CLOSED-LOOP BRIDGE RETRIEVAL FOR MULTI-HOP QUESTION ANSWERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-hop retrieval-augmented generation has to connect supporting evidence scattered across documents, and the right next retrieval target usually becomes clear only after earlier evidence accumulates. Iterative RAG systems adapt retrieval through query rewriting, subquestion generation, or sufficiency assessment, but this adaptation stays open-loop at the bridge that routes the next hop: a candidate intermediate entity steers the following retrieval step, and whatever evidence comes back is folded into the context without asking again whether the bridge–gap binding behind it is actually supported, so a wrong route is carried forward instead of revoked. What remains open is when retrieval should commit to an intermediate bridge and how it should recover once later evidence contradicts that commitment. We propose **CLBR-RAG** (Closed-Loop Bridge Retrieval), which closes the loop by turning each bridge commitment into a provisional control state that can be revoked. A commitment binds an evidence-supported intermediate entity to an unresolved information gap. Complementary lexical and semantic queries keep coverage over candidate bridges before any commitment is made; the Evidence Gate admits a source-grounded binding only when it matches an active gap and agrees with the accumulated context, and a deterministic composer then instantiates its targeted next-hop query. The evidence returned by that branch decides whether the commitment survives: unsupported branches are retracted, supported ones keep their evidence, and BROAD exploration resumes when no targeted branch is accepted. Validated cross-document bridge relations are carried into a compact, provenance-preserving context for final reasoning. Across three multi-hop benchmarks and TriviaQA as a broader control, CLBR-RAG is consistently effective. Code will be released upon publication.

1 INTRODUCTION

Retrieval-augmented generation (RAG) grounds large language models in external and updatable knowledge for knowledge-intensive question answering (Lewis et al., 2020). In multi-hop question answering, supporting facts are distributed across documents, while the next information need often becomes clear only after earlier evidence has been retrieved and interpreted (Yang et al., 2018; Ho et al., 2020). Multi-hop RAG is therefore not a one-shot retrieve-then-generate procedure. It is a sequential evidence-acquisition process in which each retrieval round decides both what information remains missing and how that information should be sought.

A particularly consequential failure occurs when the system commits to an unsupported intermediate bridge. Consider the question "Where was the director of *Aylwin* born?", whose required chain is film $\rightarrow$ director $\rightarrow$ birthplace. If retrieval incorrectly associates the film with a distractor director, the following birthplace query can retrieve locally relevant, provenance-preserving evidence for that wrong entity. The resulting answer is unsupported with respect to the original question, even though each individual retrieval step appears successful. The challenge is thus not only to obtain more evidence, but also to determine when an intermediate entity is sufficiently supported to control the next retrieval step.

Figure 1 contrasts an evidence-supported retrieval path, an unverified bridge commitment, and evidence-grounded control before bridge-targeted routing. Existing iterative RAG systems address important parts of this process through interleaved reasoning and retrieval, query rewriting, reflec-

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.

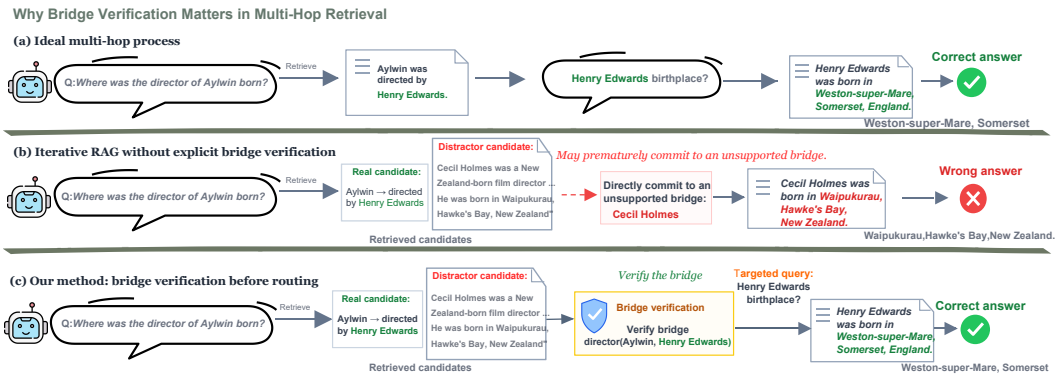


Figure 1: Motivating example of bridge verification in multi-hop retrieval. (a) An evidence-supported intermediate entity enables a targeted next-hop query. (b) Iterative RAG without explicit bridge verification can prematurely route retrieval through a distractor entity, causing later evidence to follow the wrong path. (c) CLBR-RAG verifies a candidate bridge before routing targeted retrieval.

tion signals, complexity-adaptive retrieval, and real-time information-need triggering (Trivedi et al., 2023; Ma et al., 2023; Asai et al., 2024; Jeong et al., 2024; Su et al., 2024). S2G-RAG judges evidence sufficiency and predicts unresolved information gaps for subsequent retrieval (Li et al., 2026). STRIDE constructs entity-agnostic reasoning plans and uses dependency-aware supervision to schedule subquestions and rewrite failed retrieval queries (Chen et al., 2026). A sufficiency judgment or gap prediction may identify what remains missing or suggest an intermediate target, but does not by itself establish that a particular source-grounded bridge–gap binding should remain authoritative for targeted retrieval. We study when a particular bridge–gap binding may control retrieval and when evidence returned by that branch should revoke the provisional commitment.

We formulate bridge commitment as a provisional, evidence-grounded, recoverable retrieval-control process. Until evidence supports a binding, retrieval preserves alternative bridges and open gaps. An admitted binding can instantiate a targeted query while retaining unresolved constraints from the question. The commitment remains falsifiable: if returned evidence fails to support its bridge–gap binding, that branch is retracted while supported branches retain their evidence. BROAD recovery follows only when no targeted branch is accepted.

We instantiate this recoverable commitment lifecycle in **CLBR-RAG** (**C**losed-**L**oop **B**ridge **R**etrieval). Before commitment, complementary lexical and semantic queries constructed from the question, unresolved gaps, and accumulated context preserve coverage over candidate bridges. Once source-grounded candidates emerge, explicit source, gap, and history checks admit provisional bindings for bridge-targeted retrieval. Returned evidence validates or retracts each branch independently; BROAD exploration resumes when no targeted branch is accepted. After retrieval terminates, CLBR-RAG refines the accumulated evidence and integrates validated cross-document bridge relations into a compact, provenance-preserving representation for downstream reasoning.

We evaluate CLBR-RAG on the full development sets of four open-domain QA benchmarks: three multi-hop benchmarks and TriviaQA as a broader control. The results demonstrate consistent effectiveness, and controlled analyses examine the roles of complementary retrieval, evidence-grounded closed-loop bridge control, and cross-round evidence organization.

Our contributions are threefold:

- **Coverage-preserving evidence acquisition.** Query-aware lexical and semantic requests combine the question, open gaps, and accumulated evidence to cover unresolved needs and alternative bridge paths before commitment.

- **Recoverable bridge commitments.** The Gate's deterministic source, gap, and history checks admit Anchor-Judge bridge–gap candidates as provisional retrieval-control states. TARGETED evidence validates or retracts each; BROAD resumes if all fail.

- **Validated bridge memory for final reasoning.** We compress and refine cross-round evidence, then use a separate validated ledger to inject only provenance-supported cross-document relations into a compact final context.

2 Related Work

2.1 Retrieval-Augmented Generation

Retrieval-augmented generation (RAG) augments language models with external knowledge retrieved at inference time, allowing them to access factual and updatable information beyond parametric memory (Lewis et al., 2020). Subsequent work extends the basic retrieve-then-generate paradigm through query rewriting, reflection-based retrieval and generation control, and adaptive retrieval-mode selection (Ma et al., 2023; Asai et al., 2024; Jeong et al., 2024). These directions make RAG more responsive to different information needs. Multi-hop question answering, however, requires coordinating evidence acquisition across rounds because later information needs emerge from earlier evidence.

2.2 Multi-Hop Retrieval-Augmented Generation

Multi-hop RAG extends retrieval into an iterative process in which entities, relations, and intermediate facts discovered in earlier stages can guide subsequent search. Existing work has addressed different aspects of this setting, including evidence acquisition, retrieval control, and evidence organization.

Evidence Acquisition. Iterative retrieval methods typically discover cross-document evidence through intermediate reasoning, question decomposition, or query reformulation. IR-CoT alternates chain-of-thought reasoning with retrieval (Trivedi et al., 2023), query-rewriting methods reconstruct retrieval requests from intermediate states (Ma et al., 2023), and ReSP combines repeated retrieval, summarization, and planning to accumulate multi-hop evidence (Jiang et al., 2025). These approaches use intermediate states to update retrieval targets and uncover multi-hop evidence. When multiple information gaps or potential bridge paths coexist, preserving complementary candidate coverage before a bridge is sufficiently determined remains challenging.

Retrieval Control. Related work also addresses retrieval triggering, stopping, and strategy adaptation. Self-RAG controls retrieval and generation through reflection signals (Asai et al., 2024), Adaptive-RAG selects among retrieval modes according to question complexity (Jeong et al., 2024), DRAGIN triggers retrieval from real-time information needs during generation (Su et al., 2024), and S2G-RAG explicitly judges evidence sufficiency and predicts unresolved information gaps to guide iterative retrieval (Li et al., 2026). STRIDE constructs entity-agnostic reasoning plans and uses a dependency-aware supervisor to schedule retrieval or inference and rewrite failed queries (Chen et al., 2026). CLBR-RAG models evidence-grounded bridge commitments as explicit, recoverable retrieval-control states.

