# DEPTHLENS: MEASURING HOW EVIDENCE ASSEMBLY CHANGES USEFUL RETRIEVAL DEPTH

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

A RAG score observed at one retrieval depth does not identify how long additional evidence remains useful: the assembly operator can change the entire depth–utility response. We define *useful-depth shift* as the paired displacement of the fitted low-return transition under an assembly replacement, and instantiate it in DEPTHLENS by crossing seven depths with assembly operators on a fixed retriever–reranker–generator stack and resampling matched queries through the complete estimator. On HotpotQA, 2WikiMultiHopQA, MuSiQue, and FEVER, a Manual Ledger that carries entities, supporting evidence, and unresolved hops moves this transition from 7.4–16.2 passages to 3.4–5.2, a 2.2–3.1$\times$ shift; Qwen2.5-72B-Instruct reproduces 2.1–2.9$\times$. In the operator ladder, layout-matched and random-schema controls retain CoT-like transitions, whereas a frozen automatic scaffold recovers most of the movement; the Manual-over-Auto residual rises across pooled hop strata, and the Manual-over-CoT gap widens with distractor rate. The measured displacement aligns with held-out action quality: with Manual Ledger fixed, DEPTHLENS reaches 88.9–93.1% near-SLO pair accuracy, stays within 0.5–0.9 primary-score points (EM; FEVER label accuracy) of the feasible oracle, and improves on the matched Learned-$k_r$ comparator by 0.9–2.9 points. Joint depth–assembly selection then gains 3.2–12.2 points over matched depth-only adaptation at 1.48–1.68 s P95, showing that retrieval evaluation must cross depth with evidence assembly to expose both scientific attribution and efficient operating points.

## 1 INTRODUCTION

A single observation $U(k_0, \pi)$ cannot identify a RAG system's retrieval regime. Two assembly operators may score similarly at depth $k_0$ even when one extracts the decisive evidence early and the other continues to benefit from longer prefixes; conversely, a point gain can arise from evidence organization without any change to retrieval quality. We therefore ask a curve-level question: *when assembly changes but the retrieved-prefix family does not, how far does the transition to low marginal retrieval return move?* This question separates the value of acquiring another passage from the value of retaining and organizing what has already been acquired.

The measured object is *useful-depth shift*. For an assembly operator $\pi$, $k_r^c(\pi)$ is the fitted transition between the high-return and low-return regions of its depth–utility response; replacing $\pi_a$ with $\pi_b$ induces the paired displacement $k_r^c(\pi_a)/k_r^c(\pi_b)$. Stateful assembly motivates a specific, testable signature: an operator that updates entities, accepted evidence, and unresolved hops should reach its low-return region earlier than free-form CoT, especially as reasoning chains or distractor density grow. Layout-only and semantically unrelated field schemas distinguish this state-retention account from generic formatting, while an automatically generated task schema tests whether the effect depends on a hand-written Ledger.

DEPTHLENS identifies the displacement by replaying identical retrieved prefixes across $\mathcal{K} = \{3, 5, 7, 10, 12, 15, 20\}$ and changing only assembly. Corpus, retriever, reranker, generator, context cap, decoding, and hardware remain fixed within every cross. A constrained continuous two-regime fit estimates $k_r^c$, and 1,000 paired query-bootstrap replicates rerun the complete estimator so that uncertainty includes movement of the fitted curve. The evaluation then asks whether this measurement changes decisions in two distinct tests: a Manual-Ledger-fixed comparison isolates depth-policy



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.

quality, and a 21-action comparison measures the joint depth–assembly frontier under an end-to-end latency SLO.

The primary crossed-depth estimates in Figure 1 and Table 2 reveal a consistent workload gradient. On HotpotQA, 2WikiMultiHopQA, MuSiQue, and FEVER, CoT transitions at 7.4–16.2 passages, whereas Manual Ledger transitions at 3.4–5.2, a 2.2–3.1$\times$ shift; Qwen2.5-72B-Instruct reproduces 2.1–2.9$\times$ with the same ordering. Structured-base and Random schema retain CoT-like knots, while Auto-scaffold recovers most of the shift. In pooled hop strata, the residual Manual-Ledger advantage rises from 1.2 primary-score points on one-hop queries to 4.8 on four-hop queries; at fixed $k_r = 10$, its HotpotQA advantage over CoT widens from 20.2 to 30.2 EM as topical distractors replace retrieved passages. Across the operator ladder, richer carried task state coincides with shallower transitions, while reported token footprints keep compression visible in the comparison.

In held-out decision tests, the measured shift reorders preferred actions under latency. With Manual Ledger fixed, DEPTHLENS improves on the matched Learned-$k_r$ comparator by 0.9–2.9 primary-score points, reaches 88.9–93.1% near-SLO pair accuracy, and remains within 0.5–0.9 points of the feasible oracle; joint selection gains 3.2–12.2 points at 1.48–1.68 seconds P95. Together, the estimand, matched operator ladder, and held-out decision tests support measuring retrieval depth on an assembly-conditioned utility surface rather than at one fixed depth.

## 2 RELATED WORK

Table 1: Method-level boundary to the closest RAG objectives. DEPTHLENS targets the cross-assembly movement of a matched depth–utility curve.

| Methods | Primary object | Distinction from useful-depth shift |
|---|---|---|
| Self-RAG, CRAG, IterRetGen, Stop-RAG, AutoSearch | retrieval trigger, iteration, or stopping policy | optimize acquisition with an assembly representation held as part of the method |
| DeepNote, State-Aware RAG, AdaGATE | evolving notes, working memory, or gap-aware evidence sets | design state for evidence use; their objective is answer quality rather than cross-operator curve displacement |
| SAGE and Learned-$k_r$ | query-specific depth under cost or latency constraints | select depth on one assembly-conditioned utility surface |
| DEPTHLENS | useful-depth shift and its feasible-action consequence | crosses depth $\times$ assembly, bootstraps the fitted displacement, and tests depth choice again with assembly held constant |

Table 1 locates the paper at the interaction between acquisition and evidence representation. The relevant counterfactual holds the query, retrieved-prefix family, retriever, reranker, generator, and budget constant while changing assembly, then measures movement of the complete depth–utility regime. DEPTHLENS pairs this intervention with uncertainty on the fitted transition and a held-out test over SLO-feasible depths. The resulting contribution is an evaluation object and identification protocol: an assembly method can improve answers, yet its effect on useful retrieval depth remains a separate measurable property.

Adaptive retrieval changes when and how evidence is acquired. RAG (Lewis et al., 2020), multi-hop dense retrieval (Xiong et al., 2021), decomposition (Press et al., 2022), Self-RAG (Asai et al., 2024), CRAG (Yan et al., 2024), and iterative retrieve-generate systems (Shao et al., 2023) improve coverage through learned triggers, critiques, or repeated retrieval. More recent controllers learn the value of another iteration (Park et al., 2025) or a minimal sufficient search depth (Sun et al., 2026); SAGE predicts a retrieval budget from lightweight signals under a P95 target (Raza et al., 2026). Their decision variable is acquisition under a chosen state representation. Our crossed-prefix experiment instead treats assembly as the intervention and retrieval depth as the response axis. Learned-$k_r$ and the common-adapter runs then test whether this measured interaction changes acquisition decisions.

Structured reasoning changes how intermediate information is represented. Scratchpads (Nye et al., 2021), chain-of-thought (Wei et al., 2022; Kojima et al., 2022), self-consistency (Wang et al., 2023), ReAct (Yao et al., 2023b), Tree-of-Thought (Yao et al., 2023a), least-to-most prompting (Zhou et al., 2022), program-of-thought (Chen et al., 2022), and structured prompting (Hao et al., 2022)

establish that representation affects reasoning quality. The closest RAG systems make persistent state operational in different ways: DeepNote accumulates notes that guide later retrieval (Wang et al., 2025), State-Aware RAG trains an extractor to update working memory (Man et al., 2026), and AdaGATE maintains entity gaps while assembling token-budgeted evidence (Guo et al., 2026). DEPTHLENS addresses a complementary evaluation objective: measuring how far an assembly replacement moves a matched depth–utility transition. Its common seven-depth operator ladder makes that objective executable for layout, arbitrary-field, minimal-state, automatically generated, and Manual-Ledger assemblies while holding retrieval and training fixed.

