CLEAR: LEARNING TO CLARIFY FOR EARLIER GAINS IN DOMAIN-SPECIFIC INTERACTIVE RETRIEVAL

Anonymous authors
Paper under double-blind review

ABSTRACT

Domain-specific document retrieval often begins with queries that leave relevance-critical information unspecified. Interactive retrieval elicits this information through clarification questions and query updates. However, clarification must adapt to the evolving dialogue, and questions must be prioritized to improve retrieval with fewer rounds. We propose Clarification Learning for Early and Accurate Retrieval (CLEAR), a framework that uses retrieval results and dialogue history to identify what to clarify, and retrieval feedback to learn question priorities. Progressive Residual Facet Discovery (PRFD) uses residual quantization to organize document contrasts, combining them with domain facts and dialogue history to identify requirement facets, or aspects of the information need that can be clarified. PRFD produces candidate facets; the policy selects a facet and generates a question. Turn-aware Hierarchical Policy Optimization (THPO) constructs a return matrix from per-round retrieval gains and uses hierarchical comparisons across facet categories and candidate facets for reinforcement learning. Across legal, medical, and patent benchmarks, the domain-specific configurations achieve higher NDCG@10 at R4 and mean NDCG@10 over R1–R4 than the compared interactive baselines. The patent configuration disables round-dependent prefix selection, assessing framework transfer rather than the full progressive mechanism. Legal and medical trajectories additionally quantify the proportion of queries that first reach an NDCG@10 threshold within each round budget.

1 INTRODUCTION

Relevance in domain-specific document retrieval often depends on facts that are absent from the initial query. In legal retrieval, similar cases may differ in factual details that determine their relevance (Liu et al., 2025). In medical retrieval, a description of a health issue may omit information needed to distinguish relevant literature. Users may be able to describe their circumstances or concerns but lack the domain knowledge needed to identify relevance-critical facts and express them in specialized terminology (Zeng & Tse, 2006). Such omissions leave the retriever without sufficient information to determine which documents match the underlying information need.

Interactive retrieval addresses this problem by eliciting additional information through clarification questions (Radlinski & Craswell, 2017; Zamani et al., 2023). Given an initial query and its retrieval results, the system asks a question, incorporates the user's answer into the query, and retrieves again (Aliannejadi et al., 2019; 2021). As illustrated in Figure 1, successive answers add factual constraints that help distinguish relevant documents. Retrieved documents provide evidence for subsequent questions (Sekulić et al., 2022; Liu et al., 2025), while dialogue history records what has already been clarified (Bi et al., 2021).

Clarification in this setting requires adapting both the content and priority of questions. First, what needs clarification changes as the interaction progresses. Each question should build on previous answers and target unresolved aspects of the information need, using the current retrieval results as evidence (Sekulić et al., 2021; Bi et al., 2021). Document differences alone do not establish which aspect should be clarified or at what level of detail. Asking about fine-grained details too early may rely on unsupported assumptions, whereas continuing to ask broad questions after the candidate doc-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

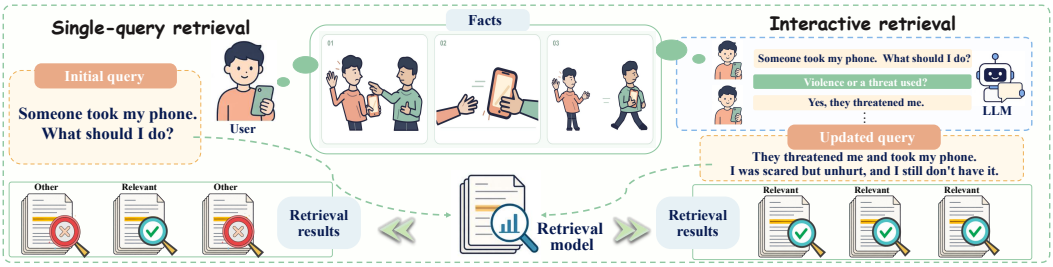

Figure 1: Multi-round interactive retrieval uses clarification answers to update the query and progressively focus the retrieved legal cases. Teal denotes relevant documents. Illustrative example.

uments become more focused may fail to distinguish closely related documents. Second, the number of questions is limited. Users may also stop early because of interaction costs or satisfaction with the current results, so early retrieval gains can improve the quality available at stopping even when policies achieve similar final quality (Radlinski & Craswell, 2017; Zamani et al., 2023). Question priorities must therefore account for when retrieval improvements occur, including the effects of a question on subsequent rounds (Zhang et al., 2018; Montazeralghaem & Allan, 2022). Terminal reward, defined here as retrieval feedback at the last round of the interaction budget, provides no direct incentive to achieve improvements earlier when used as the sole objective.

In this paper, we propose an interactive retrieval framework, Clarification Learning for Early and Accurate Retrieval (CLEAR). It uses retrieval results and dialogue history to identify what to clarify, and retrieval feedback to learn question priorities. Specifically, Progressive Residual Facet Discovery (PRFD) uses residual quantization to organize contrasts among retrieved documents from coarse to fine. It combines these contrasts with domain facts and dialogue history to identify requirement facets (hereafter facets), namely aspects of the information need that remain to be specified. PRFD produces candidate facets, from which the policy selects a facet and generates a question. Document contrasts suggest what to ask, while user answers determine which factual constraints are added to the retrieval query. Turn-aware Hierarchical Policy Optimization (THPO) then learns question priorities through reinforcement learning following supervised fine-tuning. Its return matrix assigns current and subsequent retrieval gains to individual clarification decisions. Hierarchical comparisons across facet categories and candidate facets provide the learning signals for policy optimization. CLEAR keeps the retriever fixed and improves retrieval by optimizing the clarification process, without retraining the retrieval model.

We evaluate CLEAR across legal, medical, and patent retrieval using LeCaRDv2 (Li et al., 2024), TREC-COVID (Roberts et al., 2020), and CLEF-IP (Piroi, 2010). The domain-specific configurations achieve higher NDCG@10 at R4 and mean NDCG@10 over R1–R4 than the interactive baselines in Table 1. The patent configuration disables round-dependent prefix selection; its results assess framework transfer, not the full progressive mechanism. Legal and medical analyses examine first attainment of an NDCG@10 threshold. By R2, 58.8% of legal queries and 40.0% of medical queries have reached NDCG@10 $\geq$ 0.5 at least once. Additional experiments on legal retrieval examine the effects of discovery granularity, candidate count, and policy optimization.

The contributions of this paper can be summarized as follows:

- We investigate domain-specific interactive retrieval from the perspective of adapting clarification to previous interactions and improving retrieval quality within a limited question budget.
- We propose CLEAR, which combines residual quantization for discovering unresolved facets with reinforcement learning for prioritizing clarification questions.
- We evaluate CLEAR across legal, medical, and patent retrieval. The domain-specific results assess framework applicability; legal analyses examine discovery and policy choices, while legal and medical trajectories quantify early quality attainment.
- We will make the implementation and evaluation scripts publicly available to support reproducibility and further research.

2 Related Work

Interactive retrieval and clarification. Conversational search refines information needs through interaction (Radlinski & Craswell, 2017; Zamani et al., 2023), with Qulac and ClariQ supporting clarification evaluation (Aliannejadi et al., 2019; 2021). ZeroShotCQGen uses constrained decoding to generate questions without task-specific training (Wang et al., 2023), while AGENT-CQ combines LLM-based question generation with CrowdLLM evaluation (Siro et al., 2024). In code generation, ClarifyGPT detects ambiguous requirements through code consistency checks and generates targeted questions using user feedback (Mu et al., 2024). Our retrieval adaptations use AGENT-CQ's generation stage and ClarifyGPT's clarification idea, not their complete original pipelines (Appendix B). Within this line of work, CLEAR studies clarification that elicits relevance-critical facts that are missing from the query to improve document retrieval over successive interactions.

Document-grounded facet discovery. Facet-conditioned generation uses query facets to guide clarification questions (Sekulić et al., 2021), while DocFeatures exploits features extracted from retrieved documents to support clarification (Sekulić et al., 2022). SRQG uses web search results to generate clarifying questions and includes the SRQG-LTR variant (Zhao et al., 2022). Other work incorporates inferred user intent into question generation (Zhao et al., 2024). CARQ uses differences between retrieved legal cases to generate questions without externally annotated domain knowledge (Liu et al., 2025). Building on document-grounded clarification, CLEAR's PRFD focuses on adapting the granularity of candidate facets to the available evidence and dialogue history.

Learning clarification policies. Prior work studies aspect ordering in conversational search (Zhang et al., 2018), reinforcement learning for relevant-question generation and risk-aware interaction (Montazeralghaem & Allan, 2022; Wang & Ai, 2021), and retrieval-oriented query rewriting (Wu et al., 2022). MMR-BERT balances query relevance with information from previous negative feedback when selecting questions (Bi et al., 2021). CARQ learns to select neighboring legal cases for clarification using reward-augmented maximum likelihood and retrieval feedback (Liu et al., 2025). CLEAR's THPO builds on retrieval-driven interaction learning, with an emphasis on obtaining retrieval gains earlier while accounting for a question's effects on subsequent rounds.

3 CLEAR: Method

CLEAR couples an interactive retrieval loop with two complementary modules (Figure 2). At each round, PRFD uses retrieved documents and dialogue history to propose candidate facets. The clarification policy selects a facet and asks a question; the user response updates the query and produces a new ranking. During training, THPO converts retrieval gains along these trajectories into round-specific returns and hierarchical advantages to update the policy. We first define the interaction, then describe how PRFD constructs candidate facets and how THPO learns their priorities from retrieval feedback.

