.17	**1.01**	**100.00**
	Qwen3-4B	R-con	0.5729	0.1520	30.56	22.22	0.0548	**722.76**	1.39	97.22
	Qwen3-4B	R-obj	0.7909	0.1704	62.50	**41.67**	0.0390	867.03	**1.01**	**100.00**
AutoBool ($\alpha = 1$) Mod R.O	Qwen3-4B	N.R	**0.8387**	**0.2401**	**70.83**	**51.39**	0.0499	818.31	**1.00**	**100.00**
	Qwen3-4B	R	0.6090	0.2011	31.94	19.44	**0.0782**	**696.92**	1.06	**100.00**
	Qwen3-4B	R-con	0.5885	0.1684	34.72	20.83	0.0672	697.35	1.03	**100.00**
	Qwen3-4B	R-obj	0.7619	0.2035	55.56	43.06	0.0433	880.74	**1.00**	**100.00**
AutoBool ($\alpha = 2$) Heavy R.O	Qwen3-4B	N.R	**0.8239**	**0.2193**	**73.61**	**51.39**	0.0538	830.08	**1.00**	**100.00**
	Qwen3-4B	R	0.5723	0.2054	31.94	15.28	**0.0737**	**698.60**	**1.00**	**100.00**
	Qwen3-4B	R-con	0.6571	0.1348	41.67	29.17	0.0356	869.71	1.04	**100.00**
	Qwen3-4B	R-obj	0.8177	0.1995	65.28	**51.39**	0.0406	904.31	**1.00**	**100.00**

Andrew MacFarlane, Tony Russell-Rose, and Farhad Shokraneh. 2022. Search strategy formulation for systematic reviews: Issues, challenges and opportunities. *Intelligent Systems with Applications*, page 200091.

Minh Nguyen, Toan Quoc Nguyen, Kishan KC, Zeyu Zhang, and Thuy Vu. 2024. Reinforcement learning from answer reranking feedback for retrieval-augmented answer generation. In *Proceedings of INTERSPEECH*.

Eric Sayers. 2010. A general introduction to the e-utilities. *Entrez Programming Utilities Help [Internet]. Bethesda: National Center for Biotechnology Information*.

Harrisen Scells and Guido Zuccon. 2018. Generating better queries for systematic reviews. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, pages 475–484, New York, NY, USA. ACM.

Harrisen Scells, Guido Zuccon, and Bevan Koopman. 2020a. A computational approach for objectively derived systematic review search strategies. In *Proceedings of the 42nd European Conference on Information Retrieval*, pages 385–398.

Harrisen Scells, Guido Zuccon, Bevan Koopman, and Justin Clark. 2020b. Automatic boolean query formulation for systematic review literature search. In *Proceedings of the 29th World Wide Web Conference*, pages 1071–1081.

Moritz Staudinger, Wojciech Kusa, Florina Piroi, Aldo Lipani, and Allan Hanbury. 2024. A reproducibility and generalizability study of large language models for query generation. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, pages 186–196.

Mark Stevenson and Reem Bin-Hezam. 2023. Stopping methods for technology-assisted reviews based on point processes. *ACM Transactions on Information Systems*, 42(3):1–37.

Sebastian Straube, Judith Heinz, Patrick Landsvogt, and Tim Friede. 2021. Recall, precision, and coverage of literature searches in systematic reviews in occupational medicine: an overview of cochrane reviews.

U.S. National Library of Medicine. 2003. Pmc open access subset. https://pmc.ncbi.nlm.nih.gov/tools/openftlist/. Last accessed: 2025-07-18.

Shuai Wang, Harrisen Scells, Justin Clark, Bevan Koopman, and Guido Zuccon. 2022a. From little things

Table 6: Effectiveness of LLM-generated Boolean queries on Seed Collection.. **Bold** indicates the best result for each model within a setting; Underlined indicates the overall best across all models. **Expert-Crafted** refers to results obtained by issuing the original Boolean queries from the dataset to PubMed. **Best Wang et al. (2025)-O1** denotes the O1 model using the P3 prompt from Wang et al. (2025), which achieved the highest recall in that study