Cost-aware inference allocates resources after or alongside retrieval. LongLLMLingua (Jiang et al., 2024), RECOMP (Xu et al., 2024), FrugalGPT (Chen et al., 2023), speculative decoding (Leviathan et al., 2023), energy-based calibration (Chen et al., 2025), test-time sampling (Wu et al., 2025), and trace ranking (Jiang et al., 2025) optimize context, model, or generation cost. HNSW (Malkov & Yashunin, 2020) and FAISS (Johnson et al., 2021) expose retrieval recall–latency trade-offs, while cost-based and adaptive query processing motivate exact choice over a measured action set (Selinger et al., 1979; Deshpande et al., 2007). These policies consume a utility surface. DEPTHLENS first estimates how assembly changes that surface and then enumerates the resulting feasible actions, separating the interaction measurement from the resource decision.

The distinction is empirical rather than terminological. Point accuracy at one depth cannot recover a cross-operator displacement of the utility regime. A matched sweep estimates that displacement, complete-estimator bootstrap quantifies its stability, and Ledger-fixed policies test decision quality independently of assembly quality. This combination makes useful-depth shift a testable systems quantity while guarding against overinterpreting sharp-looking transitions (Schaeffer et al., 2023; Power et al., 2022).

## 3 METHOD

RAG execution is represented by a retrieval depth $k_r$ and an assembly operator $\pi$. Within each crossed comparison, the retriever, reranker, generator, context cap, and decoding are fixed, so the response of utility to $k_r$ can be compared across assembly conditions. DEPTHLENS first estimates how $\pi$ displaces the depth–utility regime and then chooses among the measured actions under a latency constraint. Separate matched comparisons evaluate curve displacement, depth-policy quality with assembly held constant, and their joint consequence.

### 3.1 IDENTIFICATION DESIGN AND USEFUL-DEPTH ESTIMAND

Each workload uses seven observed depths, $\mathcal{K} = \{3, 5, 7, 10, 12, 15, 20\}$. For every query, depth $k_j$ is a prefix of the same ranked passage list used at $k_{j+1}$; every assembly operator therefore receives an identical prefix at an identical depth. The selectable assembly set is $\Pi = \{\text{CoT}, \text{Structured-base}, \text{Manual Ledger}\}$, evaluated on 100 calibration queries. Random schema and Auto-scaffold are attribution operators, and Minimal scaffold is an additional state-content ablation; these controls are crossed with depth to explain transition movement but are not added to the pre-specified 21-action serving set. Let $x = \log k_r$ and $m_\pi(x)$ be mean primary utility—EM for the QA workloads and normalized label accuracy for FEVER—with token F1 retained as a diagnostic metric. We fit the continuous, joined two-regime curve

$$\hat{m}_\pi(x; \tau) = \begin{cases} a + \alpha(x - x_{\min}), & x \le \tau, \\ a + \alpha(\tau - x_{\min}) + \gamma(x - \tau) - \beta(x - \tau)^2, & x > \tau, \end{cases} \tag{1}$$

by least squares over the seven observed means, constrained by $\alpha \ge 0$, $\beta \ge 0$, and $\gamma \le \alpha$. Continuity is built into Eq. 1; the slope constraint permits a plateau or decline after the knot while excluding a spurious upward acceleration. We profile $\tau$ over $[\log 3, \log 20]$, solve the constrained coefficients for each candidate, and choose the smallest minimizer under objective ties. The fitted useful-depth transition is

$$k_r^c(\pi) = \exp(\hat{\tau}_\pi). \tag{2}$$

For an assembly replacement $\pi_a \to \pi_b$, we quantify the displacement as

$$\Delta_{\text{UD}}(\pi_a, \pi_b) = \log \frac{k_r^c(\pi_a)}{k_r^c(\pi_b)}, \qquad \rho_{\text{UD}} = \exp(\Delta_{\text{UD}}) = \frac{k_r^c(\pi_a)}{k_r^c(\pi_b)}. \tag{3}$$

A positive $\Delta_{\mathrm{UD}}$ means that $\pi_b$ reaches its low-return regime at a shallower depth; tables report the multiplicative factor $\rho_{\mathrm{UD}}$. This cross-operator contrast is the assembly-conditioned retrieval-demand effect. The seven depths are observations, whereas fractional values such as $4.1$ and $11.2$ are continuous fitted knots. For interpretability, $\chi_i = \Delta m / \Delta \log k_r$ reports finite-difference marginal gain between adjacent observations; $\chi$ visualizes the curve but does not define $k_r^c$.

Uncertainty resamples matched queries 1,000 times. Each paired-bootstrap replicate recomputes both operators' seven utility means and refits Eq. 1; percentile endpoints give the marginal intervals for both fitted knots. Tables report these intervals beside the point ratio in Eq. 3, keeping the uncertainty attached to the two quantities that determine the displacement. A transition is labeled smooth when the segmented fit does not improve leave-one-depth-out squared error over a one-regime fit, when the optimum repeatedly hits an interval boundary, or when its bootstrap interval spans more than two adjacent depth intervals. Smooth cases retain their empirical per-plan utilities for selection and omit a transition-ratio claim. Every primary and second-backbone CoT–Ledger pair passes this pre-specified reporting rule; Appendix Table 12 exposes the fit status beside each primary estimate.

### 3.2 SLO-CONSTRAINED EXACT SELECTION

At inference time, DEPTHLENS enumerates the calibrated plan set and chooses

$$(k_r^\star, \pi^\star) = \arg\max_{k_r, \pi} \hat{m}(q, k_r, \pi) \quad \text{s.t.} \quad \hat{L}(q, k_r, \pi) \le L_{\max}, \tag{4}$$

where $\hat{m}(q, k_r, \pi) = \hat{m}_{b(q)}(k_r, \pi)$ and $\hat{L}(q, k_r, \pi) = \hat{L}_{b(q)}(k_r, \pi)$ are estimates conditioned on the query bucket $b(q) = (d(q), c_s(q))$. The workload family is denoted by $d(q)$; $c_s(q)$ assigns an empirical-rank bucket to the best first-hop reranker score $s_1(q)$:

$$c_s(q) = 1 + \mathbb{I}\big[F_{\mathrm{cal},d(q)}(s_1(q)) \ge \tfrac{1}{3}\big] + \mathbb{I}\big[F_{\mathrm{cal},d(q)}(s_1(q)) \ge \tfrac{2}{3}\big], \tag{5}$$

where $F_{\mathrm{cal},d}$ is the empirical CDF on the calibration slice. The requested SLO enters Eq. 4 directly through $L_{\max}$; the primary experiments use 2.5 seconds. Equation 1 is refit within each bucket and assembly. Buckets containing fewer than 12 calibration queries back off to the dataset-level assembly fit; an empty dataset/operator cell backs off to the pooled operator fit. Thus every plan has a defined estimate without imputing from evaluation data.

The latency model is factorized:

$$\hat{L}_{d,c_s,\pi}(k_r) = \hat{L}^{\mathrm{retr}}_{d,c_s}(k_r) + \hat{L}^{\mathrm{rerank}}_{d,c_s}(k_r) + \hat{L}^{\mathrm{assembly}}_{d,c_s,\pi}(k_r) + \hat{L}^{\mathrm{gen}}_{d,c_s,\pi}(k_r) + \hat{L}^{\mathrm{queue}}_{d,c_s,\pi}(k_r), \tag{6}$$

where each component is profiled during the same sweep and queueing is measured at the stated concurrency. The action set contains 21 candidates (seven depths times three assemblies), so deterministic enumeration exposes every feasibility and utility prediction. Keeping Random schema, Minimal scaffold, and Auto-scaffold in the attribution cross but outside this action set separates the mechanism experiment from the serving policy: the former asks which state properties move a curve, whereas the latter compares three pre-specified online operators. Objective ties within numerical tolerance choose lower predicted latency, then lower depth, then the fixed operator order Manual Ledger, Structured-base, CoT. If no action satisfies $L_{\max}$, the minimum-latency action is returned and tagged as a fallback. Cache entries are keyed by bucket, SLO, and calibration version. Drift detection monitors a 32-query moving mean of absolute residuals; latency residuals arrive immediately, while quality residuals enter when delayed evaluation labels are observed in the replay. Crossing the calibration 95th percentile on either available channel opens the fixed 200-query recalibration window and invalidates affected cache entries.