3.1 Interactive Retrieval Formulation

Interaction and observations. Given an initial query q_0 and document pool $\mathcal{D}$, a fixed retriever returns $D_0 = f(q_0; \mathcal{D})$. A round consists of one question, the user answer, query updating, and retrieval. At round t, PRFD selects candidate facets $\mathcal{E}_t \subseteq \mathcal{F}$ from a fixed domain inventory $\mathcal{F}$, using D_{t-1} and dialogue history H_{t-1}; it does not generate new facet categories. The information need is the user's overall retrieval goal; a candidate facet is an eligible member of $\mathcal{E}_t$ representing one aspect to clarify. The selected facet e_t identifies information to elicit, while question a_t expresses it to the user. The observation o_t contains visible dialogue and selected retrieval evidence. The interaction proceeds as

$$\begin{aligned} (e_t, a_t) \sim \pi_\theta(\cdot \mid o_t, \mathcal{E}_t), \qquad & H_t = H_{t-1} \parallel (a_t, u_t), \\ x_t = U(x_{t-1}, a_t, u_t), \quad & D_t = f(x_t; \mathcal{D}), \end{aligned} \tag{1}$$

Here, $x_0 = q_0$, H_0 is empty, and u_t is the user response. The update U incorporates valid, informative answers into the query context. Private user facts and relevance judgments are used only by the training and evaluation environment, not exposed to discovery or policy inputs.

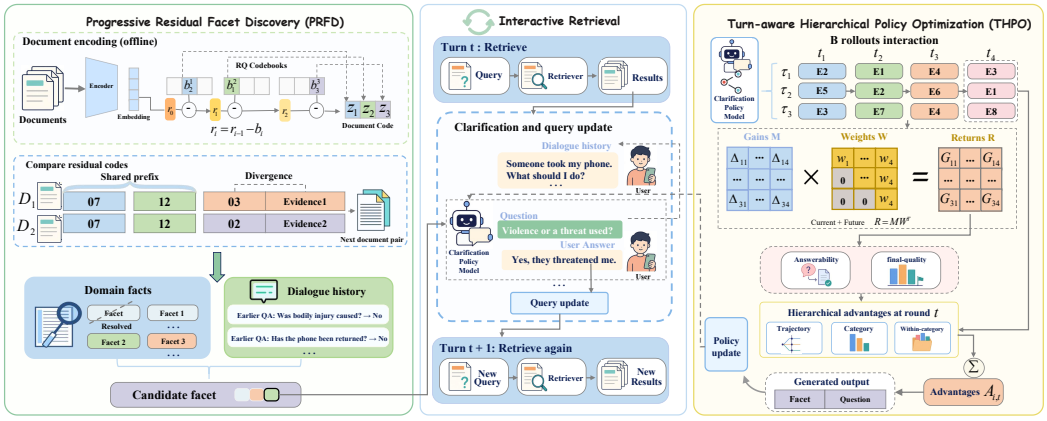

Figure 2: CLEAR overview. PRFD derives candidate facets from retrieved documents and dialogue history. User answers update the query and retrieval results. During training, THPO converts retrieval gains into a round-level return matrix and hierarchical advantages to update the clarification policy; the retriever remains fixed.

Retrieval utility. Let $N_t = \mathrm{NDCG@10}(D_t; y)$ for relevance judgments y, and $\Delta_t = N_t - N_{t-1}$. We seek early gains alongside final quality, measured by $U_{\text{final}} = N_T$ and $U_{\text{avg}} = T^{-1} \sum_{t=1}^{T} N_t$. The latter represents expected quality under a uniform stopping round; N_0 is reported separately.

3.2 Progressive Residual Facet Discovery

Inputs, contrasts, and candidates. Given retrieved documents D_{t-1} and history H_{t-1}, PRFD maps supported document contrasts to eligible facets in the fixed inventory $\mathcal{F}$. Residual quantization organizes document representations, while history filters previously attempted targets.

Organizing document differences. PRFD organizes document comparisons through successive approximations of their representations using residual vector quantization (Chen et al., 2010). For a document latent v_d (including category conditioning where used), initialize $r_d^{(0)} = v_d$. The quantization component assigns each residual to codebook $\mathcal{B}_l = \{b_k^{(l)}\}$:

$$\mathcal{L}_{\mathrm{RQ}}^{(l)} = \sum_{d \in \mathcal{D}_{\mathrm{fit}}} \min_k \|r_d^{(l-1)} - b_k^{(l)}\|_2^2,$$
$$z_l(d) = \arg\min_k \|r_d^{(l-1)} - b_k^{(l)}\|_2^2, \qquad r_d^{(l)} = r_d^{(l-1)} - b_{z_l(d)}^{(l)}. \tag{2}$$

The code sequence $z(d) = (z_1(d), \ldots, z_L(d))$ organizes document comparisons: a shared prefix denotes common early assignments, and a divergence denotes the first differing assignment. Domain-specific extraction records facts and supporting evidence under an inventory $\mathcal{F}$. Offline resource fitting is specified in Appendix C.1; encoders, codes, facts, and categories are frozen during policy learning.

From residual contrasts to candidate facets. At each round, PRFD compares nearby documents sharing a domain context and an RQ prefix but differing in supported facts. Each proposal associates a stable facet identifier with its category, contrasting values or passages, and a question template. The provider retains the highest-ranked supporting pair for each proposed facet rather than pooling incompatible evidence. Previously attempted facets are excluded, and missing facts are not interpreted as negations. The resulting candidate list guides the policy to select one facet and generate a single-focus question. Document evidence establishes what can be asked, not what the user's answer must be.

Later rounds favor document pairs with longer shared prefixes. Eligible facets are ranked using visible fact differences and residual structure, with a relaxed preference when no candidate qualifies.

PRFD supplies candidates, while the policy learns which facet to select using the retrieval feedback described next. Appendix C.1 specifies candidate construction; Appendix C.2 documents the interaction inputs.

3.3 Turn-aware Hierarchical Policy Optimization

From interactions to retrieval feedback. THPO starts from a supervised clarification policy and samples B complete T-round trajectories per training query. At each round, the policy generates a facet identifier and a question. The environment checks facet validity, single factual focus, and question alignment, then answers from the corresponding private fact slot when available. Otherwise, no factual constraint is added. Query updates and retrieval produce scores $N_{i,t}$ for trajectory i, and successive differences $\Delta_{i,t} = N_{i,t} - N_{i,t-1}$ provide the observed gains. These gains are then allocated to the current and earlier clarification decisions.

Stopping-weighted returns and matrix representation. To account for when gains occur, let $p_s = P(S = s)$ define a stopping-round distribution and $w_k = P(S \geq k) = \sum_{s=k}^{T} p_s$. Decision t receives weighted current and future gains:

$$G_{i,t} = \sum_{k=t}^{T} w_k \Delta_{i,k}. \tag{3}$$

The stopping-weighted reward-to-go in Equation 3 can be written in matrix form. Writing the gain matrix as $M_{i,k} = \Delta_{i,k}$ and the upper-triangular allocation as $W_{t,k} = w_k \mathbf{1}\{k \geq t\}$ gives

$$\mathcal{R} = MW^{\top} \in \mathbb{R}^{B \times T}, \qquad \mathcal{R}_{i,t} = G_{i,t}. \tag{4}$$

Thus, M records observed gains, W determines their temporal allocation, and $\mathcal{R}$ contains a return for each trajectory and decision. The upper-triangular structure belongs to W: each decision receives only current and subsequent gains. Under uniform stopping, $w_k = (T - k + 1)/T$ and $G_{i,1} = U_{\text{avg},i} - N_{i,0}$, so gains obtained earlier benefit more stopping points. Training uses complete trajectories; the stopping distribution defines the objective rather than modeling observed user behavior. Appendix C gives the four-round expansion.

Answerability and final-quality constraints. Let $b_{i,t}$ indicate a valid question with an available grounded answer. Question validity concerns the output protocol and factual focus; answerability additionally requires support in the private user facts. We impose $\mathbb{E}[b_t] \geq \kappa_t$ and $\mathbb{E}[N_T] \geq \nu$, where κ_t and ν are training targets for expected answerability and final quality. Nonnegative multipliers give

$$\widetilde{G}_{i,t} = G_{i,t} + \sum_{k=t}^{T} \lambda_k (b_{i,k} - \kappa_k) + \mu(N_{i,T} - \nu). \tag{5}$$

The multipliers increase when estimated expectations fall below the targets.

Hierarchical comparisons. At round t, THPO compares the returns assigned to complete trajectories in the sampled query group, mean returns across selected facet categories, and mean returns across selected facets within each category. Group standardization Z yields

$$A_{\cdot,t} = Z\left(a_{\cdot,t}^{\text{traj}} + a_{\cdot,t}^{\text{cat}} + a_{\cdot,t}^{\text{facet}}\right). \tag{6}$$

The trajectory component measures relative return across the sampled group. Category and facet components compare their mean returns, and each trajectory receives the scores associated with its selected facet. Constant or insufficient comparison groups contribute zero. The comparisons share an initial query but allow different subsequent histories.

Policy update. Let $\ell_\theta(i,t)$ be the mean log probability of generated facet and question tokens, excluding prompt tokens. The round-specific advantage supplies the learning signal:

$$\mathcal{L}_{\text{RL}} = -\frac{1}{BT} \sum_{i,t} \text{sg}(A_{i,t}) \ell_\theta(i,t) + \beta_{\text{ref}} \Omega_{\text{ref}}, \tag{7}$$

Here, sg stops gradients and Ω_{ref} penalizes clipped differences in length-normalized output scores against a frozen supervised reference; it is not an exact token-level KL divergence. Gradient descent updates only the clarification policy, which then generates new interaction trajectories. The retriever, document encoder, and codebooks remain fixed. Supervised anchor updates retain the output protocol; Appendix C.4 specifies the regularizer and stabilization rules.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Evaluation scope. We evaluate document ranking with simulated users over four clarification rounds, using a fixed pool of up to 100 candidate documents per query in each domain. Pool membership remains unchanged across rounds.

Datasets and retrieval. We construct interactive benchmarks from LeCaRDv2 case queries, TREC-COVID topics, and CLEF-IP 2010 prior-art topics (Piroi, 2010), with 160 test queries per domain. Medical and patent query variants are grouped by original topic before splitting; initial queries and candidate pools are then frozen. SAILER (Li et al., 2023), MedCPT (Jin et al., 2023), and BM25 (Robertson & Zaragoza, 2009) serve as the fixed legal, medical, and patent retrievers, respectively. Appendix C.2 details the source-to-query mapping, splits, and pool construction.