Evidence Organization. Multi-round retrieval also introduces cross-round evidence accumulation and context redundancy. RECOMP, LongLLMLingua, and LLMLingua-2 reduce irrelevant or redundant context through compression (Xu et al., 2024; Jiang et al., 2024; Pan et al., 2024). ALCE and RARR address citation attribution and evidence revision, respectively (Gao et al., 2023b;a), while FActScore evaluates factual precision at the atomic-fact level (Min et al., 2023). These works support context compression, attribution, revision, and factuality evaluation. However, unified organization of evidence accumulated across retrieval rounds remains an important issue in multi-hop RAG.

Overall, existing work has advanced multi-hop evidence discovery, adaptive retrieval control, and context management. CLBR-RAG connects these stages through a recoverable bridge-commitment state that mediates between accumulated evidence and targeted next-hop retrieval.

3 METHOD

CLBR-RAG treats an evidence-supported bridge–gap binding as a provisional retrieval-control state. Before such a binding is admitted, retrieval preserves coverage over alternative bridges. An admitted binding may direct a TARGETED branch, but the returned evidence must support the same binding for it to persist; otherwise the branch is retracted and BROAD exploration can resume. Figure 2 shows this commitment lifecycle and the evidence organization used for final reasoning. Appendix A specifies the retrieval configuration, full inference procedure, and evidence-injection format.

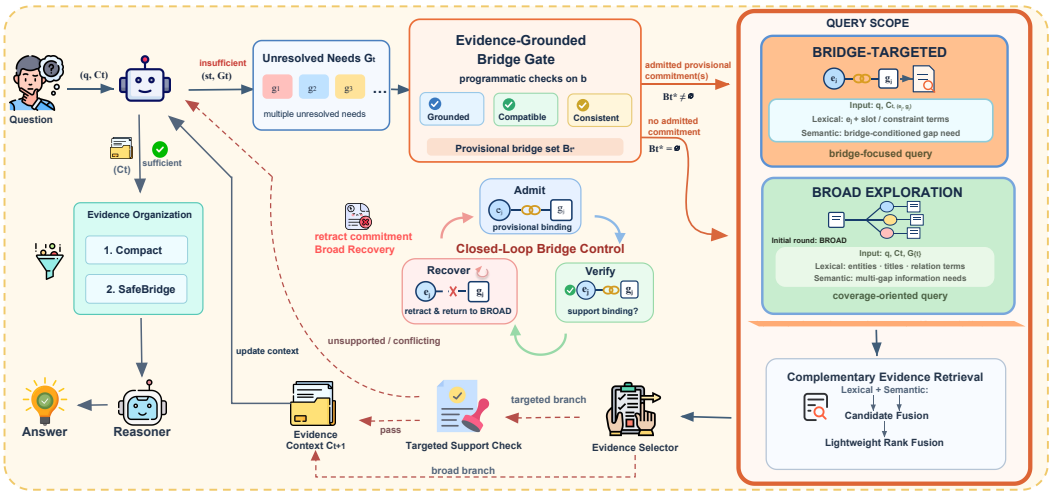


Figure 2: Overview of CLBR-RAG. Before commitment, complementary retrieval preserves coverage over candidate bridges. During commitment, evidence-grounded control admits provisional bindings, verifies targeted branches, and retracts unsupported ones. For final reasoning, accepted evidence is compressed and validated bridge relations are retained.

3.1 RETRIEVAL AS RECOVERABLE BRIDGE COMMITMENT

Given a question q and corpus $\mathcal{D}$, CLBR-RAG maintains an evidence context C_t, a ledger H_t of validated bridge commitments and their supporting source pointers, and a retry record F_t of failed targeted-witness keys. The Base Judge provides evidence-sufficiency and unresolved-gap estimates following the structured judging formulation of Li et al. (2026), together with a semantic retrieval request:

$$\text{BaseJudge}(q, C_t) \rightarrow (s_t, G_t, Q_t^S). \tag{1}$$

Here, $s_t \in \{\text{sufficient}, \text{insufficient}\}$ indicates whether C_t supports answer generation, G_t contains unresolved information needs, and Q_t^S is the next semantic request. Retrieval stops when $s_t = \text{sufficient}$ or the round budget is exhausted; Appendix A.3 gives the terminal procedure and gap schema.

When retrieval continues, the next scope is $m_{t+1} \in \{\text{BROAD}, \text{TARGETED}\}$. The first round is BROAD. Later rounds may follow a bridge binding $b = (i_s, i_g, e)$, where i_s points to a newly selected source sentence mentioning entity e and i_g points to an active gap that e can advance. Thus, G_t specifies what remains unresolved, whereas an admitted binding specifies which grounded entity can direct a TARGETED path for that gap. The Anchor Judge proposes candidate bindings; the Evidence Gate admits a candidate into the provisional set B_t^* only when its source, gap dependency, and history pass the checks in Section 3.3. An admitted candidate still requires execution checks before it directs TARGETED retrieval. For an executed branch, the lifecycle is candidate $\rightarrow$ provisional $\rightarrow$ validated (in B_t^+ and then H_{t+1}) or retracted (with a failed witness recorded in F_{t+1}).

We use the *Aylwin* question in Figure 1 as a running example. After BROAD retrieval selects the sentence *Aylwin was directed by Henry Edwards*, the director is source-grounded but his birthplace remains unresolved. Table 1 records the decision state after pre-target Gate admission and before the targeted branch returns. If that branch returns sourced evidence for Henry Edwards's birthplace, its

binding enters B_t^+ and then H_{t+1}. If the branch fails its support check, its witness key enters F_{t+1} and the binding gains no authority over later retrieval. This transition makes the bridge binding, rather than a generated entity mention alone, the state that controls retrieval scope.

Table 1: Aylwin state snapshot after BROAD evidence and pre-target Gate admission.

State	Role	Running example
C_t	Accepted sentence evidence	Aylwin was directed by Henry Edwards.
G_t	Unresolved information needs	The director's birthplace.
H_t	Previously validated bindings	$\varnothing$ before a target branch returns.
F_t	Failed targeted-witness keys	$\varnothing$ in this successful path.
B_t^*	Gate-admitted provisional bindings	Henry Edwards bound to the birthplace gap.
B_t^+	Validated target bindings	$\varnothing$ before TARGETED retrieval.

3.2 Coverage Before Commitment

BROAD retrieval explores active gaps while keeping alternative bridge candidates available. Its lexical query retains names, titles, and relation terms from the question, gaps, and accumulated context; its semantic query is the Base Judge's request:

$$\left(Q_{t+1}^{B,\mathrm{lex}}, Q_{t+1}^{B,\mathrm{sem}}\right) = \left(\mathrm{Concat}_B(q, G_t, C_t),\ Q_t^S\right). \tag{2}$$

BM25 and E5 return complementary lexical and semantic ranked lists. Both BROAD retrieval and the later TARGETED branches use the same Reciprocal Rank Fusion (RRF) backend:

$$\mathrm{RRF}_{t+1}^x(d) = \sum_{\pi \in \Pi_{t+1}^x} \frac{1}{k_{\mathrm{rrf}} + \mathrm{rank}_\pi(d)}, \qquad x \in \{B, T\}. \tag{3}$$

The Evidence Selector extracts sentence-level evidence from the fused candidates for the active gaps. This acquisition step supplies source-bound candidate bridges while retaining coverage until the evidence supports a narrower commitment. The same total retrieval budget is shared across targeted branches; Section 3.3 describes how their queries are constructed after admission.

3.3 Evidence-Grounded Commitment, Verification, and Recovery

After the initial BROAD round, an unresolved gap alone does not determine the next retrieval path. The controller tests whether newly admitted evidence supports a specific bridge binding eligible to direct the next step. Let ΔE_{t-1} be the evidence added in the preceding update. When $s_t = \mathrm{insufficient}$, the Anchor Judge proposes a routing preference and source-bound candidate bindings:

$$(a_t, \widetilde{B}_t) = \mathrm{AnchorJudge}(q, C_t, G_t, \Delta E_{t-1}, Q_t^S), \tag{4}$$

where $a_t \in \{\mathrm{BROAD}, \mathrm{TARGETED}\}$. For a proposed $b = (i_s, i_g, e)$, i_s references $s = \Delta E_{t-1}[i_s]$ and i_g references the active gap $g = G_t[i_g]$. The Evidence Gate admits it only when source grounding, bridge compatibility, and history consistency all hold:

$$\mathcal{G}(b; \Delta E_{t-1}, G_t, H_t, F_t) = \Phi_{\mathrm{ground}}(b) \wedge \Phi_{\mathrm{compat}}(b) \wedge \Phi_{\mathrm{consist}}(b). \tag{5}$$

The three predicates are fixed, deterministic program checks rather than additional model judgments:

$$\Phi_{\mathrm{ground}}(b) = \mathrm{ValidSource}(i_s, \Delta E_{t-1}) \wedge \mathrm{LiteralMatch}(\mathrm{Norm}(e), s), \tag{6}$$

$$\Phi_{\mathrm{compat}}(b) = \mathrm{ActiveGap}(i_g, G_t) \wedge \mathrm{DependencyWitness}(e, g, s), \tag{7}$$

$$\Phi_{\mathrm{consist}}(b) = \mathrm{NoBindingConflict}(e, g, H_t) \wedge \mathrm{NoFailedWitness}(b, F_t). \tag{8}$$

The first check ties e to a selected source sentence; the second requires an upstream link to an active gap; the third excludes conflicting bindings and previously failed identical witnesses. Failed-witness identity is $\kappa(b) = (\mathrm{DocID}(s), \mathrm{SentID}(s), \mathrm{Norm}(g), \mathrm{Norm}(e))$. Thus, the provisional set is

$$B_t^* = \{b \in \widetilde{B}_t : \mathcal{G}(b; \Delta E_{t-1}, G_t, H_t, F_t) = 1\}. \tag{9}$$