Learned-$k_r$ is the depth-only comparator under the same information and feasibility boundary. It fits one utility prediction per depth from workload identity, first-hop score bucket, the seven-depth calibration outcomes, and factorized latency estimates, without a transition or cross-assembly feature. Calibration allocation, score buckets, candidate depths, latency measurements, feasibility test, tie breaking, and fallback are shared with DEPTHLENS; neither policy sees evaluation labels. In the joint comparison Learned-$k_r$ receives CoT, so the comparison measures the full application of the interaction. In the Ledger-fixed comparison both policies receive Manual Ledger, making depth ranking the only changing component and the resulting 0.9–2.9 primary-score-point difference the clean selector comparison.

**Algorithm 1** DEPTHLENS calibration and serving procedure

```
Construct disjoint 100-query calibration and 1,000-query evaluation slices; freeze seeds and normalization.
for each d, π, k_r do
    Execute the fixed stack; record utility and all five latency components in Eq. 6.
end for
Form score-CDF buckets; fit Eq. 1 with backoff; bootstrap k_r^c.
Profile factorized latency and materialize all 21 actions per bucket.
for each request q do
    Enumerate actions satisfying L_max; apply the deterministic objective and tie rules.
    Serve or reuse the versioned cached plan; update residual-based drift statistics.
    if drift is flagged then
        Invalidate the affected cache and refit after the 200-query window.
    end if
end for
```

The joint frontier permits all three operators and reports their selected frequencies. The Ledger-fixed isolation restricts $\Pi = \{\text{Manual Ledger}\}$, leaving only depth choice. Its post-hoc finite-action oracle takes the best empirically feasible depth on each held-out query and is used only as an upper bound. For query $q$, normalized regret is $(u_q^{\text{oracle}} - u_q^{\text{selected}})/(u_q^{\text{oracle}} - u_q^{\min})$, set to zero when the denominator is zero; boundary pair accuracy and top-1 mis-selection then expose local ordering errors directly. Appendix Table 14 retains global fit diagnostics, while Tables 6 and 7 evaluate the decisions that determine utility at the SLO.

### 3.3 MANUAL LEDGER AS THE STATEFUL OPERATOR

Ledger assembly supplies four typed fields: entities, supporting evidence, unresolved hops, and a provisional answer. Within each plan, the generator reads passages in rank order and is instructed after each admitted passage to merge entity and evidence entries, remove only supported hops from the unresolved set, and revise the provisional answer only from retained evidence. The final serialized state conditions the answer. CoT instead carries free-form reasoning; Structured-base supplies sectioned layout without the Ledger field contract; Random schema retains a fielded form with unrelated labels. All operators share passage order, decoding, and the 4,096-token consumed-context cap. Appendix Table 9 states the read–update–write contract for every operator, and Table 18 reports pre-truncation footprints so that state semantics and compression remain separate parts of the comparison.

Each $(q, k_r, \pi)$ plan starts from an empty state and is executed independently. "Manual" denotes the hand-specified field contract, not per-query human intervention: one generator invocation assembles and answers from the admitted prefix. Field order and update instructions are fixed; field contents are model-generated. CoT, Structured-base, and Manual Ledger receive the same single-invocation and 1,024-output-token budgets at every depth.

The frozen Auto-scaffold is proposed once per task family by GPT-4o from a task description and five example queries, then evaluated without further edits. It tests whether a generated task-level state schema recovers the same regime; its hop-stratified comparison with Manual Ledger appears in Appendix Table 19. Manual Ledger is one assembly action, whereas DEPTHLENS is the instrument that estimates $\Delta_{\text{UD}}$, constructs the feasible set, and selects a depth/operator action. The second backbone tests model variation, while the secondary SEC 10-K operator-ordering check and serving replays provide descriptive evidence under filing-corpus and workload changes.

## 4 EXPERIMENTS

The central prediction is a workload gradient: an assembly operator that carries unresolved evidence state should move the low-return transition farther on longer reasoning chains, and that movement should change which depths remain valuable under an SLO. We test the prediction first in the crossed curves, immediately replicate it across backbone and domain, and then examine its fixed-assembly

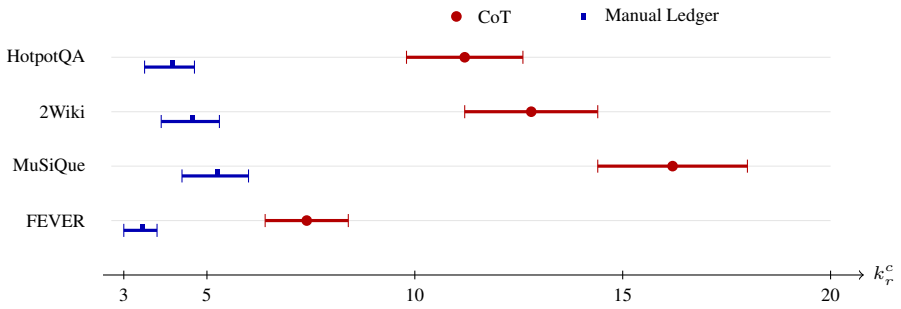



Figure 1: Assembly-conditioned transition estimates on the primary backbone. Marks show fitted $k_r^c$; bars show the marginal paired-bootstrap 95% intervals reported in Table 2. Manual Ledger reaches the low-return regime earlier on all four workloads.

Table 2: Mechanism-to-decision synthesis on the primary backbone. Knot uncertainty is the paired-bootstrap 95% half-width. Ledger-fixed $\Delta U$ compares DEPTHLENS with Learned-$k_r$ under Manual Ledger; joint $\Delta U$/P95 compares joint selection with depth-only adaptation.

| Dataset | CoT $k_r^c$ | Ledger $k_r^c$ | Shift | Ledger-fixed $\Delta U$ | Joint $\Delta U$/P95 |
|---|---|---|---|---|---|
| HotpotQA | $11.2 \pm 1.4$ | $4.1 \pm 0.6$ | 2.7× | +2.6 | +11.6 / 1.68 s |
| 2Wiki | $12.8 \pm 1.6$ | $4.6 \pm 0.7$ | 2.8× | +2.8 | +11.8 / 1.62 s |
| MuSiQue | $16.2 \pm 1.8$ | $5.2 \pm 0.8$ | 3.1× | +2.9 | +12.2 / 1.58 s |
| FEVER | $7.4 \pm 1.0$ | $3.4 \pm 0.4$ | 2.2× | +0.9 | +3.2 / 1.48 s |

and joint decision consequences. Controlled distractors, common-adapter baselines, and serving replays probe the same mechanism under noise and distribution change.

### 4.1 EXPERIMENTAL SETUP

All primary runs use 8×NVIDIA A100-80GB GPUs (CUDA 12.1, driver 535.104.05) with vLLM 0.4.2 (Kwon et al., 2023), PyTorch 2.1.2, and Transformers 4.36.2. Llama-3.1-70B-Instruct (Grattafiori et al., 2024) and Qwen2.5-72B-Instruct (Qwen et al., 2024) use bfloat16 tensor parallelism across eight GPUs and deterministic decoding (temperature 0, top-$p = 1$, 1,024 output tokens). Retrieval uses a 2023-11-01 Wikipedia snapshot containing 21.8M passages, 100-word chunks with 20-word overlap, Contriever-MSMARCO embeddings (Izacard et al., 2022), FAISS HNSW (Johnson et al., 2021; Malkov & Yashunin, 2020) with $M = 32$ and $efSearch = 128$, and BGE-Reranker-v2-M3 (Chen et al., 2024). The context cap is 4,096 consumed tokens.

HotpotQA (Yang et al., 2018), 2WikiMultiHopQA (Ho et al., 2020), MuSiQue (Trivedi et al., 2022), and FEVER (Thorne et al., 2018) each contribute 100 calibration and 1,000 disjoint evaluation queries. Calibration-split seeds are 11, 42, and 2027; thresholds, prompts, normalization, and the seven-depth grid are frozen before evaluation. The primary SLO is 2.5 seconds, with 1.5, 2.0, and 3.0 second sensitivities in Appendix A.4. End-to-end P95 includes retrieval, reranking, prompt assembly, generation, and queueing at concurrency 16. Confidence intervals use 1,000 paired query-bootstrap replicates; calibration-seed variation is reported separately. The same retriever, reranker, corpus, context cap, decoding, and hardware apply to every matched comparison.

Throughout the shared tables, "primary utility" means EM for the QA workloads and normalized label accuracy for FEVER; F1 is retained as a secondary token diagnostic.