Questioning model and simulated users. The clarification policy in CLEAR is trained from Qwen3-4B (Yang et al., 2025). During evaluation, an API-accessed GPT model acts as the response simulator, playing the role of the user. It receives the complete underlying factual description for a query and answers each clarification question using relevant information from that description. This complete description is hidden from the questioning model, which receives only the initial query, dialogue history, and permitted retrieval evidence. Each episode comprises four clarification rounds and produces one trajectory of questions, responses, and retrieval results. Simulator prompts and response-validation rules are documented in Appendix C.

Evaluation metrics. We report MAP and P@5 at R4, NDCG@10 at R4, and mean NDCG@10 over R1–R4. R0 denotes retrieval before clarification. Early improvement is measured in interaction rounds, not elapsed time.

Baselines. We compare CLEAR with domain-adapted implementations of seven clarification baselines: MMR-BERT (Bi et al., 2021), ZeroShotCQGen (Wang et al., 2023), SRQG-LTR (Zhao et al., 2022), DocFeatures (noun) (Sekulić et al., 2022), CARQ (Liu et al., 2025), ClarifyGPT (Mu et al., 2024), and AGENT-CQ-Temp (Siro et al., 2024). SRQG-LTR uses retrieval-gain training for candidate-document-pair preferences in our implementation. Appendix B distinguishes these implementations from the original systems.

Reference settings. Full-information query retrieves directly from the complete factual description, providing an empirical reference rather than a guaranteed upper bound. Initial query only retrieves once without clarification.

Evaluation series. CLEAR names the framework, and Ours is its plot or table shorthand; Full THPO denotes the complete legal policy-learning configuration. Encoders, codebooks, and facet inventories are domain-specific (Table 10); the four-round evaluation budget is shared. Patent CLEAR uses RQ evidence without round-dependent prefix selection. Appendix C.5 specifies configuration scope; Appendix B describes baseline adaptations.

4.2 MAIN RESULTS

Across the three domain-specific configurations, CLEAR has the highest observed NDCG@10 at R4 and mean NDCG@10 over R1–R4 among the compared interactive methods (Table 1). Full-information query and Initial query only provide single-retrieval reference points; dashes indicate unavailable or inapplicable results.

Table 1: Retrieval performance (↑). Bold marks column maxima among interactive baselines and CLEAR. Evaluation-series scope is specified in Appendix C.5.

Method	LeCaRDv2 (SAILER)				TREC-COVID (MedCPT)				CLEF-IP (BM25)			
	R4			Mean over R1–R4	R4			Mean over R1–R4	R4			Mean over R1–R4
	MAP	P@5	NDCG @10	NDCG @10	MAP	P@5	NDCG @10	NDCG @10	MAP	P@5	NDCG @10	NDCG @10
Full-information query	0.572	0.645	0.627	—	0.505	0.545	0.641	—	0.415	0.489	0.388	—
Initial query only	0.339	0.286	0.282	—	0.215	0.046	0.155	—	0.214	0.198	0.163	—
MMR-BERT	0.350	0.290	0.298	0.298	0.316	0.266	0.353	0.319	0.270	0.286	0.223	0.202
ZeroShotCQGen	0.403	0.379	0.366	0.362	0.309	0.225	0.307	0.282	0.225	0.209	0.174	0.174
SRQG-LTR	0.414	0.361	0.374	0.351	0.327	0.241	0.344	0.284	0.263	0.269	0.213	0.201
DocFeatures (noun)	0.406	0.388	0.378	0.332	0.305	0.203	0.318	0.282	0.254	0.249	0.201	0.191
CARQ	0.426	0.393	0.397	0.362	0.339	0.269	0.379	0.331	0.247	0.243	0.200	0.194
ClarifyGPT	0.440	0.406	0.412	0.395	0.392	0.363	0.461	0.379	0.270	0.265	0.224	0.210
AGENT-CQ-Temp	0.456	0.448	0.448	0.417	0.348	0.281	0.378	0.319	0.314	0.346	0.269	0.240
CLEAR (Ours)	**0.512**	**0.550**	**0.534**	**0.499**	**0.421**	**0.391**	**0.491**	**0.408**	**0.324**	**0.368**	**0.288**	**0.276**

Full-information query and Initial query only report their single retrieval, not R4.

4.3 Early Retrieval Gains

For query i, let $N_{i,t}$ be NDCG@10 after round t. In addition to mean retrieval quality, we report the cumulative quality attainment rate at a common threshold η:

$$\mathrm{QAR}_{\eta}(t) = \frac{100}{n} \sum_{i=1}^{n} \mathbf{1}\left[\max_{0 \le s \le t} N_{i,s} \ge \eta\right]. \tag{8}$$

This measure counts queries that have reached the threshold at least once by round t, including R0. We use $\eta = 0.5$. QAR is monotone by construction: a query remains counted even if its later ranking deteriorates. To distinguish first attainment from current quality, we also report the percentage of queries satisfying $N_{i,t} \ge \eta$ at round t, without taking a maximum over earlier rounds. Figure 3 compares Initial query only, DocFeatures, MMR-BERT, ZeroShotCQGen, AGENT-CQ-Temp, and CLEAR on the same 160 queries with complete trajectories; queries 210 and 555 are excluded because the new DocFeatures run lacks completed records. Initial query only repeats its R0 result. This common-query analysis is distinct from the 160-query main table; evaluation-series differences are documented in Appendix C.5.

Figure 3(a) shows that CLEAR reaches 0.4937 NDCG@10 at R2, exceeding AGENT-CQ-Temp at R4 (0.4479), and reaches 0.5344 at R4 on the common-query subset. In panel (b), 58.9% of CLEAR queries have reached the threshold by R2, compared with 52.5% for AGENT-CQ-Temp by R4. Panel (c) shows that 51.9% of CLEAR queries meet the threshold at R2 itself, rising to 58.9% at R4. Together, the panels describe earlier threshold attainment alongside higher current retrieval quality, rather than user satisfaction or reduced runtime.

4.4 Granularity and Candidate Budget

Figure 4 reports NDCG@10 at R4 and mean NDCG@10 over R1–R4 for discovery-granularity and candidate-count settings on 160 LeCaRDv2 queries. CLEAR uses progressive discovery with 12 candidate facets.

Discovery Granularity. Progressive prefix preferences $(1, 1, 2, 2)$ have higher NDCG@10 at R4 and mean NDCG@10 over R1–R4 than fixed coarse $(1, 1, 1, 1)$ and fixed fine $(2, 2, 2, 2)$ settings. Query-paired intervals are reported in Appendix A.6.

Candidate Budget. The CLEAR configuration with 12 candidate facets has the highest NDCG@10 at R4 and mean NDCG@10 over R1–R4. Eight candidate facets score higher than

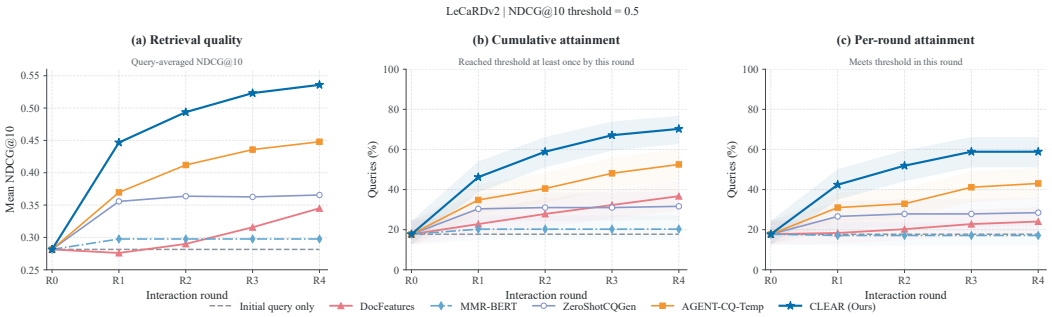

Figure 3: Retrieval quality and threshold attainment on 160 common LeCaRDv2 queries. (a) Query-averaged NDCG@10 at each round. (b) Percentage reaching NDCG@10 $\geq$ 0.5 at least once by each round, including R0. (c) Percentage meeting the same threshold in the current round. All panels use the same six methods, colors, and markers. Shading in (b) and (c) shows pointwise 95% Wilson intervals for CLEAR. Earlier attainment concerns interaction rounds, not wall-clock latency or sustained convergence.

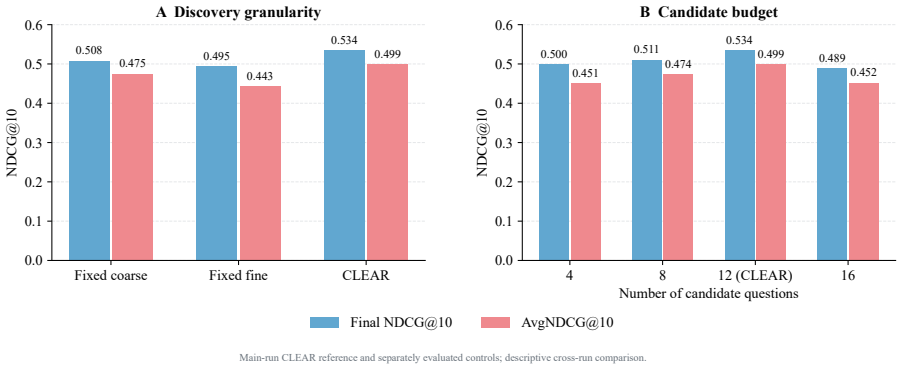

Figure 4: Discovery granularity and candidate count on LeCaRDv2. CLEAR uses progressive discovery and 12 candidate facets. The mean excludes R0.

four and sixteen on both metrics. Candidate count also changes prompt length under the fixed input budget.

Appendix A.5 reports all ten configurations and round-wise trajectories. Discovery controls vary inference settings; stopping-weight controls use separately trained policies.