Each admitted $b_j = (i_{s,j}, i_{g,j}, e_j)$ receives a separate lexical–semantic query pair:

$$\left(Q^{T,\mathrm{lex}}_{t+1,j}, Q^{T,\mathrm{sem}}_{t+1,j}\right) = \left(\mathrm{Concat}_T(q, e_j, g_j, R^*_t),\, Q^T_{t+1,j}\right), \tag{10}$$

where R^*_t carries residual constraints from the original question and $Q^T_{t+1,j}$ is the deterministic semantic request for that bridge and gap. Each admitted binding keeps a separate branch, so competing bridges are not merged into one query and returned evidence remains attributable to its provisional commitment. Query-envelope validation retains executable candidates in $B^Q_t \subseteq B^*_t$. For later rounds, the executed mode is

$$m_{t+1} = \begin{cases} \text{TARGETED}, & a_t = \text{TARGETED}, \quad B^Q_t \neq \varnothing, \\ & \mathrm{TaskClosure}(B^Q_t, G_t, C_t) = 1, \quad \rho_t \neq \text{FORCE-BROAD}, \\ \text{BROAD}, & \text{otherwise.} \end{cases} \tag{11}$$

Here ρ_t is the recovery flag, and TaskClosure requires at least one executable binding in B^Q_t for each active gap in G_t. Failed conditions execute BROAD without promoting an unexecuted candidate. Pre-gate rejections may be reconsidered after new evidence arrives; post-target failed-witness keys in F_t are not retried. Appendix A.3 specifies the guards and shared branch budget.

An executed TARGETED branch remains provisional until its selected evidence $E^T_{t+1,j}$ supports the same bridge–gap binding:

$$\mathrm{Accept}_T(e_j, g_j, E^T_{t+1,j}) = \mathbb{I}\big[E^T_{t+1,j} \neq \emptyset \ \wedge\ \mathrm{Prov}(E^T_{t+1,j}) = 1 \ \wedge\ \mathrm{Support}(E^T_{t+1,j}, e_j, g_j) = 1\big]. \tag{12}$$

The support predicate requires a provenance-bound sentence that mentions the normalized bridge entity, witnesses the target relation or slot, and satisfies branch coverage. Accepted bindings, their branch-indexed provenance, and the evidence increment are

$$\begin{aligned} B^+_t &= \{b_j \in B^Q_t : \mathrm{Accept}_T(e_j, g_j, E^T_{t+1,j}) = 1\}, \\ \mathcal{P}^+_{t+1} &= \{(b_j, E^T_{t+1,j}) : b_j \in B^+_t\}, \\ E^+_{t+1} &= \bigcup_{b_j \in B^+_t} E^T_{t+1,j}. \end{aligned} \tag{13}$$

When $B^+_t \neq \varnothing$, only evidence from these branches updates C_{t+1}, while their bindings and source pointers update the persistent ledger H_{t+1}; unsuccessful branches contribute only their $\kappa(b)$ keys to F_{t+1}. Partial success therefore preserves supported branches without endorsing the others. When $B^+_t = \varnothing$, the controller retracts all provisional commitments, leaves their gaps open, and forces the next retrieval round to BROAD.

3.4 Validated Bridge Memory for Final Reasoning

Only bindings in the validated ledger H_t are eligible for explicit injection as cross-document relations; each retains pointers to its source and returned evidence. The working context C_t contains accepted BROAD evidence and evidence from supported TARGETED branches, while F_t records failed targeted witnesses for later routing decisions. Let $\mathcal{P}_{t+1}$ denote the accepted provenance-bearing input: selected BROAD evidence in a BROAD round, or the validated TARGETED provenance $\mathcal{P}^+_{t+1}$ in a successful TARGETED round. The context update performs exact cross-round deduplication:

$$C_{t+1} = \mathrm{Update}(C_t, \mathcal{P}_{t+1}), \qquad \Delta E_t = C_{t+1} \setminus C_t. \tag{14}$$

The newly admitted ΔE_t is available to the next routing decision. At termination, slot-grounded evidence compression following Li et al. (2026) creates base records. Cross-round refinement removes redundancy while retaining a valid-provenance witness for every retained gap or validated commitment:

$$V^{\mathrm{base}}_T = \mathrm{BaseCompress}(C_T), \qquad V_T = \mathrm{Compact}(C_T; G_T, H_T) = \mathrm{Refine}(V^{\mathrm{base}}_T; G_T, H_T). \tag{15}$$

SafeBridge integrates only the validated ledger bindings with unambiguous support into the final evidence view:

$$M_T = \mathrm{SafeBridge}(V_T, H_T). \tag{16}$$

The Reasoner thus receives compact sentence-level evidence together with supported cross-document bridge relations. Appendix A.4 specifies the refinement procedure and serialization format.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets and metrics. We evaluate the full development sets of TriviaQA, HotpotQA, 2Wiki, and MuSiQue under open-domain retrieval. For the first three, we use the S2G-RAG retrieval settings (Li et al., 2026) for controlled comparison; MuSiQue uses a shared corpus built following IR-CoT's open-domain conversion (Trivedi et al., 2023) (Appendix A). HotpotQA, 2Wiki, and MuSiQue test multi-hop evidence use; TriviaQA serves as a broader open-domain control. We report Exact Match (EM), token-level F1, and Recall@6 (R@6) for evidence-acquisition diagnostics.

Baselines and implementation. We compare against NaiveGen, Standard RAG (Lewis et al., 2020), IR-CoT (Trivedi et al., 2023), Self-RAG (Asai et al., 2024), RAG-Critic (Dong et al., 2025), SIM-RAG (Yang et al., 2025), STRIDE (Chen et al., 2026), and S2G-RAG (Li et al., 2026). All methods use Qwen3.5-9B for answer generation, at most four retrieval rounds, and six final documents per round; each retains its own query construction and retrieval control. For controlled comparison, CLBR-RAG's Base Judge follows S2G-RAG's evidence-sufficiency and unresolved-gap assessment. The CLBR-RAG pipeline preserves candidate coverage through complementary lexical–semantic retrieval, admits and verifies provisional bridge bindings with recovery, and refines evidence with validated bridge injection. Its Base Judge and Anchor Judge use Qwen3.5-4B; the Evidence Selector and Reasoner use Qwen3.5-9B. BM25–E5 results are fused with RRF before the top six documents are passed to evidence selection. Unless varied, ablations share the corpus, retrieval budget, Evidence Selector, Reasoner, and decoding configuration. Appendix A gives the full configuration and inference procedure.

4.2 MAIN RESULTS

Table 2 reports results on the full development sets. CLBR-RAG achieves the highest EM and F1 on every benchmark. Against the strongest baseline on each dataset (S2G-RAG on TriviaQA and STRIDE on the three multi-hop datasets), its EM/F1 gains are 2.3/2.6, 2.2/3.6, 4.1/2.4, and 5.0/6.5 points on TriviaQA, HotpotQA, 2Wiki, and MuSiQue, respectively. The largest gains occur on MuSiQue, consistent with the value of keeping intermediate bridge choices revisable when evidence must be connected across hops.

Table 2: Main results on full development sets (EM/F1, %); bold marks the best in each column.

	TriviaQA		HotpotQA		2Wiki		MuSiQue	
Method	EM	F1	EM	F1	EM	F1	EM	F1
NaiveGen	58.0	63.6	20.5	28.9	25.8	30.2	3.4	11.5
Standard RAG (NeurIPS'20)	71.8	79.3	37.6	49.8	37.0	42.1	10.1	20.2
IR-CoT (ACL'23)	74.3	81.2	48.6	61.3	49.8	57.0	20.3	30.3
Self-RAG (ICLR'24)	72.5	79.7	39.9	51.8	37.5	45.6	13.1	21.7
RAG-Critic (ACL'25)	73.3	81.2	47.9	59.3	41.5	49.9	19.6	28.5
SIM-RAG (SIGIR'25)	73.9	79.9	45.5	58.0	44.8	52.4	19.2	27.6
STRIDE (SIGIR'26)	74.6	81.3	52.3	64.1	52.2	61.7	27.9	37.2
S2G-RAG (ACL'26)	75.9	82.6	50.4	62.7	49.6	56.3	23.2	33.1
CLBR-RAG (Ours)	**78.2**	**85.2**	**54.5**	**67.7**	**56.3**	**64.1**	**32.9**	**43.7**

4.3 ABLATION AND MECHANISM ANALYSIS

Table 3 ablates the key mechanisms that instantiate the CLBR-RAG commitment lifecycle on all four full development sets. All subsequent mechanism analyses use the fixed 3,000-example diagnostic set defined in Appendix A; scores are macro-averaged across its three datasets. The analyses follow the commitment lifecycle: candidate coverage, recoverable targeted routing, and validated evidence for reasoning.

Mechanism-level ablation. Table 3 removes one mechanism supporting the recoverable commitment lifecycle at a time. Without complementary retrieval, average EM/F1 falls by 3.5/4.8 points across four datasets; without bridge control, by 2.4/2.7; without evidence organization, by 1.4/2.2. All three mechanisms contribute to the full system; the control analysis below isolates the central claim about evidence-grounded, verifiable, and recoverable commitments.

Table 3: Mechanism-level ablation on four full development sets (EM/F1, %). Each row removes one mechanism supporting the recoverable commitment lifecycle.