### 4.2 STATE-CARRYING ASSEMBLY SHIFTS USEFUL DEPTH

Figure 1 and Table 2 put the fitted displacement and its decision consequence in one view. Manual Ledger moves the transition to 3.4–5.2 passages on every workload; MuSiQue shows the largest ratio and decision gain, while FEVER shows the smallest. The marginal knot intervals are separated, all four fits pass the pre-specified two-regime reporting rule, and Appendix Table 12 reports the same

Table 3: Matched assembly intervention on HotpotQA and SEC 10-K. EM uses DEPTHLENS depth selection with each assembly fixed; $k_r^c$ is fit from the complete seven-depth curve.

| | HotpotQA | | SEC 10-K | |
|---|---|---|---|---|
| Assembly | EM | $k_r^c$ | EM | $k_r^c$ |
| CoT | 64.8 | 11.2 | 56.2 | 14.8 |
| Structured-base | 67.4 | 10.4 | 58.8 | 13.6 |
| Random schema | 67.8 | 10.2 | 59.2 | 13.4 |
| Auto-scaffold | 77.1 | 4.8 | 69.8 | 5.6 |
| Manual Ledger (fixed) | 78.1 | 4.1 | 71.2 | 5.1 |

Table 4: Distractor sensitivity on a held-out 1,000-query HotpotQA controlled-stress slice at fixed $k_r = 10$ with Llama-3.1-70B-Instruct. $p$ is the fraction of retrieved passages replaced by topical non-evidence.

| | Distractor rate $p$ | | | |
|---|---|---|---|---|
| Assembly | 0.0 | 0.25 | 0.5 | 0.75 |
| CoT | 58.2 | 52.4 | 45.6 | 38.2 |
| Manual Ledger | 78.4 | 76.2 | 73.6 | 68.4 |
| $\Delta$EM | +20.2 | +23.8 | +28.0 | +30.2 |

estimates with fit status and hop strata. The decision columns are evaluated on disjoint 1,000-query slices rather than inferred from the 100-query calibration fits.

Layout and arbitrary fields move HotpotQA by at most 3.0 EM and leave $k_r^c$ near 10–11 passages. State-carrying scaffolds produce the qualitative break: Minimal scaffold reaches 72.6 EM at $k_r^c = 6.8$, Auto-scaffold reaches 77.1 at 4.8, and fixed Manual Ledger reaches 78.1 at 4.1 (Appendix Table 18). The monotone ordering tracks progressively richer task state while Table 18 exposes the accompanying footprint reduction. The secondary SEC 10-K check shows the same descriptive operator ordering under a filing corpus and answer style. In pooled hop strata, Manual Ledger's residual advantage over Auto-scaffold grows from +1.2 to +4.8 primary-score points (Appendix Table 19); Section 4.3 next tests the complementary prediction that carried state should become more valuable as retrieved prefixes become noisier.

### 4.3 DISTRACTOR STRESS TRACKS STATE PRESERVATION

This intervention varies distractor rate $p$ on a held-out HotpotQA slice while preserving retrieval depth, passage order, context cap, scoring, and decoding. Topical distractors share query terms but contain no supporting evidence. Unlike the fitted transition, this test fixes $k_r = 10$: it asks whether an assembly operator preserves already acquired support as a constant-length prefix becomes noisier.

CoT loses 20.0 EM across the sweep, while Manual Ledger loses 10.0. The widening advantage from 20.2 to 30.2 points is the predicted signature of state-carrying assembly under noisy retrieval: Ledger preserves more utility as the context contains less supporting evidence, consistent with controlled studies of irrelevant-context sensitivity (Yang et al., 2025; Zhang et al., 2025). This fixed-depth intervention is independent of the transition fit, and its monotone gap complements the hop-stratified result by associating retained evidence state with robustness to noisy prefixes.

Table 5: Matched Qwen2.5-72B-Instruct replication. Decision $\Delta U$ compares joint DEPTHLENS with the stronger Learned-$k_r$+Manual Ledger comparator over 1,000 queries; full utility/F1/P95 and W/T/L appear in Appendix Table 21.

| Dataset | CoT $k_r^c$ | Ledger $k_r^c$ | Shift | Decision $\Delta U$ | Joint P95 (s) |
|---|---|---|---|---|---|
| HotpotQA | 10.8 | 4.3 | 2.5× | +2.7 | 1.73 |
| 2Wiki | 12.4 | 4.8 | 2.6× | +3.0 | 1.67 |
| MuSiQue | 15.6 | 5.4 | 2.9× | +3.1 | 1.63 |
| FEVER | 7.1 | 3.4 | 2.1× | +0.8 | 1.51 |

Qwen preserves both the 2.1–2.9× displacement and its workload gradient: MuSiQue again moves farthest and FEVER least. The replicated curves also lead to positive held-out decision differences at 1.51–1.73 seconds P95. Appendix Tables 20 and 21 give the complete assembly cross, transition intervals, decision metrics, and paired outcomes.

### 4.4 MEASURED SHIFT PREDICTS USEFUL DEPTH CHOICES

Holding Manual Ledger fixed, DEPTHLENS improves over Learned-$k_r$+Ledger by 2.6 primary-score points (95% CI [1.0,4.2]) on HotpotQA, 2.8 ([1.1,4.5]) on 2Wiki, 2.9 ([1.2,4.6]) on MuSiQue, and 0.9 ([0.1,1.7]) on FEVER. Fixed-5+Ledger exceeds the matched Learned-$k_r$+Ledger comparator on all four workloads, showing that independent per-depth estimates mis-rank enough queries on the shallow Ledger plateau; the transition-constrained fit improves while adapting across queries. It remains 0.5–0.9 points from the finite-action oracle and reaches 88.9–93.1% near-SLO pair accuracy. Across calibration seeds, primary-utility standard deviation is at most 0.46 points and pair-accuracy standard deviation is at most 1.3 points. This assembly-fixed test converts the measured shallow regime into an independent depth-decision result.

Table 6: Manual-Ledger-fixed depth policies under the 2.5-second SLO. Each cell is primary utility/F1/P95; depth policy is the only changing component. The post-hoc oracle is the held-out upper bound over the same seven feasible actions.

| Policy | HotpotQA | 2Wiki | MuSiQue | FEVER |
|---|---|---|---|---|
| Oracle feasible | 79.0/83.3/2.36 | 74.7/79.8/2.32 | 67.4/73.0/2.29 | 89.9/91.7/2.25 |
| DEPTHLENS +Ledger | 78.1/82.5/1.69 | 73.9/78.8/1.64 | 66.5/72.0/1.60 | 89.4/91.2/1.49 |
| Learned-$k_r$+Ledger | 75.5/80.0/1.74 | 71.1/76.2/1.70 | 63.6/68.9/1.66 | 88.5/90.5/1.52 |
| Fixed-5+Ledger | 77.3/81.8/1.49 | 72.9/77.9/1.46 | 64.8/70.2/1.43 | 89.1/90.9/1.40 |
| Fixed-10+Ledger | 77.9/82.1/1.88 | 73.2/78.1/1.84 | 63.5/68.7/1.79 | 88.8/90.8/1.68 |

Table 7: Direct action quality for DEPTHLENS +Manual Ledger. Regret is mean feasible-set normalized utility loss. Pair accuracy covers candidates within 0.15 seconds of the SLO; mis-selection uses the corresponding boundary-query set.

| Dataset | Regret (%) | Pair accuracy | Mis-selection | Boundary queries/pairs |
|---|---|---|---|---|
| HotpotQA | 4.8 | 91.7% (1198/1307) | 57/624 | 624/1307 |
| 2Wiki | 5.2 | 90.4% (1113/1231) | 63/598 | 598/1231 |
| MuSiQue | 5.9 | 88.9% (1196/1345) | 72/612 | 612/1345 |
| FEVER | 3.6 | 93.1% (941/1011) | 39/565 | 565/1011 |

### 4.5 ASSEMBLY CHOICE DRIVES THE SHARED-STACK FRONTIER

The joint selector has the highest primary utility among the listed shared-stack methods on every dataset, with P95 between 1.48 and 1.68 seconds. Gains over depth-only adaptation are +11.6 points on HotpotQA (95% CI [8.1, 15.1]), +11.8 on 2Wiki ([8.2, 15.4]), +12.2 on MuSiQue ([8.4, 15.9]), and +3.2 on FEVER ([0.6, 5.8]); all paired intervals exclude zero. Appendix Table 17 gives the corresponding W/T/L counts. Learned-$k_r$ and DEPTHLENS tie at 1.48 seconds on FEVER at displayed precision. Under the same selector logic, replacing joint assembly choice with fixed CoT lowers HotpotQA EM from 78.4 to 64.8 (Appendix Table 22), a 13.6-point policy-level assembly gap. Across all 4,000 queries, the joint selector chooses Manual Ledger on 82.6%, with per-dataset counts in Appendix Table 16. The joint gains are therefore the application-level consequence of both factors; the preceding Ledger-fixed result isolates the selector contribution.