4.5 Ablation Study

We study component contributions and policy-learning mechanisms on LeCaRDv2, using NDCG@10 at R4 and mean NDCG@10 over R1–R4 to distinguish final effectiveness from quality across rounds.

Discovery and reward formulation. Table 2 reports four discovery and retrieval-return configurations. Appendix A.1 documents the comparison scope.

The prompt-interface configuration scores 0.0895 lower in NDCG@10 at R4 and 0.0533 lower in mean NDCG@10 over R1–R4 than CLEAR. The terminal-reward configuration scores 0.0348 and 0.0276 lower, respectively; the configuration with both replacements has the lowest scores. The same ordering appears in MAP and P@5 at R4. With prompt-only discovery, round-specific returns also exceed terminal reward on both NDCG@10 measures. Together, these observed patterns are consistent with the two roles described in CLEAR: document-grounded candidate facets inform what to clarify, while retrieval feedback informs question priorities.

Table 2: Discovery and retrieval-return configurations on LeCaRDv2 (↑). The mean excludes R0.

Discovery	Retrieval return	R4			Mean over R1–R4
		MAP	P@5	NDCG@10	NDCG@10
PRFD (CLEAR)	Round-specific	**0.5117**	**0.5501**	**0.5344**	**0.4985**
Prompt-only	Round-specific	0.4526	0.4557	0.4449	0.4452
PRFD	Terminal reward	0.4914	0.5198	0.4996	0.4709
Prompt-only	Terminal reward	0.4369	0.4200	0.4231	0.4235

THPO mechanisms. Table 3 reports Full THPO and four single-factor policy interventions: three replace the main retrieval return, and one replaces hierarchical advantage calculation. Their checkpoint-level comparison scope is specified in Appendix C.5.

Table 3: THPO policy variants on LeCaRDv2: NDCG@10 at R4 and mean NDCG@10 over R1–R4.

Variant	Change from full THPO	NDCG@10 at R4	Mean NDCG@10 over R1–R4
Full THPO	None	**0.5344**	**0.4985**
Terminal reward	Main return $N_{i,T}$	0.4928	0.4488
Shared scalar return	Main return $G_{i,1}$ at every round	0.5164	0.4584
Immediate-gain	Main return $w_t\Delta_{i,t}$	0.5063	0.4574
Flat advantage	Group-standardized advantage	0.5110	0.4532

Full THPO exceeds the four variants in mean NDCG@10 over R1–R4 by 0.0401–0.0497 (Table 3). Shared scalar return and immediate-gain respectively remove round-specific allocation and subsequent retrieval gains from the main return; flat advantage removes category- and facet-level comparisons. Their lower mean scores are consistent with the roles of temporal credit allocation and hierarchical comparisons in learning question priorities. Auxiliary constraints remain unchanged.

Query-paired 95% intervals for all four mean-score differences exclude zero, whereas R4 intervals against shared scalar return and flat advantage include zero (Appendix A.2). The evidence thus supports average interaction quality more consistently than final-round gains across all variants. Evaluation-series differences remain relevant to interpretation (Appendix C.5).

5 Conclusion

We introduced CLEAR for domain-specific interactive document retrieval. PRFD combines residual-quantized document contrasts, domain facts, and dialogue history to produce candidate facets. The policy selects a facet and generates a question, whose answer updates the query. THPO learns question priorities using current and subsequent retrieval gains and hierarchical comparisons across facet categories and candidate facets, while keeping the retriever fixed.

Across legal, medical, and patent benchmarks, the evaluated configurations improve NDCG@10 at R4 and mean NDCG@10 over R1–R4 over the compared interactive baselines. Legal trajectories show earlier threshold attainment in interaction rounds. Patent results assess framework transfer without round-dependent prefix selection. These findings support combining facet discovery with learned question priorities under simulated-user and fixed-candidate-pool evaluation; real-user interaction and full-corpus retrieval remain future work.

AI USE STATEMENT

Generative AI tools, including a coding and writing assistant accessed through Codex, assisted with research-method and experiment design feedback, implementation and debugging, English–Chinese translation, manuscript drafting and revision, interpretation of experimental results, and figure design. API-accessed language models also serve as simulated users in the evaluation; their responses are experimental inputs rather than human-subject observations. Evaluation-series distinctions and available validation details are documented in Appendices A.1 and C. Quantitative figures use recorded evaluation outputs, not image-generated measurements. The authors are responsible for the accuracy of the final text, code, claims, and other AI-assisted artifacts.

ETHICS STATEMENT

This work studies retrieval over legal, medical, and patent documents, not the provision of professional advice. Retrieved documents and clarification questions may reflect errors or biases in the underlying collections, encoders, and language models. In particular, a question can introduce an unsupported assumption, and an apparently relevant document need not be appropriate for a user's circumstances. The evaluations use simulated users and do not establish safety or usefulness for real clients or patients. Deployment would require domain-specific validation, safeguards for sensitive user information, and careful consideration of the information sent to external model services.

REPRODUCIBILITY STATEMENT

We will release our code as open source to support reproducibility. Section 3.1 defines the interaction protocol and evaluation measures. Appendix C describes the implementation, data construction, and training and evaluation procedures, while Appendix B documents baseline adaptations. Additional experimental results and query-level analyses are provided in Appendices A.5 and A.6. The scope of comparisons across evaluation configurations is summarized in Appendix C.5.

REFERENCES

Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W. Bruce Croft. Asking clarifying questions in open-domain information-seeking conversations. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 475–484, 2019. doi: 10.1145/3331184.3331265.

Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail Burtsev. Building and evaluating open-domain dialogue corpora with clarifying questions. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 4473–4484, 2021. doi: 10.18653/v1/2021.emnlp-main.367. URL https://aclanthology.org/2021.emnlp-main.367/.

Keping Bi, Qingyao Ai, and W. Bruce Croft. Asking clarifying questions based on negative feedback in conversational search. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*, pp. 157–166, 2021. doi: 10.1145/3471158.3472232.

Yongjian Chen, Tao Guan, and Cheng Wang. Approximate nearest neighbor search by residual vector quantization. *Sensors*, 10(12):11259–11273, 2010. doi: 10.3390/s101211259.

Qiao Jin, Won Kim, Qingyu Chen, Donald C. Comeau, Lana Yeganova, W. John Wilbur, and Zhiyong Lu. MedCPT: Contrastive pre-trained transformers with large-scale PubMed search logs for zero-shot biomedical information retrieval. *Bioinformatics*, 39(11):btad651, 2023. doi: 10.1093/bioinformatics/btad651.

Haitao Li, Qingyao Ai, Jia Chen, Qian Dong, Yueyue Wu, Yiqun Liu, Chong Chen, and Qi Tian. SAILER: Structure-aware pre-trained language model for legal case retrieval. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1035–1044, 2023. doi: 10.1145/3539618.3591761.

Haitao Li, Yunqiu Shao, Yueyue Wu, Qingyao Ai, Yixiao Ma, and Yiqun Liu. LeCaRDv2: A large-scale chinese legal case retrieval dataset. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2251–2260, 2024. doi: 10.1145/3626772.3657887.

Bulou Liu, Yiran Hu, Qingyao Ai, Yiqun Liu, Yueyue Wu, Chenliang Li, and Weixing Shen. Generating clarifying questions for conversational legal case retrieval without external knowledge. *ACM Transactions on Information Systems*, 43(4):1–26, 2025. doi: 10.1145/3736161.

Ali Montazeralghaem and James Allan. Learning relevant questions for conversational product search using deep reinforcement learning. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, pp. 746–754, 2022. doi: 10.1145/3488560.3498526.

Fangwen Mu, Lin Shi, Song Wang, Zhuohao Yu, Binquan Zhang, ChenXue Wang, Shichao Liu, and Qing Wang. ClarifyGPT: A framework for enhancing LLM-based code generation via requirements clarification. *Proceedings of the ACM on Software Engineering*, 1(FSE):2332–2354, 2024. doi: 10.1145/3660810.

Florina Piroi. CLEF-IP 2010: Retrieval experiments in the intellectual property domain. In *Working Notes of CLEF 2010*, 2010. URL https://ceur-ws.org/Vol-1176/CLEF2010wn-CLEF-IP-PiroiEt2010.pdf.

Filip Radlinski and Nick Craswell. A theoretical framework for conversational search. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval*, pp. 117–126, 2017. doi: 10.1145/3020165.3020183.

Kirk Roberts, Tasmeer Alam, Steven Bedrick, Dina Demner-Fushman, Kyle Lo, Ian Soboroff, Ellen Voorhees, Lucy Lu Wang, and William R. Hersh. TREC-COVID: Rationale and structure of an information retrieval shared task for COVID-19. *Journal of the American Medical Informatics Association*, 27(9):1431–1436, 2020. doi: 10.1093/jamia/ocaa091.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 4(1–2):1–174, 2009. doi: 10.1561/1500000019.

Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. Towards facet-driven generation of clarifying questions for conversational search. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*, pp. 167–175, 2021. doi: 10.1145/3471158.3472257.

Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. Exploiting document-based features for clarification in conversational search. In *Advances in Information Retrieval*, pp. 413–427, 2022. doi: 10.1007/978-3-030-99736-6_28.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. URL https://arxiv.org/abs/2402.03300.

Clemencia Siro, Yifei Yuan, Mohammad Aliannejadi, and Maarten de Rijke. AGENT-CQ: Automatic generation and evaluation of clarifying questions for conversational search with LLMs. *arXiv preprint arXiv:2410.19692*, 2024. URL https://arxiv.org/abs/2410.19692.

Zhenduo Wang and Qingyao Ai. Controlling the risk of conversational search via reinforcement learning. In *Proceedings of the Web Conference 2021*, pp. 1968–1977, 2021. doi: 10.1145/3442381.3449893.

Zhenduo Wang, Yuancheng Tu, Corby Rosset, Nick Craswell, Ming Wu, and Qingyao Ai. Zero-shot clarifying question generation for conversational search. In *Proceedings of the ACM Web Conference 2023*, pp. 3288–3298, 2023. doi: 10.1145/3543507.3583420.