	MuSiQue		2Wiki		HotpotQA		TriviaQA	
Variant	EM	F1	EM	F1	EM	F1	EM	F1
w/o Complementary Retrieval	28.2	36.6	52.7	58.4	51.4	63.9	75.7	82.5
w/o Closed-Loop Bridge Control	30.3	40.9	53.3	60.7	52.6	65.0	76.2	83.4
w/o Evidence Organization	31.7	41.2	54.6	61.1	53.2	65.8	76.7	83.8
Full CLBR-RAG	**32.9**	**43.7**	**56.3**	**64.1**	**54.5**	**67.7**	**78.2**	**85.2**

4.3.1 Complementary Retrieval for Candidate-Bridge Coverage

Table 4 separates signal fusion from query construction under a shared BM25–E5–RRF backend and budget. BM25 and E5 are single-view baselines; Vanilla Hybrid fuses their original-query ranks, while Query-Aware Hybrid uses the question, unresolved gaps, and accumulated evidence. Relative to E5, Vanilla Hybrid gains 1.4 R@6, 1.4 EM, and 3.7 F1 with 1.61 s lower latency and similar rounds (3.66 versus 3.70). Query-aware construction adds 2.5 R@6, 2.1 EM, and 3.1 F1 over Vanilla Hybrid; rounds fall to 2.61 (1.05 fewer, 28.7%) for 0.16 s more latency.

Table 4: Mean evidence-acquisition metrics on the fixed diagnostic set (3,000 examples); latency and rounds are per example.

Method	R@6 ↑	EM ↑	F1 ↑	Latency (s) ↓	Rounds ↓
BM25	72.2	45.4	52.5	**8.09**	3.81
E5	78.8	45.7	52.6	12.67	3.70
Vanilla Hybrid	80.2	47.1	56.3	11.06	3.66
Query-Aware Hybrid	**82.7**	**49.2**	**59.4**	11.22	**2.61**

4.3.2 Evidence-Grounded Closed-Loop Bridge Control

Table 5 compares direct commitment, pre-target Gate admission, and recovery. Direct commitment trails BROAD-only by 0.3 EM/0.1 F1, showing that TARGETED retrieval alone does not account for the gain and can slightly hurt when it narrows around an unscreened entity. Pre-target admission adds 1.7 EM/2.1 F1 over direct commitment, and recovery adds a further 0.3 EM/0.5 F1. In the stage-wise diagnostics (Appendix B.4), only 52.3% of initial TARGETED decisions pass pre-target admission, so the Gate rejects 47.7% before they consume a retrieval hop. Among admitted branches, 82.9% remain supported after retrieval, while the remaining 17.1% are retracted by post-retrieval validation. Together, these results show that the gain comes from constraining targeted retrieval with evidence-grounded admission, post-retrieval validation, and recovery rather than from direct commitment itself.

Figure 3 summarizes shared-prefix replay and on-policy routing. Among bridge-triggered states, TARGETED is better in 76.0% of cases, tied in 13.0%, and worse in 11.0%. When TARGETED is better, it gains 3.8 EM/4.4 F1 on average; when BROAD is better, TARGETED loses 1.1 EM/1.5 F1. On policy, 70.2% continue BROAD, 24.7% yield supported TARGETED branches, and 5.1% trigger fallback, motivating coverage and recovery.

Table 5: Cumulative ablation of closed-loop bridge control on the fixed diagnostic set (3,000 examples).

Variant	Direct	Pre-gate	Recover	EM ↑	F1 ↑
BROAD-only	–	–	–	47.5	56.9
+ Direct commitment	✓	–	–	47.2	56.8
+ Pre-target Gate	✓	✓	–	48.9	58.9
Full control	✓	✓	✓	**49.2**	**59.4**

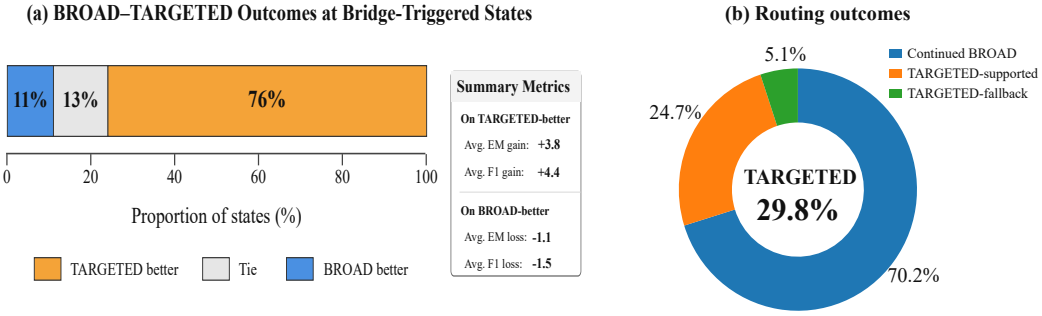


Figure 3: Routing diagnostics. (a) Shared-prefix replay compares BROAD and TARGETED continuations at bridge-triggered states. (b) On-policy outcomes under CLBR-RAG.

4.3.3 Provenance-Preserving Evidence Organization

Table 6 evaluates evidence organization cumulatively. Base compression reduces context from 2692.1 to 686.2 tokens (74.5%) and improves EM/F1 by 2.8/3.0 over raw evidence. Refinement removes another 122.7 tokens and improves both metrics; integrating verified bridge relations adds only 8.8 tokens but yields a further 0.5 EM and 1.8 F1. The final representation thus preserves answer-relevant cross-round relations while remaining compact.

Table 6: Cumulative ablation of provenance-preserving evidence organization on the fixed diagnostic set (3,000 examples).

Variant	Base Comp.	Refine	Bridge Rel.	Tokens ↓	EM ↑	F1 ↑
Raw evidence	–	–	–	2692.1	45.1	53.6
+ Base compression	✓	–	–	686.2	47.9	56.6
+ Refinement	✓	✓	–	**563.5**	48.7	57.6
Full organization	✓	✓	✓	572.3	**49.2**	**59.4**

5 Conclusion

Multi-hop RAG must decide when an intermediate entity has enough evidence to constrain the next retrieval step. CLBR-RAG makes this bridge commitment explicit and recoverable: it preserves candidate coverage before commitment, verifies targeted branches against their bound gaps, retracts unsupported branches with BROAD recovery, and retains only validated relations for final reasoning. On the full development sets of four QA benchmarks, CLBR-RAG achieves the highest EM and F1 among the evaluated methods. Controlled analyses support the roles of coverage, branch validation, and evidence organization in this commitment lifecycle.

AI use statement

Generative AI tools were used to assist with drafting portions of the manuscript, language polishing, improving clarity of presentation, and retrieving and discovering literature and materials relevant to

this work. AI tools were also used to organize and summarize experimental results produced by the authors and to assist in presenting these results in tables, figures, and accompanying textual descriptions. We reviewed all AI-assisted work. All literature and citations were verified by the authors, and all reported experimental data and numerical results were obtained from actual experiments rather than generated or fabricated by AI. AI-generated content was used only as auxiliary support. All research decisions and final manuscript content remained with the authors, who take responsibility for the accuracy of the final work, including all text, claims, and artifacts produced with the aid of generative AI.

Reproducibility Statement

To facilitate reproduction of our results, we provide the relevant implementation and experimental details throughout the main paper and appendices. The CLBR-RAG framework is described in Section 3, while the experimental setup and evaluation protocol are provided in Section 4. The main experiments use Qwen3.5-4B for the Base Judge and Anchor Judge components and Qwen3.5-9B for the Evidence Selector and downstream Reasoner. Appendix A further documents the datasets, retrieval corpora, implementation configurations, and detailed inference procedure. Appendix C lists the Base Judge, Anchor Judge, and Evidence Selector system prompts, while Appendix B reports additional robustness and diagnostic experiments. All experiments are conducted on publicly available benchmarks under the configurations specified in the paper and appendices. The source code will be released upon publication.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/25f7be9694d7b32d5cc670927b8091e1-Abstract-Conference.html.

Wei Chen, Lili Zhao, Zhi Zheng, Huijun Hou, and Tong Xu. STRIDE: Strategic iterative decision-making for retrieval-augmented multi-hop question answering. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 202–212. ACM, 2026. doi: 10.1145/3805712.3809703. URL https://doi.org/10.1145/3805712.3809703.

Guanting Dong, Jiajie Jin, Xiaoxi Li, Yutao Zhu, Zhicheng Dou, and Ji-Rong Wen. RAG-critic: Leveraging automated critic-guided agentic workflow for retrieval augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3551–3578. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.179. URL https://aclanthology.org/2025.acl-long.179/.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. RARR: Researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 16477–16508. Association for Computational Linguistics, 2023a. doi: 10.18653/v1/2023.acl-long.910. URL https://aclanthology.org/2023.acl-long.910/.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 6465–6488. Association for Computational Linguistics, 2023b. doi: 10.18653/v1/2023.emnlp-main.398. URL https://aclanthology.org/2023.emnlp-main.398/.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6609–6625. International Committee on Computational Linguistics, 2020. doi: 10.18653/v1/2020.coling-main.580. URL https://aclanthology.org/2020.coling-main.580/.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong Park. Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 7036–7050. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.naacl-long.389. URL https://aclanthology.org/2024.naacl-long.389/.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. LongLLMlingua: Accelerating and enhancing LLMs in long context scenarios via prompt compression. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1658–1677. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.91. URL https://aclanthology.org/2024.acl-long.91/.

Zhouyu Jiang, Mengshu Sun, Lei Liang, and Zhiqiang Zhang. Retrieve, summarize, plan: Advancing multi-hop question answering with an iterative approach. In *Companion Proceedings of the ACM on Web Conference 2025*, pp. 1677–1686. ACM, 2025. doi: 10.1145/3701716.3716889. URL https://doi.org/10.1145/3701716.3716889.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Kuttler, Mike Lewis, Wen-tau Yih, Tim Rocktaschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Minghan Li, Junjie Zou, Xinxuan Lv, Chao Zhang, and Guodong Zhou. S2G-RAG: Structured sufficiency and gap judging for iterative retrieval-augmented QA. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 25846–25862. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.1185. URL https://aclanthology.org/2026.acl-long.1185/.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. Query rewriting in retrieval-augmented large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5303–5315. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.322. URL https://aclanthology.org/2023.emnlp-main.322/.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12076–12100. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.741. URL https://aclanthology.org/2023.emnlp-main.741/.