Table 8: Shared-stack consequence on 1,000 queries per dataset (primary utility %, P95 seconds). Utility is EM except FEVER normalized label accuracy. Self-RAG, CRAG, and Iter-RetGen are reimplemented on the shared stack; Appendix Tables 22–23 separately report their official code paths behind the common adapter.

| | HotpotQA | | 2Wiki | | MuSiQue | | FEVER | |
|---|---|---|---|---|---|---|---|---|
| Method | Utility | P95 | Utility | P95 | Utility | P95 | Utility | P95 |
| Fixed-10 | 58.2 | 2.72 | 52.6 | 2.68 | 47.2 | 2.62 | 82.4 | 2.48 |
| Self-RAG (reimpl.) | 64.8 | 2.24 | 59.2 | 2.18 | 52.4 | 2.12 | 85.4 | 2.05 |
| CRAG (reimpl.) | 66.2 | 2.42 | 61.4 | 2.35 | 54.2 | 2.28 | 86.2 | 2.22 |
| Iter-RetGen (reimpl.) | 67.8 | 4.52 | 62.4 | 4.48 | 55.2 | 4.35 | 86.8 | 4.18 |
| Learned-$k_r$ | 66.8 | 1.62 | 62.4 | 1.58 | 54.6 | 1.55 | 86.4 | 1.48 |
| DEPTHLENS (joint) | 78.4 | 1.68 | 74.2 | 1.62 | 66.8 | 1.58 | 89.6 | 1.48 |

## 5 DISCUSSION

Useful-depth shift changes the unit of RAG comparison. Crossing retrieval depth with evidence representation reveals whether an operator changes the complete utility regime; the assembly-fixed test then determines whether that shift supports better depth choices. The smaller FEVER shift and larger MuSiQue shift are consistent with greater opportunity to avoid repeated state reconstruction on longer chains. For system design, the resulting assembly-conditioned surface should be measured before resources are allocated: DEPTHLENS fits the displacement, identifies SLO-feasible actions, and exposes the final trade-off by exact enumeration. The central consequence is to treat state representation and retrieval budget as one coupled operating surface.

The evidence establishes this interaction for one Contriever/BGE stack, two independently trained 70B-class generators, English factual workloads, and 100-query calibration slices. Within that regime, layout and arbitrary-field controls remain CoT-like, minimal and automatically generated state recover progressively more movement, and the specifically manual advantage concentrates on higher-hop queries. The shorter Ledger footprint can contribute; together, the operator ladder, hop gradient, and fixed-depth distractor intervention provide convergent evidence for carried task state while keeping compression inside the measured operator effect. SLO and replay conditions test serving changes, while SEC 10-K supplies a secondary descriptive check under a filing corpus. Residual errors concentrate in reranker mis-ranking and ambiguity (58 of 80 audited cases), identifying retrieval ranking as the next systems bottleneck.

## 6 CONCLUSION

DEPTHLENS measures assembly-conditioned retrieval demand. Across four workloads, Manual Ledger yields a 2.2–3.1× shallower fitted low-return transition than CoT; controls and a second generator preserve the state-carrying interpretation and workload ordering. With assembly fixed, DEPTHLENS is within 0.5–0.9 primary-score points of the feasible oracle, while joint selection gains 3.2–12.2 points at 1.48–1.68 seconds P95.

## REFERENCES


Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. ICLR, 2024. URL https://arxiv.org/abs/2310.11511.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. arXiv preprint, 2024. URL https://arxiv.org/abs/2402.03216.

Lingjiao Chen, Matei Zaharia, and James Zou. Frugalgpt: How to use large language models while reducing cost and improving performance. arXiv preprint, 2023. URL https://arxiv.org/abs/2305.05176.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. arXiv preprint, 2022. URL https://arxiv.org/abs/2211.12588.

Zhikang Chen, Sen Cui, Deheng Ye, Yu Zhang, Yatao Bian, and Tingting Zhu. Think consistently, reason efficiently: Energy-based calibration for implicit chain-of-thought. arXiv preprint, 2025. URL https://arxiv.org/abs/2511.07124.

Amol Deshpande, Zachary G. Ives, and Vijayshankar Raman. Adaptive query processing. *Foundations and Trends® in Databases*, 1(1):1–140, 2007. doi: 10.1561/1900000001.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, et al. The llama 3 herd of models. arXiv preprint, 2024. URL https://arxiv.org/abs/2407.21783.

Yilin Guo, Yinshan Wang, and Yixuan Wang. Adagate: Adaptive gap-aware token-efficient evidence assembly for multi-hop retrieval-augmented generation. arXiv preprint, 2026. URL https://arxiv.org/abs/2605.05245.

Yaru Hao, Yutao Sun, Li Dong, Zhixiong Han, Yuxian Gu, and Furu Wei. Structured prompting: Scaling in-context learning to 1,000 examples. arXiv preprint, 2022. URL https://arxiv.org/abs/2212.06713.

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, Barcelona, Spain (Online), 2020. International Committee on Computational Linguistics. doi: 10.18653/v1/2020.coling-main.580. URL https://aclanthology.org/2020.coling-main.580/.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*, 2022. URL https://arxiv.org/abs/2112.09118.

Eric Hanchen Jiang, Haozheng Luo, Shengyuan Pang, Xiaomin Li, Zhenting Qi, Hengli Li, Cheng-Fu Yang, Zongyu Lin, Xinfeng Li, Hao Xu, Kai-Wei Chang, and Ying Nian Wu. Learning to rank chain-of-thought: Using a small model. arXiv preprint, 2025. URL https://arxiv.org/abs/2505.14999.

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, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/V1/2024.ACL-LONG.91. URL https://arxiv.org/abs/2310.06839.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547, 2021. doi: 10.1109/TBDATA.2019.2921572. URL https://ieeexplore.ieee.org/document/8733051.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. arXiv preprint, 2022. URL https://arxiv.org/abs/2205.11916.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*. ACM, 2023. doi: 10.1145/3600006.3613165. URL https://dl.acm.org/doi/10.1145/3600006.3613165.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19274–19286, Honolulu, Hawaii, USA, 2023. PMLR. URL https://proceedings.mlr.press/v202/leviathan23a.html.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020. URL https://arxiv.org/abs/2005.11401.

Yury A. Malkov and Dmitry A. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(4):824–836, 2020. doi: 10.1109/TPAMI.2018.2889473.

Hieu Man, Ro-ee Tal, Abhishek Kumar, Jaejin Cho, and Benjamin Hsu. Reasoning with memory: Adaptive information management for retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 36825–36841, San Diego, California, United States, 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.findings-acl.1834. URL https://aclanthology.org/2026.findings-acl.1834/.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. Show your work: Scratchpads for intermediate computation with language models. arXiv preprint, 2021. URL https://arxiv.org/abs/2112.00114.

Jaewan Park, Solbee Cho, and Jay-Yoon Lee. Stop-rag: Value-based retrieval control for iterative rag. arXiv preprint, 2025. URL https://arxiv.org/abs/2510.14337.

Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets. arXiv preprint, 2022. URL https://arxiv.org/abs/2201.02177.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. arXiv preprint, 2022. URL https://arxiv.org/abs/2210.03350.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. URL https://arxiv.org/abs/2412.15115.

Muhammad Faizan Raza, Shuo (Luna) Yang, and Satish Mahadevan Srinivasan. Sage: Slo-aware adaptive retrieval for production rag systems. In *2026 12th International Conference on Control, Decision and Information Technologies (CoDIT)*, pp. 169–175, 2026. doi: 10.1109/CoDIT70676.2026.11631166. URL https://doi.org/10.1109/CoDIT70676.2026.11631166.

Rylan Schaeffer, Brando Miranda, and Sanmi Koyejo. Are emergent abilities of large language models a mirage? In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 55565–55581, 2023. URL https://arxiv.org/abs/2304.15004.

Patricia G. Selinger, Morton M. Astrahan, Donald D. Chamberlin, Raymond A. Lorie, and Thomas G. Price. Access path selection in a relational database management system. In *Proceedings of the 1979 ACM SIGMOD International Conference on Management of Data*, pp. 23–34, Boston, Massachusetts, USA, 1979. ACM. doi: 10.1145/582095.582099.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy, 2023. URL https://arxiv.org/abs/2305.15294.