Zeqiu Wu, Yi Luan, Hannah Rashkin, David Reitter, Hannaneh Hajishirzi, Mari Ostendorf, and Gaurav Singh Tomar. CONQRR: Conversational query rewriting for retrieval with reinforcement learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 10000–10014, 2022. doi: 10.18653/v1/2022.emnlp-main.679.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Hamed Zamani, Johanne R. Trippas, Jeff Dalton, and Filip Radlinski. Conversational information seeking. *Foundations and Trends in Information Retrieval*, 17(3–4):244–456, 2023. doi: 10.1561/1500000081.

Qing T. Zeng and Tony Tse. Exploring and developing consumer health vocabularies. *Journal of the American Medical Informatics Association*, 13(1):24–29, 2006. doi: 10.1197/jamia.M1761.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025. URL https://arxiv.org/abs/2506.05176.

Yongfeng Zhang, Xu Chen, Qingyao Ai, Liu Yang, and W. Bruce Croft. Towards conversational search and recommendation. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pp. 177–186, 2018. doi: 10.1145/3269206.3271776.

Ziliang Zhao, Zhicheng Dou, Jiaxin Mao, and Ji-Rong Wen. Generating clarifying questions with web search results. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 234–244, 2022. doi: 10.1145/3477495.3531981.

Ziliang Zhao, Zhicheng Dou, and Yujia Zhou. Generating intent-aware clarifying questions in conversational information retrieval systems. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pp. 3384–3394, 2024. doi: 10.1145/3627673.3679851.

A Additional Experimental Results

This appendix reports additional policy comparisons, parameter settings, and query-level analyses. The results quantify retrieval gains across rounds, discovery settings, and success and failure cases. Implementation details and the mathematical interpretation of THPO are provided in Appendix C.

A.1 Ablation protocols and supplementary controls

Tables 2 and 3 give the discovery–return configurations and single-factor policy interventions, respectively; Appendix C.4 defines the return replacements and retained auxiliary terms. Appendix A.5 supplies the parameter settings and R0–R4 trajectories. Their evaluation-series scope is centralized in Appendix C.5.

A.2 Uncertainty in THPO Comparisons

Figure 5 applies the query-paired bootstrap procedure in Appendix A.6 to Table 3. Differences are Full THPO minus each variant, so positive values favor Full THPO. The intervals describe query-sampling variation conditional on the saved policies and responses.

All four intervals for differences in mean NDCG@10 over R1–R4 lie above zero. For NDCG@10 at R4, the intervals exclude zero for terminal reward and immediate-gain, but include zero for shared scalar return and flat advantage. These intervals support differences in the round-mean metric more consistently than in the R4 metric; their interpretation is subject to the evaluation-series boundaries in Appendix C.5.

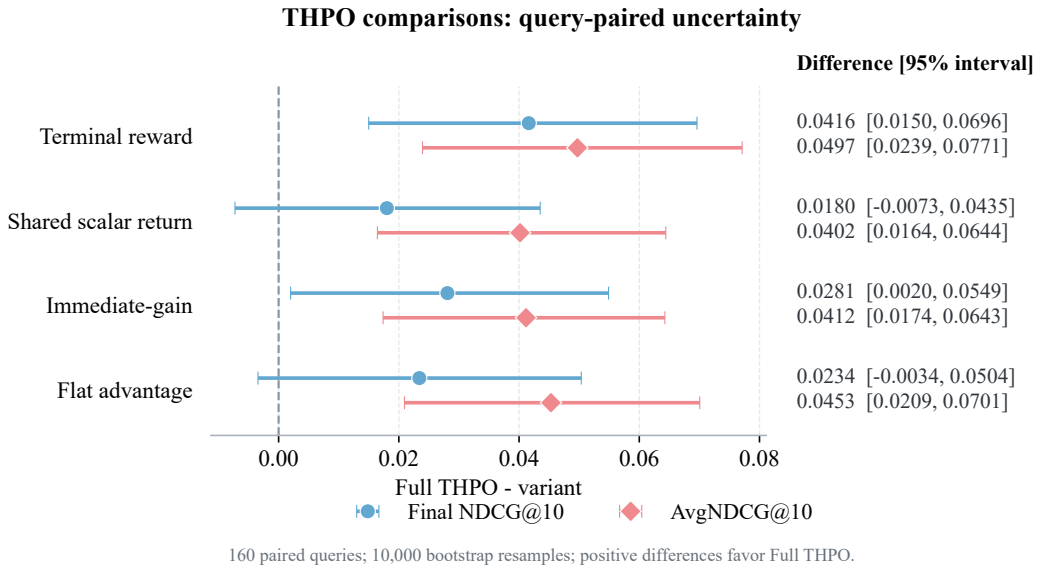

Figure 5: Query-paired uncertainty for Table 3. Points show Full THPO minus each variant; bars show 95% percentile bootstrap intervals, without adjustment for multiple comparisons. The numerical column gives each difference and its interval. Positive differences favor Full THPO.

A.3 Evaluation scope and limitations

The evaluations use simulated users, four clarification rounds, and restricted candidate pools. These settings limit conclusions about real-user answerability, adaptive stopping, and full-corpus retrieval. Average round-level quality corresponds to a uniform stopping assumption rather than an observed user stopping distribution. Evaluation-series differences are specified in Appendix C.5.

CLEAR requires domain-specific codebooks and fact inventories, so discovery depends on retrieved evidence, document representations, and extracted facts. A fixed retriever still entails preprocessing

and clarification-generation costs. Earlier gains in interaction rounds do not establish lower wall-clock latency; human studies, larger retrieval collections, and latency measurements are needed to assess deployment utility.

A.4 Policy–discovery combinations

Table 4 retains the additional combinations outside the main comparison, including SFT and Terminal GRPO (Shao et al., 2024) policies paired with prompt-based discovery. These are not all single-component ablations of CLEAR: changing both the policy checkpoint and the discovery mechanism does not isolate the contribution of facet discovery. Evaluation-series differences are specified in Appendix C.5.

Table 4: Additional policy–discovery combinations on Test160. N denotes NDCG@10 at R4 and A denotes mean NDCG@10 over R1–R4.

Method	LeCaRDv2				TREC-COVID			
	MAP	P@5	N	A	MAP	P@5	N	A
SFT + Prompt	0.448	0.458	0.438	0.438	0.379	0.353	0.448	0.375
Terminal GRPO + Prompt	0.437	0.420	0.423	0.424	0.382	0.358	0.445	0.375
CLEAR policy + Prompt	0.453	0.456	0.445	0.440	0.393	0.366	0.464	0.388

A.5 Complete Parameter Sensitivity

Tables 5–8 report aggregate and round-wise metrics for ten configurations on 160 legal queries over four interaction rounds. CLEAR obtains NDCG@10 at R4 of .5344 and mean NDCG@10 over R1–R4 of .4985. All CLEAR trajectories and query-level diagnostics use the same main-run outputs.

The CLEAR reference uses round-weight schedule $(1, 0.75, 0.5, 0.25)$, prefix schedule $(1, 1, 2, 2)$, 12 candidate facets, and four RQ levels. The alternative round-weight schedules emphasize earlier or later stopping. Prefix schedules, candidate caps, and RQ depth are inference settings; RQ depth is the number of retained code levels, without refitting codebooks. Prefix length is a structural preference, not a direct measure of question specificity.

Table 5: Round-weight schedules on 160 LeCaRDv2 queries. Upper panel: R4 metrics and mean NDCG@10 over R1–R4. Lower panel: NDCG@10 at each round.

Round weights	MAP	P@5	NDCG@10 at R4	Mean NDCG@10 over R1–R4
$(1, 0.75, 0.5, 0.25)$ (CLEAR)	0.5117	0.5501	0.5344	0.4985
$(1, 0.6, 0.3, 0.1)$	0.5014	0.5387	0.5194	0.4719
$(1, 0.9, 0.7, 0.4)$	0.4932	0.5150	0.5012	0.4576

Round weights	R0	R1	R2	R3	R4
$(1, 0.75, 0.5, 0.25)$ (CLEAR)	0.2820	0.4460	0.4923	0.5214	0.5344
$(1, 0.6, 0.3, 0.1)$	0.2820	0.3889	0.4788	0.5004	0.5194
$(1, 0.9, 0.7, 0.4)$	0.2820	0.3796	0.4578	0.4916	0.5012

Early Gains and Trade-offs. Tables 5–8 provide aggregate scores and exact trajectories; Figure 9 visualizes those same measurements. Among the nine Sol-evaluated variants, round-weight schedule $(1, 0.6, 0.3, 0.1)$ has the highest NDCG@10 at R4, whereas RQ depth 2 has the highest mean NDCG@10 over R1–R4.

Interaction Outcomes. Question-acceptance and supported-answer counts must be computed from the same saved dialogue run as the retrieval metrics.

Table 6: Prefix schedules on 160 LeCaRDv2 queries. Upper panel: R4 metrics and mean NDCG@10 over R1–R4. Lower panel: NDCG@10 at each round.

Prefix schedule	MAP	P@5	NDCG@10 at R4	Mean NDCG@10 over R1–R4
$(1, 1, 2, 2)$ (CLEAR)	0.5117	0.5501	0.5344	0.4985
$(1, 1, 1, 1)$	0.4959	0.5150	0.5084	0.4752
$(2, 2, 2, 2)$	0.4916	0.5112	0.4947	0.4426

Prefix schedule	R0	R1	R2	R3	R4
$(1, 1, 2, 2)$ (CLEAR)	0.2820	0.4460	0.4923	0.5214	0.5344
$(1, 1, 1, 1)$	0.2820	0.4016	0.4843	0.5064	0.5084
$(2, 2, 2, 2)$	0.2820	0.3647	0.4363	0.4747	0.4947

Table 7: Candidate counts on 160 LeCaRDv2 queries. Upper panel: R4 metrics and mean NDCG@10 over R1–R4. Lower panel: NDCG@10 at each round.