Zhuoshi Pan, Qianhui Wu, Huiqiang Jiang, Menglin Xia, Xufang Luo, Jue Zhang, Qingwei Lin, Yuqing Yang, Chin-Yew Lin, H. Vicky Zhao, Lili Qiu, and Dongmei Zhang. Llmlingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 963–981. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-acl.57. URL https://aclanthology.org/2024.findings-acl.57/.

Weihang Su, Yichen Tang, Qingyao Ai, Zhijing Wu, and Yiqun Liu. DRAGIN: Dynamic retrieval augmented generation based on the real-time information needs of large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12991–13013. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.702. URL https://aclanthology.org/2024.acl-long.702/.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long*

Papers), pp. 10014–10037. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.acl-long.557. URL https://aclanthology.org/2023.acl-long.557/.

Fangyuan Xu, Weijia Shi, and Eunsol Choi. RECOMP: Improving retrieval-augmented LMs with context compression and selective augmentation. In *International Conference on Learning Representations*, pp. 43478–43502, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/bda88ed2892f5e61c9a9bf215c566913-Paper-Conference.pdf.

Diji Yang, Linda Zeng, Jinmeng Rao, and Yi Zhang. Knowing you don't know: Learning when to continue search in multi-round RAG through self-practicing. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1305–1315. ACM, 2025. doi: 10.1145/3726302.3730018. URL https://doi.org/10.1145/3726302.3730018.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2369–2380. Association for Computational Linguistics, 2018. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259/.

A Experimental Details and Inference Procedure

A.1 Datasets and Retrieval Corpora

Main experiments use the full development sets of TriviaQA, HotpotQA, 2Wiki, and MuSiQue under an open-domain retrieval protocol. For TriviaQA, HotpotQA, and 2Wiki, we follow the retrieval settings used by S2G-RAG (Li et al., 2026) and use the corresponding shared retrieval corpora. This keeps the retrieval environment aligned with the primary baseline.

MuSiQue is an additionally evaluated multi-hop benchmark. In its original reading-comprehension formulation, each question is paired with a 20-paragraph candidate context containing supporting and non-supporting evidence. Following the open-domain conversion protocol of IR-CoT (Trivedi et al., 2023), we construct one shared retrieval corpus by pooling all supporting and non-supporting paragraphs associated with questions in the training, development, and test splits. After deduplication, the resulting corpus contains 139,416 paragraphs. At inference time, each MuSiQue question retrieves from this shared corpus rather than from its question-specific 20-paragraph candidate set. These corpus choices are fixed within each benchmark and shared by the controlled retrieval comparisons.

For analyses reported on the diagnostic set, we randomly sample 1,000 development examples from each of MuSiQue, 2Wiki, and HotpotQA using a fixed seed, yielding 3,000 examples in total. This fixed set is shared by the retrieval, control, evidence-organization, efficiency, robustness, and routing analyses and is reported separately from the full-development results.

A.2 Implementation Details

Table 7 summarizes the pure BM25–E5–RRF configuration used for CLBR-RAG's retrieval analysis. The Base Judge determines evidence sufficiency, unresolved gaps, and its native search request. The Anchor Judge proposes BROAD or TARGETED routing together with source-bound anchors for the latter; the Evidence Gate then admits only candidates that pass the fixed source-grounding, bridge-compatibility, and history-consistency checks. A deterministic Anchor Query Composer subsequently instantiates the lexical and semantic request templates for each admitted targeted branch. The Base Judge and Anchor Judge use Qwen3.5-4B with deterministic decoding at temperature 0 and top-p 1; the Evidence Selector and Reasoner use Qwen3.5-9B, and the query composer is programmatic with no model decoding.

Table 7: Pure BM25–E5–RRF configuration used for CLBR-RAG's retrieval analysis.

Component	Setting
Query mode	BROAD: programmatic lexical + Base Judge semantic; TARGETED: deterministic branch queries
Retrievers	BM25 and `e5-base-v2`
Fusion	Reciprocal rank fusion with $k_{\mathrm{rrf}} = 60$
Candidate budget	Top-6 candidates from each retriever; at most 12 unique documents before fusion
Final retrieval budget	Top-6 fused documents
Maximum retrieval turns	4, followed by one terminal Base Judge call without retrieval
Cross-turn duplicate removal	Enabled
Reranker scoring	Disabled
Evidence Selector	Qwen3.5-9B; sentence-level selection with provenance retention
Safe-Bridge	Enabled before the final Reasoner call

For an ordinary query, BM25 and `e5-base-v2` each contribute at most six candidates. Their union is scored by Eq. 3 with $k_{\mathrm{rrf}} = 60$, after which the fused list is truncated to six documents. The smoothing constant is independent of the $6 + 6$ candidate budget. For multiple targeted branches, the same total retrieval budget is allocated across branches without changing k_{rrf}.

A.3 INFERENCE PROCEDURE

The procedure separates candidate admission from round execution. $\mathrm{EvidenceGate}$ first forms the provisional-commitment set B_t^* from source-bound candidate bindings. For a TARGETED preference, the deterministic query composer instantiates branch queries, after which query-envelope validation and $\mathrm{TaskClosure}$ apply. $\mathrm{TaskClosure}$ passes only if each active gap in G_t is referenced by at least one valid-envelope candidate in B_t^Q. If no executable branch remains, the controller executes BROAD retrieval. Thus, the guards can redirect a round without promoting a rejected candidate or adding an unvalidated commitment to the ledger.

Algorithm 1 CLBR-RAG inference with evidence-grounded bridge control

Require: question q, corpus $\mathcal{D}$, maximum retrieval turns T

Ensure: answer $\hat{y}$, terminal sufficiency status s_T, and provenance-preserving terminal evidence view

$C_0 \leftarrow \varnothing$, $H_0 \leftarrow \varnothing$, $F_0 \leftarrow \varnothing$, $\Delta E_{-1} \leftarrow \varnothing$, and $\rho_0 \leftarrow$ FORCE-BROAD
for $t \leftarrow 0$ **to** $T-1$ **do**
  $(s_t, G_t, Q_t^S) \leftarrow \mathrm{BaseJudge}(q, C_t)$
  if $s_t =$ sufficient **then**
    $M_t \leftarrow \mathrm{SafeBridge}(\mathrm{Compact}(C_t; G_t, H_t), H_t)$; **return** $\mathrm{Reasoner}(q, M_t)$
  end if
  $B_t^* \leftarrow \varnothing$
  if $t = 0$ **or** $\rho_t =$ FORCE-BROAD **then** $m_{t+1} \leftarrow$ BROAD
  else
    $(a_t, \widetilde{B}_t) \leftarrow \mathrm{AnchorJudge}(q, C_t, G_t, \Delta E_{t-1}, Q_t^S)$
    for each $\tilde{b} \in \widetilde{B}_t$ **do**
      $r_{\tilde{b}}^{\mathrm{pre}} \leftarrow \mathrm{EvidenceGate}(\tilde{b}, \Delta E_{t-1}, G_t, H_t, F_t)$; **if** $r_{\tilde{b}}^{\mathrm{pre}}$.pass **then** $B_t^* \leftarrow B_t^* \cup \{\tilde{b}\}$
    end for
    if $a_t =$ TARGETED $\wedge\ |B_t^*| \geq 1$ **then** $m_{t+1} \leftarrow$ TARGETED **else** $m_{t+1} \leftarrow$ BROAD
  end if
  if $m_{t+1} =$ TARGETED **then**
    $B_t^Q \leftarrow \varnothing$
    for each $b \in B_t^*$ **do**
      $(Q_b^{\mathrm{lex}}, Q_b^{\mathrm{sem}}) \leftarrow \mathrm{Compose}(q, G_t, C_t, b)$
      if $\mathrm{ValidEnvelope}(Q_b^{\mathrm{lex}}, Q_b^{\mathrm{sem}}, b, G_t)$ **then** $B_t^Q \leftarrow B_t^Q \cup \{b\}$
    end for
    if $B_t^Q = \varnothing$ **or** $\mathrm{TaskClosure}(B_t^Q, G_t, C_t) = 0$ **then** $m_{t+1} \leftarrow$ BROAD
  end if
  end if
  if $m_{t+1} =$ TARGETED **then**
    $(k_b)_{b \in B_t^Q} \leftarrow \mathrm{AllocateBudget}(6, B_t^Q)$
    for each $b \in B_t^Q$ **do**
      $(Q_b^{\mathrm{lex}}, Q_b^{\mathrm{sem}}) \leftarrow \mathrm{Compose}(q, G_t, C_t, b)$
      $L_B \leftarrow \mathrm{BM25}(Q_b^{\mathrm{lex}}, \mathcal{D}, k_b)$; $L_E \leftarrow \mathrm{E5}(Q_b^{\mathrm{sem}}, \mathcal{D}, k_b)$

$D_{t+1,b} \leftarrow \mathrm{Top}k_b(\mathrm{RRF}(L_B, L_E; k_{\mathrm{rrf}} = 60))$

$E^T_{t+1,b} \leftarrow \mathrm{EvidenceSelector}(q, \{g_b\}, D_{t+1,b})$

end for

$B^+_t \leftarrow \{b \in B^Q_t : \mathrm{TargetedSupportCheck}(b, E^T_{t+1,b}, G_t) = 1\}$

if $B^+_t \neq \varnothing$ **then**

$\mathcal{P}^+_{t+1} \leftarrow \{(b, E^T_{t+1,b}) : b \in B^+_t\}$

$(C_{t+1}, \Delta E_t) \leftarrow \mathrm{DeduplicateAndUpdate}(C_t, \mathcal{P}^+_{t+1})$