Jingbo Sun, Wenyue Chong, Songjun Tu, Qichao Zhang, Yaocheng Zhang, Jiajun Chai, Xiaohan Wang, Wei Lin, Guojun Yin, and Dongbin Zhao. Autosearch: Adaptive search depth for efficient agentic rag via reinforcement learning. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 28059–28079, San Diego, California, United States, 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.findings-acl.1399. URL https://aclanthology.org/2026.findings-acl.1399/.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. FEVER: A large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 809–819, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/V1/N18-1074. URL https://aclanthology.org/N18-1074.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Musique: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022. doi: 10.1162/tacl_a_00475. URL https://arxiv.org/abs/2108.00573.

Ruobing Wang, Qingfei Zhao, Yukun Yan, Daren Zha, Yuxuan Chen, Shi Yu, Zhenghao Liu, Yixuan Wang, Shuo Wang, Xu Han, Zhiyuan Liu, and Maosong Sun. Deepnote: Note-centric deep retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 19688–19715, Suzhou, China, 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.findings-emnlp.1073. URL https://aclanthology.org/2025.findings-emnlp.1073/.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://arxiv.org/abs/2203.11171.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. URL https://arxiv.org/abs/2201.11903.

Yuheng Wu, Azalia Mirhoseini, and Thierry Tambe. On the role of temperature sampling in test-time scaling. arXiv preprint, 2025. URL https://arxiv.org/abs/2510.02611.

Wenhan Xiong, Xiang Lorraine Li, Srinivasan Iyer, Jingfei Du, Patrick Lewis, William Yang Wang, Yashar Mehdad, Wen-tau Yih, Sebastian Riedel, Douwe Kiela, and Barlas Oğuz. Answering complex open-domain questions with multi-hop dense retrieval. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2009.12756.

Fangyuan Xu, Weijia Shi, and Eunsol Choi. Recomp: Improving retrieval-augmented lms with compression and selective augmentation. ICLR, 2024. URL https://arxiv.org/abs/2310.04408.

Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. Corrective retrieval augmented generation. arXiv preprint, 2024. URL https://arxiv.org/abs/2401.15884.

Minglai Yang, Ethan Huang, Liang Zhang, Mihai Surdeanu, William Wang, and Liangming Pan. How is llm reasoning distracted by irrelevant context? an analysis using a controlled benchmark. arXiv preprint, 2025. URL https://arxiv.org/abs/2505.18761.

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, Brussels, Belgium, 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259/.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, volume 36, 2023a. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/271db9922b8d1f4dd7aaef84ed5ac703-Abstract-Conference.html.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023b. URL https://openreview.net/forum?id=WE_vluYUL-X.

Zhehao Zhang, Weijie Xu, Shixian Cui, and Chandan K. Reddy. Distractor injection attacks on large reasoning models: Characterization and defense. arXiv preprint, 2025. URL https://arxiv.org/abs/2510.16259.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and Ed Chi. Least-to-most prompting enables complex reasoning in large language models. arXiv preprint, 2022. URL https://arxiv.org/abs/2205.10625.

## AI USE STATEMENT

GPT-4o proposed one Auto-scaffold per task family from a task description and five example queries; each resulting schema was frozen before evaluation and received no per-query human editing. Generative language tools also assisted copyediting and presentation.

## A EVALUATION PROTOCOL AND EXTENDED RESULTS

### A.1 EVALUATION PROTOCOL

Every condition follows the same sequence: select a disjoint calibration slice, execute seven depths with frozen assembly prompts, fit transition and latency models, freeze thresholds, and evaluate 1,000 held-out queries. Dataset versions, model checkpoints, answer normalization, and latency boundaries are stated in Sections 3 and 4.1. HotpotQA, 2WikiMultiHopQA, MuSiQue, FEVER, Llama-3.1-70B-Instruct, Qwen2.5-72B-Instruct, Contriever, BGE-Reranker-v2-M3, FAISS, and vLLM use their cited releases. The Wikipedia 2023-11-01 snapshot contains 21.8M passages and is processed into 100-word chunks with 20-word overlap.

The SEC 10-K condition is a secondary descriptive operator-ordering and drift check over filing-grounded factual questions retrieved from EDGAR 10-K documents. It uses the same disjoint 100-query calibration and 1,000-query evaluation sequence, depth grid, context cap, paired scoring, and frozen operator prompts as the public QA conditions. SEC results are reported separately and do not enter the four-workload shift range, backbone replication, or primary mechanism claim.

Each 100-query, seven-depth, three-operator calibration sweep requires 2,100 generator requests per dataset and backbone. Each five-policy Ledger-fixed evaluation requires 5,000 requests per dataset; adaptive methods may issue multiple retrieval rounds inside one request. Offline latency runs use concurrency 16, while replay runs sweep to 64 clients. All runs use eight A100-80GB GPUs, bfloat16 tensor parallelism, CUDA 12.1, vLLM 0.4.2, PyTorch 2.1.2, and Transformers 4.36.2. Exact match and token F1 follow each dataset's official normalization, with FEVER scored as normalized label accuracy.

### A.2 OPERATOR AND COMPARATOR CONTRACTS

Table 9 makes the assembly intervention explicit at the level used by the crossed-depth comparison. Every operator presents the same rank-ordered prefix and one assembly contract to the generator under the common 4,096-token consumed-context cap; the generator writes the resulting state and answer in one invocation. The intervention changes the state contract and update behavior, not the retrieved passages or their order.

Table 9: Assembly contracts. The first three operators form the 21-action serving set; the remaining operators test which properties are associated with transition movement.

| Operator | Carried representation | Read–update–write behavior | Experimental role |
|---|---|---|---|
| CoT | free-form reasoning trace | reads the admitted prefix and carries untyped reasoning without field-level merge or hop closure | selectable reference |
| Structured-base | sectioned free-form text | preserves a fixed visual layout while leaving entity, evidence, and open-hop semantics unspecified | selectable layout control |
| Manual Ledger | entities, supporting evidence, unresolved hops, provisional answer | merges retained entries, removes supported hops, and revises the answer only from retained evidence | selectable state operator |
| Random schema | fields with task-unrelated labels | preserves fielded syntax without a task-state contract | schema-form control |
| Minimal scaffold | reduced task-state scaffold | applies the same rank-ordered update cadence with fewer explicit state distinctions | state-content control |
| Auto-scaffold | task-level fields proposed from a task description and five examples | freezes the generated schema before evaluation and updates it after each admitted prefix | generated-state control |

The Manual-Ledger state is serialized in the fixed field order:

```
ENTITIES; SUPPORTING EVIDENCE;
UNRESOLVED HOPS; PROVISIONAL ANSWER.
```

At update $j$, the generator applies the fixed contract to state $S_{j-1}$ and the newly admitted rank-ordered passage, merges entity and evidence entries into $S_j$, removes an unresolved hop only when retained evidence supports its resolution, and revises the provisional answer only from the retained evidence set. The serialized state is formed before the shared context cap is applied; no operator may reorder passages or exceed the cap. Table 18 reports the resulting pre-truncation lengths, making token-footprint differences auditable beside the semantic controls.