Candidate cap	MAP	P@5	NDCG@10 at R4	Mean NDCG@10 over R1–R4
12 (CLEAR)	0.5117	0.5501	0.5344	0.4985
4	0.4915	0.5112	0.4997	0.4509
8	0.4955	0.5200	0.5108	0.4735
16	0.4819	0.5025	0.4895	0.4524

Candidate cap	R0	R1	R2	R3	R4
12 (CLEAR)	0.2820	0.4460	0.4923	0.5214	0.5344
4	0.2820	0.3690	0.4518	0.4831	0.4997
8	0.2820	0.4003	0.4806	0.5024	0.5108
16	0.2820	0.3864	0.4515	0.4823	0.4895

Table 8: RQ depths on 160 LeCaRDv2 queries. Upper panel: R4 metrics and mean NDCG@10 over R1–R4. Lower panel: NDCG@10 at each round.

RQ depth	MAP	P@5	NDCG@10 at R4	Mean NDCG@10 over R1–R4
4 (CLEAR)	0.5117	0.5501	0.5344	0.4985
2	0.4961	0.5300	0.5131	0.4758
3	0.4955	0.5175	0.5040	0.4678

RQ depth	R0	R1	R2	R3	R4
4 (CLEAR)	0.2820	0.4460	0.4923	0.5214	0.5344
2	0.2820	0.3969	0.4857	0.5074	0.5131
3	0.2820	0.3911	0.4759	0.5003	0.5040

A.6 Query-Level and Visual Diagnostics

Paired Analysis. For query q and setting v, define $F_q^{(v)} = N_{q,4}^{(v)}$ and $A_q^{(v)} = \frac{1}{4}\sum_{t=1}^{4} N_{q,t}^{(v)}$. A paired difference averages $F_q^{(v)} - F_q^{(0)}$ or $A_q^{(v)} - A_q^{(0)}$ over the same 160 query IDs, where setting 0 is the main-table CLEAR run. Each bootstrap replicate samples 160 query IDs with replacement and retains all rounds and both settings for every sampled query. We use 10,000 replicates with seed

73 and report the 2.5th and 97.5th percentiles. Resampling entire queries preserves within-dialogue dependence. These intervals quantify query-sampling variation conditional on the saved runs.

Figure 7 summarizes these differences. Both fixed-fine intervals exclude zero. For fixed coarse discovery, the intervals for differences in NDCG@10 at R4 ($[-.0529,\ .0010]$) and mean NDCG@10 over R1–R4 ($[-.0470,\ .0004]$) both include zero.

First Attainment. For a descriptive quality threshold η, define

$$T_q(\eta) = \min\{t \in \{0, \ldots, 4\} : N_{q,t} \geq \eta\}, \qquad \min \varnothing = \infty. \tag{9}$$

We retain queries that already satisfy the threshold at R0 and those that never satisfy it. For $\eta = .5$, the main CLEAR run's first-attainment counts at R0–R4 and never are $(29, 45, 20, 13, 5, 48)$, compared with $(29, 31, 29, 10, 6, 55)$ for fixed coarse and $(29, 21, 20, 16, 10, 64)$ for fixed fine. First attainment measures the first crossing, not sustained quality, and the threshold is not a calibrated measure of user satisfaction.

Figure 6 uses the same main-run reference and fixed-prefix controls as Figure 7. First attainment uses an absolute NDCG@10 threshold of .5, retaining R0 and never-reached outcomes. The scatter of mean NDCG@10 over R1–R4 has 88 wins, one tie, and 71 losses for CLEAR versus fixed coarse preferences. The reference therefore improves more queries than it worsens in this comparison, but improvements are not universal and the corresponding difference interval still spans zero.

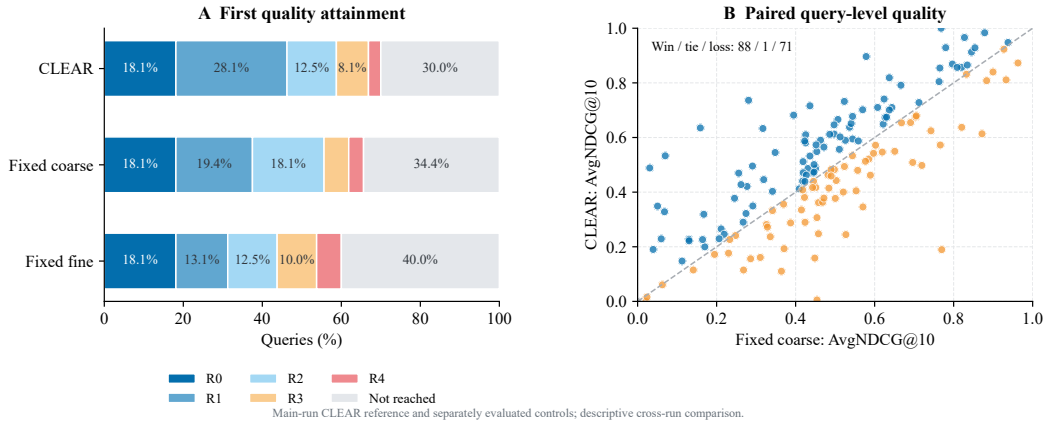

Figure 6: First attainment and query-paired quality. Blue and orange scatter points indicate higher and lower CLEAR scores than fixed coarse discovery, respectively; gray indicates ties. The diagonal denotes equality.

Success and Failure Trajectories. The cases in Figure 8 are selected by the largest positive and negative R0–R4 changes, respectively, rather than by manual selection of appealing dialogues. In the main CLEAR run, query 325 follows $(.0734, 1.0000, 1.0000, 1.0000, 1.0000)$, matching its NDCG@10 at R4 after R1. Query 195 instead follows $(.6984, .1983, .0854, .0991, .0767)$, illustrating that additional interaction can damage an initial ranking with NDCG@10 of .6984. Query 675 improves to .9216 at R3 before falling to .8701 at R4, showing that even a large overall gain need not be monotonic. These traces motivate examining intermediate retrieval states rather than only NDCG@10 at R4.

A.7 Archived RQ comparison

Table 9 compares two archived legal discovery configurations. PRFD adds round-dependent structural scoring, prefix eligibility with fallback, and coverage tracking.

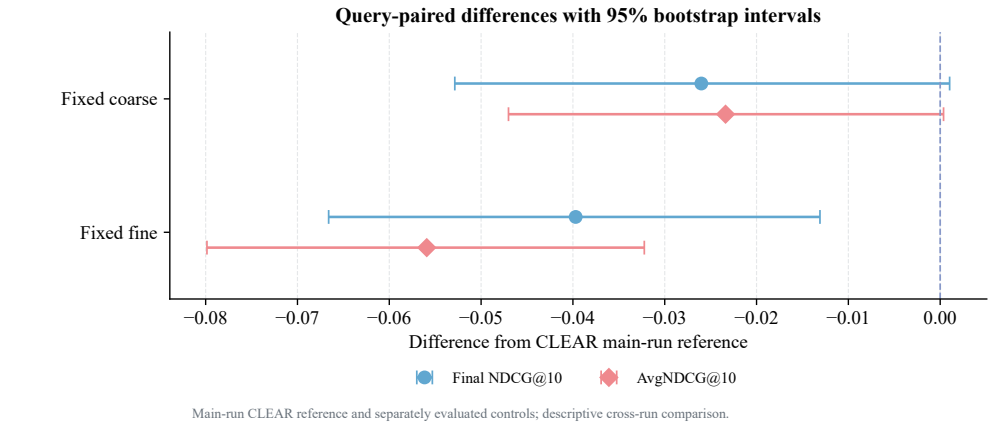

Figure 7: Control differences from the main-run CLEAR reference; negative values favor CLEAR. Intervals are 95% percentile bootstrap intervals from 10,000 paired query resamples (seed 73), conditional on the recorded runs rather than variation across training seeds. Both fixed-fine intervals exclude zero; neither fixed-coarse interval does.

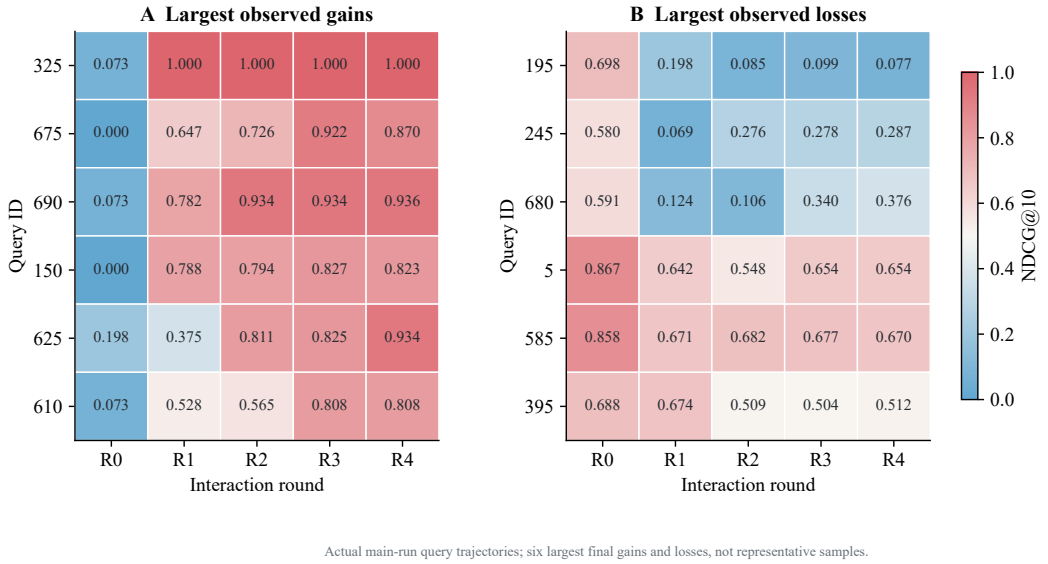

Figure 8: Query trajectories with the six largest increases and six largest decreases in NDCG@10 from R0 to R4 among 160 queries. Colors and annotations show query-level NDCG@10, not individual document ranks.