$H_{t+1} \leftarrow H_t \cup \mathcal{P}^+_{t+1}$; $F_{t+1} \leftarrow F_t \cup \{\kappa(b) : b \in B^Q_t \setminus B^+_t\}$; $\rho_{t+1} \leftarrow \mathrm{AUTO}$

else

$C_{t+1} \leftarrow C_t$; $H_{t+1} \leftarrow H_t$; $F_{t+1} \leftarrow F_t \cup \{\kappa(b) : b \in B^Q_t\}$; $\Delta E_t \leftarrow \varnothing$; $\rho_{t+1} \leftarrow \mathrm{FORCE\text{-}BROAD}$

end if

else

$L_B \leftarrow \mathrm{BM25}(\mathrm{Lexical}(q, G_t, C_t), \mathcal{D}, 6)$; $L_E \leftarrow \mathrm{E5}(Q^S_t, \mathcal{D}, 6)$

$D_{t+1} \leftarrow \mathrm{Top6}(\mathrm{RRF}(L_B, L_E; k_{\mathrm{rrf}} = 60))$

$\widetilde{E}_{t+1} \leftarrow \mathrm{EvidenceSelector}(q, G_t, D_{t+1})$

$(C_{t+1}, \Delta E_t) \leftarrow \mathrm{DeduplicateAndUpdate}(C_t, \widetilde{E}_{t+1})$; $H_{t+1} \leftarrow H_t$; $F_{t+1} \leftarrow F_t$; $\rho_{t+1} \leftarrow \mathrm{AUTO}$

end if

end for

$(s_T, G_T, \cdot) \leftarrow \mathrm{BaseJudge}(q, C_T)$ *(terminal assessment; no retrieval)*

$M_T \leftarrow \mathrm{SafeBridge}(\mathrm{Compact}(C_T; G_T, H_T), H_T)$; **return** $\mathrm{Reasoner}(q, M_T)$ with terminal status s_T

Controller checks. Algorithm 1's controller checks are deterministic and use no additional model calls. `EvidenceGate` implements Eq. 5 through Eqs. (6)–(8). `TargetedSupportCheck` implements Eq. 12: returned provenance-bearing evidence must mention the normalized bridge entity, witness the bound relation or slot, and satisfy branch coverage. `ValidEnvelope` and `TaskClosure` are execution guards that neither admit nor validate bridge commitments.

A.4 Evidence Organization and Bridge Injection Format

The Evidence Selector returns globally indexed sentence identifiers, which the runtime deterministically maps back to text and titles. At termination, BaseCompress converts the retained evidence into slot-grounded records following Li et al. (2026). Refine then canonicalizes each record by its normalized entity, slot, and value; removes a record only when another valid-provenance record duplicates that fact; and restores any record that is the sole witness for an active gap or validated ledger binding. It never creates new evidence or bridge relations. SafeBridge subsequently adds a relation only from a validated ledger binding with unambiguous supporting records. Before final reasoning, it serializes the question, refined evidence, and verified bridge relations in the following injection format. The `e` fields index entries in `evidence`; bridge relations are included only after their supporting evidence has passed the targeted support check.

```
{
  "question": "<original_question>",
  "evidence": [
    {
      "text": "<selected_evidence_sentence_0>",
      "title": "<source_title_0>"
    },
    {
      "text": "<selected_evidence_sentence_1>",
      "title": "<source_title_1>"
    },
    {
      "text": "<selected_evidence_sentence_2>",
      "title": "<source_title_2>"
    }
  ],
  "bridge": [
    {
      "r": "same_slot",
      "s": "<normalized_information_slot_0>",
      "e": [0, 2]
    },
    {
      "r": "linked_slots",
      "s": [
        "<information_slot_1>",
        "<information_slot_0>"
      ],
      "e": [1, 2]
    }
  ]
}
```

Each evidence item retains the text and source title used by the Reasoner. A `same_slot` record identifies evidence entries that normalize to the same information slot, while a `linked_slots` record expresses a verified cross-document relation between slots. A failed TARGETED branch contributes neither its temporary evidence nor its bridge relation to this view.

B Additional Results

B.1 End-to-End Efficiency and Retrieval-Round Analysis

Table 8: End-to-end efficiency and retrieval-round analysis on the fixed diagnostic set (3,000 examples). Latency is the mean wall-clock time per question in seconds, and Rounds counts retrieval rounds actually executed.

	S2G-RAG (BM25)		S2G-RAG (E5)		STRIDE		CLBR-RAG	
Dataset	Latency ↓	Rounds ↓	Latency ↓	Rounds ↓	Latency ↓	Rounds ↓	Latency ↓	Rounds ↓
MuSiQue	15.81	3.508	18.22	4.000	19.32	**2.61**	**10.39**	3.340
2Wiki	**7.58**	2.601	18.79	3.994	10.85	**2.10**	7.83	2.370
HotpotQA	**6.37**	2.328	33.57	3.988	23.48	**2.02**	15.45	2.131
Overall	**9.92**	2.812	23.52	3.994	17.88	**2.24**	11.22	2.614

Table 8 compares full-pipeline latency and executed retrieval rounds on this diagnostic set. STRIDE uses the fewest rounds on every dataset (2.24 overall); CLBR-RAG has lower mean latency (11.22 versus 17.88 s, a 37.2% reduction) and is fastest on MuSiQue (10.39 s). CLBR-RAG also improves both measures over S2G-RAG (E5), reducing latency from 23.52 to 11.22 s and rounds from 3.994 to 2.614. S2G-RAG (BM25) has the lowest overall latency (9.92 s), while CLBR-RAG completes fewer rounds (2.614 versus 2.812). Retrieval-round count alone therefore does not determine end-to-end latency.

B.2 Backbone Robustness Across Model Families

To test whether the improvement from closed-loop bridge retrieval transfers beyond the Qwen3.5 family used in the main experiments, we replace the Qwen3.5-4B/9B stack with Qwen3-4B/8B and evaluate STRIDE, S2G-RAG, and CLBR-RAG on the same fixed 3,000-example diagnostic set: 1,000 examples each from 2Wiki, HotpotQA, and MuSiQue. Table 9 shows that CLBR-RAG achieves the highest EM and F1 on every dataset under this alternative backbone. Relative to the strongest competing method, STRIDE, its gains are 1.4/2.4 EM/F1 on 2Wiki, 1.7/1.6 on HotpotQA, and 2.8/3.1 on MuSiQue, yielding average gains of 2.0 EM and 2.4 F1. Although the absolute scores are lower than in the main Qwen3.5 setting, CLBR-RAG's consistent lead over both iterative baselines indicates that closed-loop bridge retrieval is not tied to the Qwen3.5 model family.

Table 9: Backbone robustness on the fixed 3,000-example diagnostic set using the Qwen3-4B/8B stack. CLBR-RAG achieves the highest EM and F1 on every dataset.

	2Wiki		HotpotQA		MuSiQue		Avg.	
Method	EM	F1	EM	F1	EM	F1	EM	F1
STRIDE	44.3	50.2	48.1	57.5	21.9	30.3	38.1	46.0
S2G-RAG	40.4	46.5	46.6	56.0	19.1	27.7	35.4	43.4
CLBR-RAG	**45.7**	**52.6**	**49.8**	**59.1**	**24.7**	**33.4**	**40.1**	**48.4**

B.3 Model Capacity Robustness

We separately vary the capacity of the retrieval controller and the downstream Reasoner on the fixed 3,000-example diagnostic set. This separation tests whether the evidence-acquisition and organization pipeline remains effective when either model-side component changes.

B.3.1 Base Judge Capacity

Table 10 varies the Base Judge backbone while fixing the downstream Reasoner to Qwen3.5-9B; all other components follow the main configuration. Average EM/F1 rises from 45.4/55.0 with Qwen3.5-2B to 49.2/59.4 with Qwen3.5-4B and 51.6/61.7 with Qwen3.5-9B. The larger gain occurs from 2B to 4B, whereas moving from 4B to 9B yields a smaller additional improvement. The default Qwen3.5-4B Base Judge therefore captures much of the controller-side gain at an intermediate model capacity.

Table 10: Base Judge capacity robustness on the fixed 3,000-example diagnostic set. The Reasoner is fixed to Qwen3.5-9B. Bold denotes the highest result in each column; the default Base Judge is indicated explicitly.

	2Wiki		HotpotQA		MuSiQue		Avg.	
Base Judge	EM	F1	EM	F1	EM	F1	EM	F1
Qwen3.5-2B	55.0	62.4	54.5	65.6	26.6	36.8	45.4	55.0
Qwen3.5-4B (default)	57.0	64.5	58.3	69.9	32.3	43.9	49.2	59.4
Qwen3.5-9B	**58.5**	**66.5**	**60.6**	**71.3**	**35.8**	**47.2**	**51.6**	**61.7**

B.3.2 Reasoner Capacity

Table 11 fixes the Base Judge to Qwen3.5-4B and varies the downstream Reasoner. Scaling the Reasoner from Qwen3.5-9B to Qwen3.5-27B raises the average score from 49.2/59.4 to 54.6/64.5 EM/F1, a gain of 5.4 EM and 5.1 F1. The improvement appears on all three datasets and is largest on 2Wiki. Thus, the evidence acquired and organized by CLBR-RAG remains useful as downstream reasoning capacity increases.

Table 11: Reasoner capacity robustness on the fixed 3,000-example diagnostic set. The Base Judge is fixed to Qwen3.5-4B. Bold denotes the highest result in each column; the default Reasoner is indicated explicitly.