Table 10: Decision-comparator contract. Shared-stack rows use the corpus, generator, decoding, normalization, hardware, and end-to-end latency accounting in Section 4.1.

| Comparator | Decision signal | Retained method behavior | Matched boundary |
|---|---|---|---|
| Learned-$k_r$ | workload, first-hop score tertile, calibrated depth utilities and latency | predicts utility separately for each of seven depths | same 100-query calibration, SLO test, ties, and fallback |
| Self-RAG | reflection/control tokens | published retrieval-trigger and critique logic | common retrieval adapter and stack |
| CRAG | retrieval evaluation and corrective branch | published correction logic and stopping behavior | common retrieval adapter and stack |
| Iter-RetGen | generated state from the preceding round | published iterative retrieval–generation control flow | common retrieval adapter and stack |
| DEPTHLENS | fitted assembly-conditioned transition and factorized latency | exact enumeration over the calibrated feasible actions | same calibration and latency boundary |

### A.3 CALIBRATION SWEEP AND ASSEMBLY-CONDITIONED TRANSITIONS

The fractional knot summarizes the joined two-regime fit rather than selecting one sweep row. The 78.2 value is the $k_r = 10$ mean on the 100-query calibration split. The 78.4 values in Tables 8 and 22 reuse the same joint-selector operating point on the 1,000-query HotpotQA evaluation split. Tables 3, 6, and 18 reuse the 78.1 Ledger-fixed operating point on that split. Table 4 uses a separate

Table 11: HotpotQA calibration sweep with Manual Ledger ($n = 100$). $\chi = \Delta\text{EM}/\Delta \log k_r$; the continuous fit gives $k_r^c = 4.1$. The displayed latency subtotal contains only the two named components and is distinct from end-to-end P95.

| $k_r$ | EM (%) | $\chi$ | Retrieval (ms) | Generation (ms) | Shown subtotal (ms) |
|---|---|---|---|---|---|
| 3 | 68.4 | – | 42 | 1280 | 1322 |
| 5 | 77.1 | 17.0 | 58 | 1420 | 1478 |
| 7 | 78.4 | 3.9 | 76 | 1580 | 1656 |
| 10 | 78.2 | $-0.6$ | 102 | 1780 | 1882 |
| 12 | 77.6 | $-3.3$ | 118 | 1920 | 2038 |
| 15 | 76.8 | $-3.6$ | 142 | 2120 | 2262 |
| 20 | 75.2 | $-5.6$ | 178 | 2420 | 2598 |

1,000-query controlled-stress slice at fixed $k_r = 10$. Reranking, prompt assembly, and queueing are omitted from the calibration table's component subtotal; all reported P95 values include them through the factorization in Section 3.

Table 12: Assembly-conditioned retrieval-demand shift from continuous two-regime fits. Knot values are estimate $\pm$ paired-bootstrap 95% half-width; Shift is the point estimate $\rho_{\text{UD}}$ from Eq. 3. "Pass" means the segmented fit improves leave-one-depth-out error, has an interior optimum, and its interval spans at most two adjacent depth intervals.

| Dataset | Hops | CoT $k_r^c$ | Manual Ledger $k_r^c$ | Shift | Fit |
|---|---|---|---|---|---|
| HotpotQA | 2 | $11.2 \pm 1.4$ | $4.1 \pm 0.6$ | 2.7× | Pass |
| 2Wiki | 2–3 | $12.8 \pm 1.6$ | $4.6 \pm 0.7$ | 2.8× | Pass |
| MuSiQue | 3–4 | $16.2 \pm 1.8$ | $5.2 \pm 0.8$ | 3.1× | Pass |
| FEVER | 1 | $7.4 \pm 1.0$ | $3.4 \pm 0.4$ | 2.2× | Pass |

Table 13: Transition stability across calibration seeds. Triples give $k_r^c$ for seeds 11/42/2027; depth distributions pool assignments from the three seed-specific selectors, so their support may exceed the three score buckets of one selector.

| Dataset | Assembly | Seed-wise $k_r^c$ | Selected-depth distribution |
|---|---|---|---|
| HotpotQA | CoT | 11.2/11.5/10.9 | $k = 7/10/12/15$: 6/45/38/11% |
| | Ledger | 4.1/4.3/4.0 | $k = 3/5/7$: 18/57/25% |
| 2Wiki | CoT | 12.8/12.4/13.1 | $k = 7/10/12/15/20$: 4/21/42/27/6% |
| | Ledger | 4.6/4.8/4.5 | $k = 3/5/7/10$: 13/54/29/4% |
| MuSiQue | CoT | 16.2/15.7/16.6 | $k = 10/12/15/20$: 8/23/48/21% |
| | Ledger | 5.2/5.5/5.0 | $k = 3/5/7/10$: 8/49/36/7% |
| FEVER | CoT | 7.4/7.1/7.6 | $k = 3/5/7/10$: 21/36/35/8% |
| | Ledger | 3.4/3.6/3.3 | $k = 3/5/7$: 47/45/8% |

Table 14: Global fit diagnostics retained for calibration inspection. The main text reports direct boundary ranking and regret.

| Dataset | MAE (EM pts.) | $R^2$ | MAPE (%) | Latency MedAE (s) |
|---|---|---|---|---|
| HotpotQA | 0.86 | 0.95 | 5.8 | 0.12 |
| 2Wiki | 0.94 | 0.94 | 6.4 | 0.14 |
| SEC 10-K (no update) | 1.82 | 0.88 | 9.4 | 0.22 |
| SEC 10-K (200-query update) | 1.08 | 0.93 | 6.8 | 0.15 |

### A.4 Ledger-fixed sensitivities and joint action frequencies

For the canonical evaluation seed, each workload's three score buckets map deterministically to one selected action; the counts in Table 16 are the evaluation-query populations assigned to those bucket actions. Table 13 instead pools assignments from three separately calibrated selectors. A query can contribute more than one candidate pair within 0.15 seconds of the SLO, so the pair denominators in Table 7 can exceed 1,000 even though its boundary-query denominator cannot.

### A.5 Assembly controls and cross-backbone durability

Table 15: Primary-utility advantage of DEPTHLENS +Manual Ledger over Learned-$k_r$+Manual Ledger across SLOs. Each condition retains the seven-depth grid and three calibration seeds.

| SLO (s) | HotpotQA | 2Wiki | MuSiQue | FEVER |
|---|---|---|---|---|
| 1.5 | +1.8 | +2.1 | +2.4 | +0.6 |
| 2.0 | +2.4 | +2.6 | +2.8 | +0.8 |
| 2.5 | +2.6 | +2.8 | +2.9 | +0.9 |
| 3.0 | +2.2 | +2.5 | +2.6 | +0.8 |

Table 16: Assembly actions selected by the joint DEPTHLENS row in Table 8; each dataset has 1,000 evaluation queries.

| Dataset | Manual Ledger | Structured-base | CoT |
|---|---|---|---|
| HotpotQA | 841/1000 | 106/1000 | 53/1000 |
| 2Wiki | 867/1000 | 91/1000 | 42/1000 |
| MuSiQue | 889/1000 | 74/1000 | 37/1000 |
| FEVER | 707/1000 | 196/1000 | 97/1000 |

Table 17: Paired primary-utility uncertainty for joint DEPTHLENS versus Learned-$k_r$ in Table 8. W/T/L gives query-level win/tie/loss counts; every denominator is 1,000.

| Dataset | $\Delta U$ | 95% paired CI | W/T/L |
|---|---|---|---|
| HotpotQA | +11.6 | [8.1, 15.1] | 224/668/108 |
| 2Wiki | +11.8 | [8.2, 15.4] | 239/640/121 |
| MuSiQue | +12.2 | [8.4, 15.9] | 251/620/129 |
| FEVER | +3.2 | [0.6, 5.8] | 104/824/72 |

Table 18: Scaffold controls under DEPTHLENS depth selection. "Pre-trunc." is mean assembled length before the common 4,096-token context truncation; every model consumes at most 4,096 tokens.

| | HotpotQA | | | SEC 10-K | | |
|---|---|---|---|---|---|---|
| Assembly | EM | $k_r^c$ | Pre-trunc. | EM | $k_r^c$ | Pre-trunc. |
| CoT | 64.8 | 11.2 | 4096 | 56.2 | 14.8 | 4096 |
| Structured-base | 67.4 | 10.4 | 4096 | 58.8 | 13.6 | 4096 |
| Random schema | 67.8 | 10.2 | 4120 | 59.2 | 13.4 | 4118 |
| Minimal scaffold | 72.6 | 6.8 | 3820 | 65.4 | 8.4 | 3840 |
| Auto-scaffold | 77.1 | 4.8 | 3710 | 69.8 | 5.6 | 3725 |
| Manual Ledger | 78.1 | 4.1 | 3680 | 71.2 | 5.1 | 3695 |

Table 19: Hop-stratified Auto-scaffold versus Manual Ledger on the pooled four-dataset evaluation set. $U$ is the primary utility; deltas and intervals are paired over the $n$ queries in each stratum.

| Hops | $n$ | Auto $U$/F1 | Ledger $U$/F1 | $\Delta U$ | 95% CI |
|---|---|---|---|---|---|
| 1 | 1030 | 88.1/90.0 | 89.3/91.0 | +1.2 | [0.2,2.2] |
| 2 | 1320 | 75.8/80.5 | 78.1/82.6 | +2.3 | [1.1,3.5] |
| 3 | 1050 | 66.2/71.8 | 69.8/75.2 | +3.6 | [2.0,5.2] |
| 4 | 600 | 56.7/62.4 | 61.5/66.9 | +4.8 | [2.4,7.2] |