Setting	Model	Prompt	Recall	F3	Recall >80%	Recall >90%	Precision	Avg Retrieved	Avg Regen	%Success
Zero-shot	Expert Crafted		**0.7241**	**0.1869**	**57.50**	**25.00**	0.0341	1416.55	/	**100.00**
	Best Wang et al. (2025)-O1		0.5786	0.0852	/	/	0.0523	/	/	/
	GPT-4O	N.R	0.3106	0.1073	2.50	**2.50**	0.0694	**369.30**	**1.00**	**100.00**
	GPT-4O	R	0.3330	0.1076	5.00	**2.50**	0.0317	426.77	1.05	**100.00**
	GPT-4O	R-con	**0.3936**	**0.1262**	2.50	0.00	0.0382	559.98	**1.00**	**100.00**
	GPT-4O	R-obj	0.2647	0.0847	**7.50**	**2.50**	**0.0882**	478.00	1.25	**100.00**
	O3	R	**0.7027**	0.1142	**47.50**	**25.00**	0.0174	733.10	1.48	**100.00**
	O3	R-con	0.6482	0.1336	30.00	15.00	0.0235	659.42	1.35	**100.00**
	O3	R-obj	0.5411	**0.1418**	22.50	12.50	**0.0424**	**653.65**	**1.12**	**100.00**
	Qwen3-4B	N.R	0.0003	0.0000	0.00	0.00	0.0000	**124.00**	8.10	25.00
	Qwen3-4B	R	0.0306	0.0144	0.00	0.00	**0.0327**	170.43	4.35	75.00
	Qwen3-4B	R-con	**0.3768**	**0.0567**	**10.00**	**10.00**	0.0193	596.70	**1.23**	**100.00**
	Qwen3-4B	R-obj	0.0384	0.0113	0.00	0.00	0.0275	194.13	2.58	97.50
AutoBool ($\alpha = 0.5$) Weak R.O	Qwen3-4B	N.R	0.5820	0.0855	**37.50**	20.00	0.0126	803.08	1.23	97.50
	Qwen3-4B	R	0.5445	**0.0899**	27.50	25.00	0.0147	726.08	**1.02**	**100.00**
	Qwen3-4B	R-con	0.4701	0.0873	17.50	15.00	**0.0198**	**651.63**	1.48	95.00
	Qwen3-4B	R-obj	**0.6419**	0.0820	**37.50**	**27.50**	0.0170	817.83	1.15	**100.00**
AutoBool ($\alpha = 1$) Mod R.O	Qwen3-4B	N.R	**0.6828**	0.0943	**47.50**	**35.00**	0.0136	827.00	1.23	97.50
	Qwen3-4B	R	0.5301	0.0668	30.00	17.50	0.0109	**735.50**	**1.00**	**100.00**
	Qwen3-4B	R-con	0.5307	**0.1051**	30.00	12.50	**0.0226**	736.30	1.15	**100.00**
	Qwen3-4B	R-obj	0.5890	0.0642	35.00	27.50	0.0097	831.35	1.05	**100.00**
AutoBool ($\alpha = 2$) Heavy R.O	Qwen3-4B	N.R	0.6539	0.0749	35.00	25.00	0.0099	855.03	1.25	97.50
	Qwen3-4B	R	0.5963	**0.1071**	32.50	20.00	**0.0215**	**730.12**	1.05	**100.00**
	Qwen3-4B	R-con	0.5066	0.0528	20.00	17.50	0.0123	786.40	**1.00**	**100.00**
	Qwen3-4B	R-obj	**0.6804**	0.0672	**50.00**	**35.00**	0.0089	896.00	1.23	97.50

big things grow: A collection with seed studies for medical systematic review literature search. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3176–3186.

Shuai Wang, Harrisen Scells, Bevan Koopman, and Guido Zuccon. 2022b. Neural rankers for effective screening prioritisation in medical systematic review literature search. In *Proceedings of the 26th Australasian Document Computing Symposium*, pages 1–10.

Shuai Wang, Harrisen Scells, Bevan Koopman, and Guido Zuccon. 2023. Can chatgpt write a good boolean query for systematic review literature search? In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, pages 1426–1436, New York, NY, USA. Association for Computing Machinery.

Shuai Wang, Harrisen Scells, Bevan Koopman, and Guido Zuccon. 2025. Reassessing large language model boolean query generation for systematic reviews. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '25, pages 3296–3305, New York, NY, USA. Association for Computing Machinery.

Shuai Wang, Harrisen Scells, Ahmed Mourad, and Guido Zuccon. 2022c. Seed-driven document ranking for systematic reviews: A reproducibility study. In *European Conference on Information Retrieval*, pages 686–700. Springer.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Weinan Zhang, Xiangyu Zhao, Li Zhao, Dawei Yin, Grace Hui Yang, and Alex Beutel. 2020. Deep reinforcement learning for information retrieval: Fundamentals and advances. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2468–2471.

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*.