B External Baseline Adaptations

ClarifyGPT originally addresses requirements clarification for code generation (Mu et al., 2024); our retrieval baseline transfers the clarification idea to document search rather than reproducing its code-consistency mechanism. Our adaptation of SRQG-LTR (Zhao et al., 2022) uses downstream retrieval gains to learn candidate-document-pair preferences. These implementation choices describe the evaluated adaptations rather than the original systems.

DocFeatures extracts nouns from the top-10 retrieved documents using jieba for Chinese and NLTK for English, ranks candidate facets by entropy, and generates questions with the method's facet-question template. MMR-BERT selects questions from a bank constructed exclusively from domain training trajectories. Its BERT cross-encoder is initialized with a domain encoder and trained using weak question supervision from these trajectories. Question selection combines relevance with

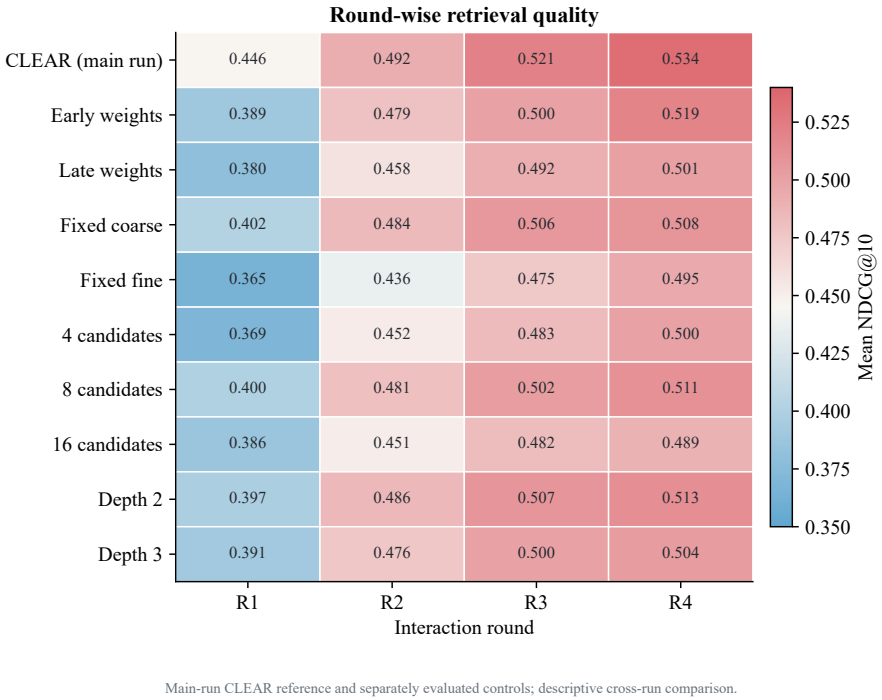

Figure 9: Round-wise NDCG@10 for CLEAR and nine legal parameter settings.

Table 9: Archived legal discovery comparison (160 queries). PRFD is called Round-aware RQ in the original run records. Generation and validation settings belong to the earlier series in Appendix C.5.

Discovery configuration	NDCG@10 at R4	Mean NDCG@10 over R1–R4
Original RQ ordering	0.5148	0.4819
PRFD	0.5344	0.4985

negative feedback from previous interactions. Checkpoints are selected by validation MRR, and the retrieval encoder remains frozen.

AGENT-CQ-Temp uses Qwen3-4B to generate candidate questions through temperature-based sampling at each interaction round and selects among them using a weighted combination of relevance and clarification scores. ZeroShotCQGen uses Qwen3-4B with the author's NeuroLogic constraint-search implementation. Nouns extracted from the visible dialogue supply lexical constraints, with language-specific tokenization and candidate scoring for Chinese and English.

Within each evaluation series, all methods use fixed test queries, candidate pools, and a frozen retriever. Initial retrieval is checked against the R0 reference. The questioning models receive only the visible query, dialogue history, and permitted retrieval evidence. Relevance labels and private user facts are withheld from their inference inputs. Ambiguous duplicate-question cases in legal retrieval are reviewed using the visible context. Response and validation settings are summarized in Appendix C.5.

C Implementation Details and Return-Matrix Expansion

C.1 PRFD Resources and Candidate Construction

Facts and Facets. PRFD selects candidates from a fixed inventory $\mathcal{F}$ of facets, each with a category, label, and example question; discovery does not create new inventory entries or categories.

Legal categories cover characterization, penalty, and procedure; medical categories include population, disease, and study evidence; patent categories include purpose, structure, and operation. Legal facts combine cached structured subfacts with concept-extraction rules. Medical and patent facts use canonical-value matching and retain supporting passages. Medical parent categories come from clustered training-topic information gaps and review of cluster examples; patent parents are predefined. These categories are not semantic labels assigned to RQ codes.

Document Representations. Table 10 separates the representations used for RQ from the retrieval models. The patent RQ representation uses Qwen3-Embedding-8B (Zhang et al., 2025), whereas patent retrieval uses BM25. Codebooks, domain facts, and facet inventories are constructed for each domain. Legal base fitting uses training-split relevance supervision and alignment with annotated legal facts. Medical fitting uses training-pool documents after removing test-pool overlap; patent fitting uses training-query pools. All three resource sets have four RQ levels, but entries per level and facet counts differ. Encoders, codebooks, and domain facts remain fixed during policy learning; test relevance labels and private user descriptions are not online discovery inputs.

Table 10: Domain-specific RQ resources and retrieval models. Levels $\times$ entries per level describes the quantizer, not the retriever.

Domain	RQ representation	Levels $\times$ entries per level	Facets	Categories	Retriever
Legal	SAILER + neural projection	4×64	41	3	SAILER
Medical	MedCPT + neural projection	4×64	30	8	MedCPT
Patent	Qwen3-Embedding-8B	4×32	28	6	BM25

From Document Pairs to Candidates. The provider compares nearby retrieved documents that share a domain context, such as a charge, disease, or IPC subclass, but differ in supported facts. RQ prefixes organize these comparisons, and the inventory maps their differences to candidate facets. Attempted or resolved facets are excluded. For duplicate candidate facets, the highest-ranked supporting pair is retained rather than combining incompatible evidence. Missing information is not treated as a negative fact.

Let A_{t-1} contain attempted facets, regardless of answer availability, and $C_{t-1} \subseteq A_{t-1}$ contain those resolved by informative answers. The state and candidate set are

$$s_t = (x_{t-1}, H_{t-1}, D_{t-1}, A_{t-1}, C_{t-1}, t), \qquad \mathcal{E}_t = \mathcal{C}(s_t) \subseteq \mathcal{F} \setminus A_{t-1}. \tag{10}$$

For document codes $z(d), z(d')$, their shared-prefix length is

$$p(d, d') = \max\big\{\ell \in \{0, \ldots, L\} : z_j(d) = z_j(d') \text{ for all } j \le \ell\big\}. \tag{11}$$

For a candidate e, $p(e)$ is the prefix length of its supporting pair, averaged over available legal code paths. The generic provider stores a divergence feature $\ell(e)$ and ranks candidates by

$$S_t(e) = b(e) + \alpha d(e) + \beta\gamma_t p(e) + \eta\gamma_t \ell(e) - \rho[\tau_t - p(e)]_+, \tag{12}$$

where $[x]_+ = \max(0, x)$. For discrete legal codes, $\ell(e) = p(e)$, reducing the structural terms to $(\beta + \eta)\gamma_t p(e)$. The alternative convention $\ell(e) = p(e) + 1$ adds only a constant within each round; neither convention defines an independent divergence mechanism. The base score $b(e)$ favors visible differences that remain unresolved in the dialogue. For legal narrative facts, it multiplies passage dissimilarity by a discount for similarity to the current context, using character-bigram Jaccard similarity. For legal concept sets, it is the normalized number of unresolved differences. The difference term $d(e)$ uses passage dissimilarity or the number of differing concepts, capped at four and divided by four. Pair-ranking ties use prefix locality, retrieval rank, and rank proximity. These scores measure candidate contrast, not the probability of a user supplying an answer.

The round-aware legal configuration uses prefix preferences $(1, 1, 2, 2)$ with a decreasing structural bonus and fallback when no candidate qualifies. Medical candidates are selected within parent categories. The patent provider keeps at most one per parent and supplies RQ evidence at inference without round-dependent re-ranking.

Question Generation. The policy receives the visible dialogue and candidate IDs, labels, and example questions; the patent evidence interface also exposes paired passages. Its prompt requires selecting one listed facet and asking about one unresolved fact. For example, contrasting document populations can motivate a question about the desired age group, but cannot establish the user's preferred population. Only an informative answer adds that condition to the query. Discovery identifies what to ask; the response determines what is known about the user's information need.

C.2 Data Construction and Interaction Inputs

Splits and Initial Queries. LeCaRDv2 retains the official 640 training and 160 test queries, with frozen query-ID lists dividing the training portion into 590 training and 50 validation queries.

Table 11 distinguishes original queries or topics from the constructed query instances. TREC-COVID Round 5 contains 50 topics, each expanded into 16 variants. Splitting by original topic yields 36/4/10 supervised training/validation/test topics and 576/64/160 query instances. The validation topics are 6, 8, 41, and 48; the test topics are 5, 10, 12, 17, 20, 24, 26, 27, 30, and 46. All remaining topics form the supervised training set. Query IDs retain the original-topic identifier, and the three partitions have no overlapping query or topic IDs. Medical RL uses the 640 non-test query variants; the supervised validation partition is not a separate RL holdout.

Table 11: Data construction for the policy resources. Query counts denote distinct initial-query instances, not individual question–answer training examples. The patent source count refers to selected topics.

Source	Original units	Queries per unit	Train / validation / test
LeCaRDv2	800 case queries	1	590 / 50 / 160
TREC-COVID Round 5	50 topics	16	576 / 64 / 160
CLEF-IP 2010 PAC	200 selected topics	4	576 / 64 / 160

The expanded medical queries are generated with `gpt-5.6-terra` from each original topic and seed-paper evidence. The prompt requests natural Chinese queries with varied wording and user perspectives while preserving the information need; a grounded four-round plan supplies distinct clarification factors. Patent construction selects 160 official training and 40 official test topics from CLEF-IP 2010 PAC, with four variants per topic. Holding out 16 training topics gives 144/16/40 source topics and the query counts in Table 11; train and test patent families are disjoint. Patent initial queries are generated from source technical descriptions with identifiers masked and without relevance labels or candidate rankings. They omit details that subsequent answers can supply, and hidden facts must be supported by source passages. Initial queries are generated once and frozen, not regenerated for each method.