Table 20: Full Qwen2.5-72B-Instruct assembly cross with each assembly fixed under the DEPTHLENS depth selector. Cells report primary utility / fitted $k_r^c$; bracketed ranges for endpoint operators are paired-bootstrap 95% intervals. The joint selector in Table 5 may choose among assemblies.

| Assembly | HotpotQA | 2Wiki | MuSiQue | FEVER |
|---|---|---|---|---|
| CoT | 64.1/10.8 [9.5,12.1] | 60.8/12.4 [10.8,14.0] | 53.2/15.6 [14.0,17.3] | 85.7/7.1 [6.2,8.0] |
| Structured-base | 66.4/10.1 | 62.9/11.7 | 55.1/14.8 | 86.4/6.8 |
| Random schema | 66.1/10.3 | 62.6/11.9 | 54.8/14.9 | 86.2/6.9 |
| Auto-scaffold | 76.4/4.9 | 72.4/5.3 | 64.8/6.1 | 89.1/3.8 |
| Manual Ledger | 77.4/4.3 [3.8,4.9] | 73.3/4.8 [4.1,5.5] | 65.4/5.4 [4.6,6.2] | 89.4/3.4 [3.0,3.9] |

Table 21: Qwen2.5-72B-Instruct decision details for Table 5. $U$ is the primary utility; L-$k_r$+L denotes Learned-$k_r$+Manual Ledger, while DL jointly selects depth and assembly. W/T/L compares them over 1,000 queries.

| Dataset | L-$k_r$+L | | | DL | | | W/T/L |
|---|---|---|---|---|---|---|---|
| | $U$ | F1 | P95 | $U$ | F1 | P95 | |
| HotpotQA | 74.9 | 79.5 | 1.78 | 77.6 | 82.0 | 1.73 | 192/643/165 |
| 2Wiki | 70.6 | 75.8 | 1.73 | 73.6 | 78.4 | 1.67 | 208/614/178 |
| MuSiQue | 62.8 | 68.5 | 1.69 | 65.9 | 71.4 | 1.63 | 218/595/187 |
| FEVER | 88.9 | 90.8 | 1.55 | 89.7 | 91.5 | 1.51 | 98/812/90 |

### A.6 Official implementations beyond HotpotQA

With published control logic and stopping rules retained behind a common retrieval adapter, DepthLens exceeds the highest-EM official-code row by 10.0 EM on HotpotQA, 11.2 on 2Wiki, and 10.8 on MuSiQue. Tables 22–23 report the matched normalization, latency, and throughput evidence.

Table 22: HotpotQA official-code comparison ($n = 1{,}000$, concurrency 16). Official control tokens and stopping rules are retained; retrieval calls use the common corpus adapter and all outputs use identical normalization.

| Method | Source | EM | F1 | P95 (s) | QPS |
|---|---|---|---|---|---|
| Iter-RetGen | official | 68.4 | 73.1 | 4.38 | 2.2 |
| Self-RAG | official | 65.2 | 70.2 | 2.32 | 3.8 |
| CRAG | official | 66.8 | 71.6 | 2.48 | 3.6 |
| Learned-$k_r$ | matched | 66.8 | 71.9 | 1.62 | 5.5 |
| DepthLens | matched | 78.4 | 82.7 | 1.68 | 5.4 |
| DepthLens selector + CoT | matched | 64.8 | 69.8 | 2.14 | 4.2 |

The final row preserves the DepthLens selector, seven-depth grid, retriever, reranker, generator, split, and budget while fixing assembly to CoT. Its 13.6-point gap from the joint row measures the policy-level effect of replacing joint assembly choice with fixed CoT under the same decision procedure. Against the highest-EM official row, DepthLens gains 10.0 EM with paired CI [7.4,12.6] and W/T/L 207/686/107.

Table 23: Official-code extensions to 2Wiki and MuSiQue. All cells use 1,000 matched queries at concurrency 16.

| Dataset | Method | EM | F1 | P95 (s) | QPS |
|---|---|---|---|---|---|
| 2Wiki | Self-RAG | 59.8 | 66.2 | 2.29 | 3.8 |
| | CRAG | 61.9 | 68.0 | 2.42 | 3.5 |
| | Iter-RetGen | 63.0 | 68.8 | 4.55 | 2.1 |
| | Learned-$k_r$ | 62.8 | 68.7 | 1.61 | 5.3 |
| | DepthLens | 74.2 | 79.1 | 1.62 | 5.2 |
| MuSiQue | Self-RAG | 52.8 | 58.7 | 2.21 | 3.9 |
| | CRAG | 54.7 | 60.1 | 2.36 | 3.6 |
| | Iter-RetGen | 56.0 | 61.4 | 4.42 | 2.2 |
| | Learned-$k_r$ | 55.0 | 60.5 | 1.58 | 5.4 |
| | DepthLens | 66.8 | 72.3 | 1.58 | 5.3 |

The strongest official adaptive row is Iter-RetGen on both extensions. DepthLens improves by 11.2 EM on 2Wiki (CI [8.4,14.0], W/T/L 222/668/110) and 10.8 on MuSiQue (CI [7.9,13.7], W/T/L 226/656/118). The common adapter replaces each implementation's external corpus interface while retaining its published control flow, critique tokens, and stopping logic; this isolates controller behavior from retrieval-stack differences.

### A.7 Serving replays

The primary 5,000-query replay reaches 77.8 primary utility, 2.18-second P95, and 7.6 q/s SLO goodput. Under injected reranker drift, 45 queries elapse before detection; after the fixed 200-query collection and refit window, 180 additional queries elapse before quality and latency re-enter the

operating band. A second 5,000-query replay combines retrieval-recall degradation with query-mix shift and retains a 0.8 q/s goodput advantage over Learned-$k_r$+Ledger across three arrival seeds. These operating results preserve the ordering established by the crossed curves and fixed-assembly decision test.

Table 24: Primary 5,000-query replay: HotpotQA/MuSiQue/FEVER $= 60/25/15\%$, Poisson arrivals, retriever drift at query 3,000. Violation counts cover the recovery interval.

| Metric | DEPTHLENS | Learned-$k_r$+Ledger | Iter-RetGen |
|---|---|---|---|
| Primary utility | 77.8 | 73.2 | 67.1 |
| P95 latency (s) | 2.18 | 2.32 | 5.62 |
| SLO goodput (q/s) | 7.6 | 6.8 | 4.4 |
| QPS at concurrency 64 | 9.2 | 8.4 | 5.1 |
| Plan-cache hit rate | 72.4% | – | – |
| Detection queries | 45 | – | – |
| Recovery queries | 180 | – | – |
| Recovery SLO violations | 90/5000 | 210/5000 | 1930/5000 |

The three replay clocks measure distinct events: 45 queries from injected drift to detection, a fixed 200-query collection window before refitting, and 180 post-refresh queries until quality and latency re-enter the pre-drift band. DEPTHLENS improves SLO goodput by 1.7× over Iter-RetGen and by 0.8 q/s over the Ledger-fixed learned-depth policy.

Table 25: Held-out 5,000-query replay over 2Wiki/SEC 10-K/FEVER $= 45/35/20\%$, with retrieval-recall degradation at query 2,500 and query-mix shift at 3,800. Means and standard deviations use three arrival seeds.

| Method | Utility | P95 (s) | Goodput | QPS | Detect | Recover |
|---|---|---|---|---|---|---|
| DEPTHLENS | $73.4 \pm 0.5$ | $2.26 \pm 0.04$ | $7.1 \pm 0.2$ | $8.8 \pm 0.2$ | $52 \pm 8$ | $211 \pm 24$ |
| Learned-$k_r$+Ledger | $69.8 \pm 0.6$ | $2.41 \pm 0.05$ | $6.3 \pm 0.2$ | $8.0 \pm 0.3$ | $88 \pm 12$ | $318 \pm 31$ |
| Iter-RetGen | $64.9 \pm 0.7$ | $5.78 \pm 0.09$ | $4.0 \pm 0.1$ | $4.8 \pm 0.2$ | – | – |

### A.8 FAILURE AUDIT

Table 26: Primary-cause audit of 80 suboptimal HotpotQA plans. Counts and percentages share the same denominator.

| Failure mode | Count | Share |
|---|---|---|
| Reranker mis-ranking | 36/80 | 45.0% |
| Ambiguous query | 22/80 | 27.5% |
| High distractor density | 14/80 | 17.5% |
| Output-budget pressure | 8/80 | 10.0% |