	2Wiki		HotpotQA		MuSiQue		Avg.	
Reasoner	EM	F1	EM	F1	EM	F1	EM	F1
Qwen3.5-9B (default)	57.0	64.5	58.3	69.9	32.3	43.9	49.2	59.4
Qwen3.5-27B	**65.4**	**73.1**	**62.0**	**73.6**	**36.5**	**46.9**	**54.6**	**64.5**

Together, these controlled studies separate controller sensitivity from downstream reasoning capacity. A stronger Base Judge improves evidence-sufficiency assessment and gap prediction, while a stronger Reasoner further exploits the cross-document evidence supplied by the same retrieval-control mechanism.

B.4 Closed-Loop Bridge-Control Diagnostics

Using the same fixed 3,000-example diagnostic set, we examine the controller beyond final answer accuracy by analyzing stage-wise retention of targeted-routing decisions and rejection causes before and after targeted retrieval. These diagnostics show how the controller filters bridge commitments before they can constrain subsequent retrieval.

Shared-prefix replay. We collect the decision states at which pre-target admission and downstream execution guards permit a valid TARGETED continuation, forming $\mathcal{S}_{\text{replay}}$. From each cached prefix, we independently replay BROAD and TARGETED continuations under the same remaining retrieval budget, BM25–E5–RRF configuration, Evidence Selector, and Reasoner. We compare their final answers by EM first and use token-level F1 to break EM ties; a state is counted as a tie only when both EM and F1 are identical. Figure 3(a) reports the proportions of TARGETED-better, tied, and BROAD-better states over $N = |\mathcal{S}_{\text{replay}}|$ replayed states.

B.4.1 Stage-Wise Retention of Targeted Routing

Figure 4 reports the proportion of states retained through the targeted-routing pipeline. Among states considered for bridge-directed retrieval, 57.0% initially select TARGETED. The pre-target Evidence Gate retains 29.8% of the analyzed states, a conditional pass rate of 52.3%. After targeted retrieval, 24.7% remain supported by the returned evidence, corresponding to an 82.9% conditional pass rate among branches admitted by the first-stage gate. The main filtering therefore occurs before targeted retrieval: the gate removes nearly half of initially proposed targeted decisions, while the post-target support check revalidates the remaining commitments against new evidence.

B.4.2 Rejection Causes Before and After Targeted Retrieval

Figure 5(a) analyzes candidate bridge commitments rejected before targeted retrieval. Grounding failures account for 52.5% of pre-target rejections, followed by compatibility failures (29.3%), consistency failures (14.1%), and structural or other failures (4.0%). These categories correspond to the first failed source-grounding, bridge-compatibility, or history-consistency check, with structural issues reserved for routing or execution guards. The first-stage gate therefore primarily acts as an evidence-grounding filter: most rejected bridges cannot be directly anchored in the retrieved evidence, while the remaining checks exclude grounded but unsuitable commitments.

Figure 5(b) analyzes branches rejected after targeted retrieval. Insufficient Evidence is the largest category (42.1%), followed by Commitment Unsupported (31.6%), Commitment Inconsistent (21.1%), and structural or other issues (5.3%). Relevant retrieved material is therefore not sufficient by itself; the second-stage support check requires direct evidence for the committed bridge–gap binding. A failing branch is retracted and recovery returns to broader exploration.

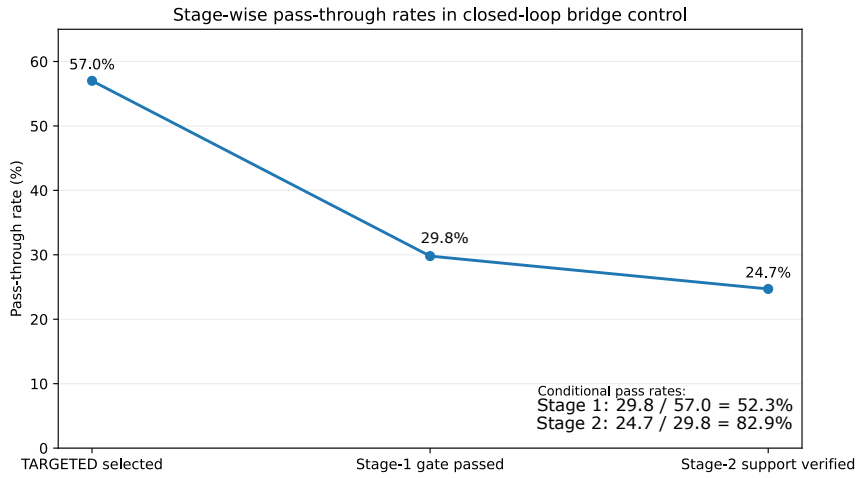


Figure 4: Stage-wise pass-through rates in closed-loop bridge control. Of the states selecting TARGETED retrieval, 29.8% of all analyzed states pass the pre-target Evidence Gate and 24.7% remain supported after targeted retrieval. The corresponding conditional pass rates are 52.3% for the first-stage gate and 82.9% for post-target support verification.

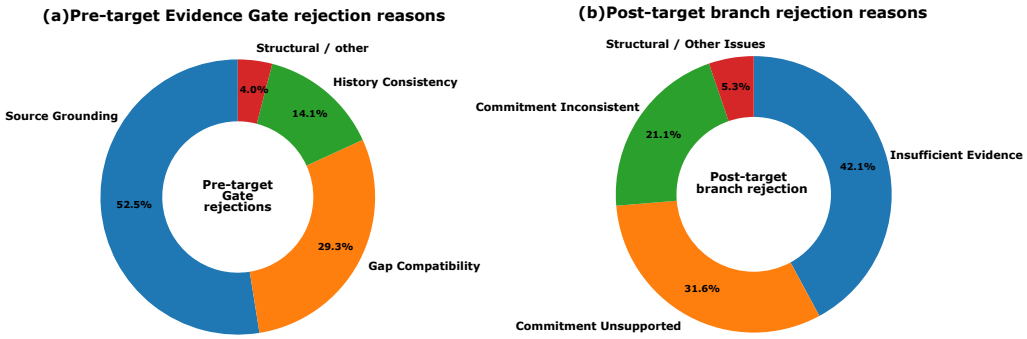


Figure 5: Breakdown of closed-loop rejection reasons. (a) Pre-target Evidence Gate rejections are dominated by grounding failures, with bridge-compatibility and consistency providing additional screening. (b) Post-target rejections distinguish insufficient retrieved evidence from unsupported or inconsistent bridge commitments.

B.4.3 Limitation: Anchor-Judge Proposal Coverage

CLBR-RAG's TARGETED routing depends on candidate bridge–gap bindings proposed by the Anchor Judge. Before retrieval, the Evidence Gate applies deterministic checks for source grounding, compatibility with active information gaps, and history consistency. After retrieval, a separate support check retains a branch only when its returned evidence supports the binding; otherwise, the branch is retracted. These checks can reject or retract proposed bindings, but they cannot generate a valid binding omitted by the Anchor Judge. Thus, an omitted binding cannot guide TARGETED retrieval in the current round; later BROAD exploration or a new proposal may still recover the bridge, while TARGETED path coverage remains bounded by upstream proposal coverage.

Current diagnostics characterize the admission and post-retrieval retention of proposed bindings, but do not directly measure the Anchor Judge's proposal precision and recall over valid bridge–gap bindings. The capacity analysis in Appendix B.3 varies the Base Judge while keeping the Anchor Judge at its default capacity, so the effect of Anchor Judge capacity on candidate coverage and routing remains untested. Future work could use diagnostic sets with gold intermediate evidence or bridge annotations, mapped from existing benchmark annotations where possible, to measure

proposal precision and recall. It could also study distillation from stronger bridge-routing policies or their decision trajectories, and evaluate confidence estimation and calibration for candidate proposals while retaining CLBR-RAG's deterministic admission and failure-recovery mechanisms.

B.5 Qualitative Case Studies of Closed-Loop Bridge Retrieval

Complementing the aggregate controller diagnostics in Appendix B.4, we present two representative execution traces that show how the retrieval scope changes as evidence accumulates. The first trace illustrates a successful transition from coverage-oriented BROAD retrieval to bridge-conditioned TARGETED retrieval. The second illustrates failure-aware recovery: when a targeted branch does not obtain admissible support for its provisional commitment, the controller returns to BROAD exploration. We report only the evidence state, retrieval path, selected evidence, and control outcome relevant to routing.

B.5.1 Successful Bridge-Conditioned Retrieval

Question. What nationality is the director of the film *Abeti Masikini: Le Combat D'Une Femme*?
Dataset. 2Wiki. **Gold answer.** Congolese. **Prediction.** Congolese.

The initial BROAD retrieval identifies Ne Kunda Nlaba as the film's director, but does not establish his nationality. This source-grounded intermediate entity becomes a bridge to the unresolved attribute. The subsequent TARGETED retrieval conditions on this bridge and returns the answer-bearing evidence.

Table 12: A successful bridge-conditioned retrieval trace. BROAD identifies the director, and TARGETED resolves the director's nationality while retaining the evidence accumulated in the first round.

Step	Evidence state	Retrieval	Key selected evidence	Control outcome
R1	Missing attribute: director nationality	BROAD	[*Abeti Masikini: Le Combat d'Une Femme*] "The film was directed by Ne Kunda Nlaba."	Source-grounded bridge emerges; evidence remains insufficient.
R2	Ne Kunda Nlaba $\rightarrow$ nationality	TARGETED	Query: "Ne Kunda Nlaba nationality." [*Ne Kunda Nlaba*] "... is a Congolese Film Producer, Director, ..."	Accepted branch resolves the nationality slot.
Final	Film $\rightarrow$ director $\rightarrow$ nationality	No retrieval	The R1 director and R2 nationality evidence remain in the compact final context.	Next Base Judge: sufficient; *Congolese*.