Candidate Pools and Query Updates. Legal pools reuse the official per-query candidate lists and judgments, excluding unavailable files and retaining up to 100 documents. In the expanded medical resources, MedCPT ranks indexed, judged documents using the original topic and fixes the top 100; all 16 topic variants share this pool. Judgment membership determines eligibility, but relevance grades do not determine this initial ordering. Patent pools retain available relevant documents, excluding the source patent and its family. Pool membership is unchanged across rounds. The policy sees only the initial query, visible dialogue, and permitted evidence; accepted informative answers update the retrieval context. Relevance judgments and private user facts are not policy inputs.

C.3 Explicit return allocation

The survival weights $w_k = \sum_{s=k}^{T} p_s$ relate retrieval increments to expected quality at stopping. For the first decision, exchanging the summations gives

$$\begin{aligned} G_{i,1} &= \sum_{s=1}^{T} p_s \sum_{k=1}^{s} (N_{i,k} - N_{i,k-1}) \\ &= \sum_{s=1}^{T} p_s N_{i,s} - N_{i,0} = U_{\mathrm{avg},i} - N_{i,0} \quad \text{if } p_s = 1/T. \end{aligned} \tag{13}$$

The stopping distribution specifies the optimization objective; training uses full-horizon trajectories. Uniform stopping over four rounds yields $w = (1, 0.75, 0.5, 0.25)$.

For an individual trajectory, this allocation takes the explicit form

$$\underbrace{\begin{bmatrix} G_{i,1} \\ G_{i,2} \\ G_{i,3} \\ G_{i,4} \end{bmatrix}}_{\mathbf{g}_i} = \underbrace{\begin{bmatrix} 1.00 & 0.75 & 0.50 & 0.25 \\ 0 & 0.75 & 0.50 & 0.25 \\ 0 & 0 & 0.50 & 0.25 \\ 0 & 0 & 0 & 0.25 \end{bmatrix}}_{W} \underbrace{\begin{bmatrix} \Delta_{i,1} \\ \Delta_{i,2} \\ \Delta_{i,3} \\ \Delta_{i,4} \end{bmatrix}}_{\boldsymbol{\delta}_i}. \tag{14}$$

Row t of W specifies which retrieval increments contribute to the return for decision t: current and subsequent increments are included, while past increments are excluded. Stacking the transposed return vectors for all B trajectories gives

$$\mathcal{R} = \begin{bmatrix} \mathbf{g}_1^\top \\ \vdots \\ \mathbf{g}_B^\top \end{bmatrix} = \begin{bmatrix} G_{1,1} & G_{1,2} & G_{1,3} & G_{1,4} \\ \vdots & \vdots & \vdots & \vdots \\ G_{B,1} & G_{B,2} & G_{B,3} & G_{B,4} \end{bmatrix} \in \mathbb{R}^{B \times 4}. \tag{15}$$

C.4 Policy-update details

Why Later Decisions Exclude Past Gains. Assume the stopping distribution is fixed independently of the sampled trajectory and $\sum_{s=1}^{T} p_s = 1$. Expanding each N_s as $N_0 + \sum_{k=1}^{s} \Delta_k$ gives Equation 13 without requiring nonnegative increments. At decision t, earlier increments $\Delta_1, \ldots, \Delta_{t-1}$ are determined before the current action. Under a score-function gradient for the unnormalized objective, their expected contribution is zero conditional on the history:

$$\mathbb{E}\left[\nabla_\theta \log \pi_\theta(a_t, e_t \mid o_t, \mathcal{E}_t) \sum_{k<t} w_k \Delta_k \,\middle|\, o_t, \mathcal{E}_t, \Delta_{<t}\right] = 0. \tag{16}$$

Removing past gains therefore yields a reward-to-go learning signal. This argument explains temporal allocation before advantage normalization; it does not prove that hierarchical standardization and length-normalized token scoring are an unbiased gradient estimator of expected stopping utility. The implemented update is the surrogate specified in Section 3.3.

Earlier Gains and Temporary Regressions. Moving an increment of size $\epsilon > 0$ from a later round v to an earlier round $u < v$, while holding other increments fixed, changes G_1 by $\epsilon(w_u - w_v) \geq 0$. The final score is unchanged, but stopping points between u and $v - 1$ benefit. Conversely, a negative increment is retained rather than clipped away. A temporary decrease can be outweighed by subsequent improvements, which is why the objective does not impose monotonic retrieval quality. The stopping-weight interpretation assumes a prescribed distribution, not a learned model of actual user stopping behavior.

Controlled Return Interventions. The main retrieval return in THPO is $G_{i,t} = \sum_{k=t}^{T} w_k \Delta_{i,k}$. Shared scalar return assigns $G_{i,1}$ to every decision in trajectory i, keeping the trajectory-level retrieval objective but removing round-specific allocation. Immediate-gain assigns $w_t \Delta_{i,t}$, retaining the current-round survival weight but removing subsequent retrieval gains from the main return. Terminal-only feedback assigns the final retrieval score to each decision; subtracting the shared initial score is a query-specific constant before normalization. These interventions concern the main retrieval return: when auxiliary constraints are retained, their future answerability and terminal-quality terms remain. Thus, immediate-gain does not mean that every part of its training signal is local to the current round. The flat-advantage control instead retains temporal returns and replaces hierarchical comparisons with ordinary group standardization.

Hierarchical comparisons. At round t, we compare trajectories sharing an initial query, their selected categories, and distinct facets within each category; subsequent histories may differ. The trajectory component is $a_{i,t}^{\mathrm{traj}} = Z(\widetilde{G}_{\cdot,t})_i$. We average returns for each selected category, standardize these means across represented categories, and assign the category score to its member trajectories.

Within each category, we likewise standardize the mean returns of its distinct selected facets to obtain $a_{i,t}^{\text{facet}}$. A comparison contributes zero when it contains fewer than two represented items or has constant values; in particular, the within-category component is zero when fewer than two distinct facets are represented. Equation 6 standardizes the sum of all three components separately for each round.

Token scoring and reference regularization. For serialized output tokens $v_{i,t,1:m_{i,t}}$, the implementation masks prompt and padding tokens and computes

$$\ell_\theta(i,t) = \frac{1}{m_{i,t}} \sum_{j=1}^{m_{i,t}} \log \pi_\theta(v_{i,t,j} \mid o_{i,t}, \mathcal{E}_{i,t}, v_{i,t,<j}). \tag{17}$$

Thus, the advantage weights the mean score of the generated facet–question output. The frozen supervised model supplies $\ell_{\text{ref}}(i,t)$, with regularization

$$\Omega_{\text{ref}} = \frac{1}{BT} \sum_{i,t} \big(\exp(h_{i,t}) - h_{i,t} - 1\big), \qquad h_{i,t} = \text{clip}(\ell_{\text{ref}}(i,t) - \ell_\theta(i,t), -10, 10). \tag{18}$$

Equation 18 is a clipped reference-score penalty, weighted by β_{ref}, rather than an exact token-level KL divergence. The update uses a group-relative score-function surrogate without old-policy ratio clipping.

Constraint and anchor updates. Constraint multipliers are adjusted using smoothed training statistics and projected into bounded nonnegative ranges, increasing in response to violations. Scheduled supervised anchors and recovery anchors triggered by a low valid-question rate preserve the output protocol. Anchors use environment-supported training targets and replace the reinforcement update at the selected steps. These auxiliary targets can use private training facts; inference observations cannot.

C.5 Training and evaluation reproducibility

The implementation separates facet discovery, question generation, user-response simulation, query updating, and retrieval evaluation. Training starts from a supervised Qwen3-4B policy and updates the questioning model using retrieval feedback, with a frozen supervised reference and auxiliary constraints. The retriever, document encoder, and discovery resources remain fixed. Table 12 summarizes the legal sensitivity training configuration; it does not describe every main-table run.

Evaluation workflow. Each evaluation starts from a fixed initial query and candidate pool. The policy asks a clarification question, a simulated user answers from the underlying factual description, and accepted information updates the query before retrieval. Private user facts are hidden from the questioning policy. We evaluate four clarification rounds and summarize both NDCG@10 at R4 and mean NDCG@10 over R1–R4.

Configuration and comparison scope. The main legal results and supplementary controls come from separate evaluation series. Simulator and response-validation settings also differ across some series; these comparisons describe the evaluated configurations rather than isolating every component effect. Baseline adaptations are described in Appendix B.

Table 12: Training configuration for the legal sensitivity experiments. Scheduled anchor frequency is not the realized update fraction after recovery anchors. This table does not describe all main-table runs.

Setting	Value
Model	
Policy	Qwen3-4B
Initialization	Supervised legal question-generation checkpoint
Reference policy	Frozen supervised initialization
Precision	BF16
Optimization	
Optimizer	AdamW
Learning rate	2×10^{-7}
Adam coefficients	$(0.9, 0.999)$
Weight decay	0.01
Training queries	368
Epochs	3
Seed	73
Data-parallel workers	4
Query microbatch per worker	1
Gradient accumulation steps	1
Trajectory group per query	12
Distributed optimizer	ZeRO stage 2; optimizer states on GPU
Gradient norm clipping	1.0
Reference-score penalty weight β_{ref}	0.03
Returns and constraints	
Interaction rounds	4
Round-weight schedule	$(1, 0.75, 0.5, 0.25)$
Round answerability targets κ_t	$(0.70, 0.40, 0.18, 0.09)$
Expected final-quality target ν	0.04
Dual learning rate	0.02
EMA coefficient	0.98
Maximum constraint multipliers	5.0
Scheduled SFT anchor interval	Every 4 steps
Recovery valid-question threshold	0.50