This trace shows the intended transition from broad candidate coverage to narrow bridge-conditioned retrieval. After identifying the director, CLBR-RAG does not issue another generic query. It uses the resolved intermediate entity to focus retrieval on the remaining nationality slot. The source pointer, active gap, and bridge entity define the pre-target candidate under the fixed gate predicates, while the targeted sentence provides the entity, nationality witness, and provenance required by the post-target support check. The film–director evidence from R1 remains available for final organization.

B.5.2 Recovery from an Unsupported Targeted Branch

Question. Phil Ruffin, an American businessman born in 1935, owned a greyhound racing track called Wichita Greyhound Park. In which state is this track located?
Dataset. HotpotQA. **Gold answer.** Kansas. **Prediction.** Kansas.

This trace illustrates failure-aware recovery. The first targeted branch is plausible under the current evidence state, but its selected evidence concerns a different track and does not support the commit-

ted Wichita Greyhound Park–state binding. The branch is rejected, so it contributes neither evidence nor a binding to the persistent state. The controller then executes BROAD retrieval, which exposes Wichita as a new source-grounded anchor for a subsequent TARGETED branch.

Table 13: Recovery from an unsupported targeted branch. When targeted retrieval fails to support the provisional Wichita Greyhound Park–state binding, the controller restores BROAD retrieval and subsequently forms a new bridge through Wichita.

Step	Evidence state	Retrieval	Key selected evidence	Control outcome
R1	Missing: Wichita Greyhound Park → state	BROAD	[*Wild West World*] links Phil Ruffin to the track; no selected passage establishes its state.	Insufficient; candidate bridge remains provisional.
R2	Provisional: Wichita Greyhound Park	TARGETED	Query: "Wichita Greyhound Park Wild West World location." [*Hinsdale Greyhound Park*] discusses closure and New England spectators, not the committed track or state.	Rejected; no branch evidence or binding is committed.
R3	Original location gap open	BROAD	[*Wichita Transit*] and [*Sedgwick County Zoo*] both locate Wichita in Kansas.	BROAD recovery exposes Wichita as a new source-grounded anchor.
R4	Wichita → location	TARGETED	Query: "Wichita location." [*Lawrence–Dumont Stadium*], [*Wichita West High School*], and [*Wichita Public Library*] independently locate Wichita in Kansas.	Accepted Wichita–Kansas evidence; prediction: *Kansas*.

The critical transition occurs between R2 and R3. Although the targeted branch retrieves documents, its selected evidence neither mentions the committed track nor establishes its location. No targeted branch is accepted in R2, so the controller forces the next retrieval round to BROAD. The failed park-specific branch is absent from the next query and does not update the persistent evidence context or bridge ledger. BROAD recovery exposes alternative evidence centered on Wichita, and a new TARGETED branch retrieves repeated support for Wichita, Kansas. The archived route sequence is

$$\text{BROAD} \rightarrow \text{TARGETED} \rightarrow \text{BROAD} \rightarrow \text{TARGETED}. \tag{17}$$

After R4, the terminal Base Judge marks the accumulated evidence as insufficient. The retrieval budget prevents an additional query, so the Reasoner returns Kansas from the compact terminal evidence view. This is a budget-exhausted, non-certified termination rather than a gate-certified terminal success. Accordingly, the case demonstrates recovery from an unsupported branch and subsequent answer-bearing retrieval; it is not presented as a fully certified evidence chain for the original track–state relation.

Together, the traces illustrate complementary functions of closed-loop bridge retrieval. The first shows when narrowing is useful: a supported intermediate entity focuses retrieval on a missing downstream attribute. The second shows why the commitment remains provisional: when retrieved evidence fails to support the bridge–gap binding, the controller restores BROAD exploration instead of repeatedly narrowing along an unsupported path. These behaviors instantiate the aggregate patterns in Appendix B.4, where the controller screens commitments before TARGETED retrieval and revalidates admitted branches against newly retrieved evidence.

C Prompt Templates for Closed-Loop Retrieval

This appendix specifies the system prompts for the Base Judge, Anchor Judge, and Evidence Selector at inference time. They are presented in runtime order: the Base Judge assesses evidence

sufficiency and forms a semantic retrieval request; the Anchor Judge proposes a routing preference and source-bound candidate bindings; and the Evidence Selector extracts sentence-level evidence from retrieved candidates. The programmatic routing, query-composition, and validation procedures, including the Evidence Gate and Anchor Query Composer, are summarized in Appendix A.

C.1 BASE JUDGE

Table 14: Base Judge system prompt for evidence sufficiency assessment and semantic query formulation.

Base Judge — System Prompt

You are a QA/RAG sufficiency judge and retrieval-query writer.
Given a QUESTION and a CONTEXT (documents retrieved so far), decide whether the CONTEXT alone contains enough information to reliably answer the QUESTION. If not, list the gap items that describe what information is still missing and write one standalone search query optimized for dense retrieval.
Output exactly one JSON object and nothing else.
The JSON object must have exactly three keys: `"sufficient"`, `"gap_items"`, and `"search_query"`.
Schema:

```
{"sufficient": true/false,
  "gap_items": [
    {"category": "bridge entity | attribute | relation |
evidence span | other",
     "target": "string", "slot": "string", "description":
"string"}
  ], "search_query": "string"}
```

Search-query rules when `"sufficient"` is false:
1. Write exactly one concise, standalone query for the most blocking missing fact.
2. Include the concrete named entity or entities and the missing relation or attribute.
3. Resolve pronouns and vague phrases such as the director, that country, or the person.
4. Use natural language suitable for semantic dense retrieval; do not repeat phrases or copy the whole question blindly.
5. Use only entities and facts available in the QUESTION or CONTEXT. Do not guess the answer.
6. For a comparison that requires both sides, include both concrete entities in the query.
Gap-item rules:
1. Return only the 1 to 3 distinct gap items that are most necessary for answering the QUESTION.
2. Do not repeat or paraphrase the same gap more than once.
3. Every gap item must be one complete JSON object inside the `"gap_items"` array.
If `"sufficient"` is true, `"gap_items"` must be `[]` and `"search_query"` must be `""`.
If `"sufficient"` is false, `"gap_items"` must be non-empty and `"search_query"` must be non-empty.
Use only the CONTEXT as answer evidence.

C.2 ANCHOR JUDGE

Table 15: Anchor Judge system prompt for source-bound bridge routing.

Anchor Judge — System Prompt

You are the bridge-routing judge in a multi-hop retrieval system.
Use only QUESTION, BASE_QUERY, G_ALL, EVIDENCE_CONTEXT_SO_FAR, and DELTA_S. Do not answer the question, write retrieval queries, use unstated knowledge, or explain your reasoning.
Propose whether the next retrieval should be TARGETED or BROAD, and for a TARGETED proposal provide candidate bridge anchors. The action field is a routing preference, not final authorization: the programmatic Evidence Gate admits an anchor only when it passes the fixed source-grounding, bridge-compatibility, and history-consistency checks; downstream execution guards determine whether the round ultimately executes TARGETED.
Return TARGETED when DELTA_S contains one or more source-stated bridge entities that can advance an active G_ALL gap. Do not decide whether the proposed anchors jointly cover all unresolved information needs; the programmatic TaskClosure guard evaluates that set after Evidence-Gate admission.
Each anchor links a DELTA_S item that explicitly states a bridge entity to a G_ALL gap that this entity can advance.
Return exactly one JSON object and no prose:
`{"action": "TARGETED|BROAD", "anchors": [{"delta_index": 0, "gap_index": 0, "bridge_entity": "string"}]}`
Constraints:
- For BROAD, `anchors` must be `[]`.
- For TARGETED, each `bridge_entity` must be explicitly stated in its DELTA_S item.
- `delta_index` identifies the source DELTA_S item, and `gap_index` identifies the associated gap in the original zero-based G_ALL order.
- You may propose anchors for one or more active gaps; do not repeat a bridge entity.
- The program applies fixed source-grounding, bridge-compatibility, and history-consistency checks after this proposal; TaskClosure and query-envelope validation remain downstream execution guards.

C.3 EVIDENCE SELECTOR

Table 16: Evidence Selector system prompt for gap-conditioned sentence extraction.

Evidence Selector — System Prompt

You are a sentence-level evidence selector for a multi-hop RAG system.
You will receive:
1. an ORIGINAL QUESTION,
2. MISSING FACTS that describe what information is still missing,
3. a numbered list of SENTENCES from retrieved documents.
Your task is to select the sentence ids that maximize answerability for the ORIGINAL QUESTION.
Selection policy:
1. First prioritize sentences that fill the MISSING FACTS, especially bridge entities, attributes, relations, and evidence spans needed for the next hop.
2. Then prioritize sentences that directly support the final answer to the ORIGINAL QUESTION.
3. Prefer sentences that are self-contained and explicit: they mention the key entity, relation, attribute, date, number, or answer-bearing fact; and they remain understandable when extracted alone.
4. If a selected sentence depends on nearby context to be understandable or useful, include the minimal additional sentence(s) needed to preserve that context.
5. Do not infer, rewrite, paraphrase, or generate evidence text. Only return ids from the provided list.
6. If no sentence is useful, return an empty list.
Output format (strict):
Return exactly one JSON object and nothing else:
`{"evidence_global_ids": [1, 5, 7]}`
Constraints:
- `"evidence_global_ids"` must be a JSON array of integers.
- Select at most K sentences, where K is given in the user message.
- Only use ids that appear in the numbered sentence list.
- Do not repeat ids.