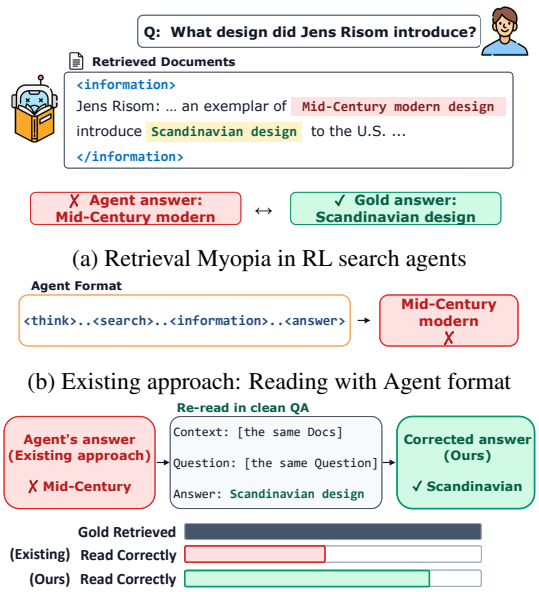
Retrieval Myopia: Why RL Search Agents Fail to Read What They Find

Anonymous ACL submission

Abstract

RL-trained search agents interleave retrieval with reasoning to answer knowledge-intensive questions, achieving strong end-to-end accuracy on multi-hop QA benchmarks. Yet end-to-end accuracy conflates two distinct abilities: *searching* for relevant documents and *reading* what they contain. Existing evaluations do not measure whether the agent reads the documents it retrieves. We design a controlled evaluation that force-injects gold-bearing documents into each agent's retrieval pipeline, isolating reading from retrieval. Across five RL search agents on three multi-hop QA benchmarks, up to 46.2% of samples are answered incorrectly even when the gold answer is *literally present* in the retrieved documents—retrieval succeeds but reading fails. We term this phenomenon **Retrieval Myopia** (**RM**). The agent's interaction format is itself a substantial cause: simply re-presenting the same documents to the *same model* in a plain QA format, with no weight updates, recovers up to 48.9% of RM cases. Guided by this diagnosis, we propose **CWFT** (**C**onfidence-**W**eighted **F**ormat **T**ransfer), a training-free pipeline that anchors the agent's prediction against k confidence-weighted clean-format re-reads. CWFT consistently improves accuracy across all evaluated system–dataset configurations on three multi-hop QA benchmarks.



(a) Retrieval Myopia in RL search agents

(b) Existing approach: Reading with Agent format

(c) Ours: Format Transfer by Confidence-Weight

Figure 1: **(a) Retrieval Myopia (RM)**: the agent retrieves a document containing the gold answer but emits a wrong noun phrase—retrieval succeeds, reading fails. **(b)** Existing RL search agents read through a rigid **agent format** (`<think>`/`<search>`/`<information>`/`<answer>`); this scaffolding suppresses reading even when the gold answer is literally present. **(c) Confidence-Weighted Format Transfer (CWFT)** re-reads the same documents in clean QA formats, weighted by self-confidence—no parameter update required.

1 Introduction

RL-trained search agents—Search-R1 (Jin et al., 2025), R1-Searcher (Song et al., 2025), AutoRefine (Yaorui et al., 2025), s3 (Jiang et al., 2025), HiPRAG (Wu et al., 2025)—interleave retrieval with chain-of-thought reasoning, achieving strong results on knowledge-intensive QA. Their training shares a single recipe: an outcome reward on final-answer correctness, combined with *retrieved-token masking* that zeros out gradients on the documents the agent reads. The reward signal drives the model to produce correct final answers, but it does not decompose success into its constituent abilities—*searching* for relevant documents versus *reading* what they contain.

This decomposition has not been performed. Prior work evaluates these agents on end-to-end accuracy, implicitly assuming that improvements reflect gains in both abilities. We challenge that assumption: an agent can score well on standard accuracy with strong search and weak reading (or vice versa), and the two failure modes call for very

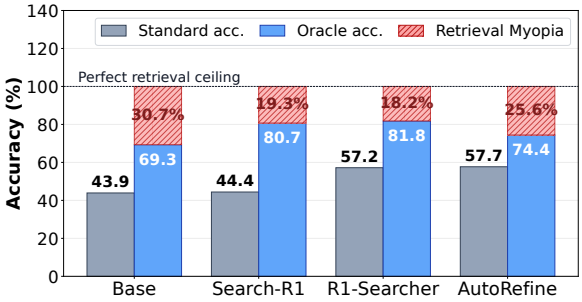


Figure 2: Oracle RM on HotpotQA: accuracy when gold-bearing documents are force-injected into each system's retrieval pipeline. The residual 18.2–30.7% is pure reading failure.

different remedies. We provide the first decomposed measurement of search and reading ability across multiple RL search agents on multi-hop QA, isolating reading by force-injecting gold documents into each agent's retrieval pipeline.

Our headline finding—Figure 1 illustrates a canonical instance—is that even when retrieval is guaranteed to succeed, up to 46.2% of samples remain incorrectly answered. The agent retrieves the document containing the gold answer, often quotes the gold-bearing sentence in its own reasoning, and still emits the wrong answer. We call this phenomenon **Retrieval Myopia** (RM). RM specialises a broader pattern of reading failure documented in plain-prompting RAG—positional bias (Liu et al., 2024), distractor sensitivity (Shi et al., 2024a), prompt-format sensitivity (Sclar et al., 2024), and outright ignoring of retrieved evidence (Asai et al., 2024)—to the RL-search-agent setting, with a previously unidentified causal driver. A counterfactual diagnosis traces a substantial fraction of these RM cases (48.9% for Search-R1) to the rigid agent interaction format itself: re-presenting the same documents to the *same model* in a plain QA format recovers these cases without any change to the model.

Contributions.

- **First decomposed measurement** of search vs. reading in RL search agents: even with perfect retrieval, up to 46.2% of samples remain wrong—reading is the bottleneck (Sec. 3).
- **Causal isolation** of the agent interaction format as a substantial cause of reading failure: a confound-controlled comparison shows clean wrapping beats agent wrapping by +9.8–20.8 **pp** ($p < 10^{-4}$; Sec. 3.5).
- **CWFT**, a training-free pipeline (anchor + confidence-weighted clean-format re-reads) that universally improves accuracy across all 15 evaluated configurations (mean +1.2 **pp**; Sec. 4–5).

2 Related Work

RL-trained search agents. A recent line of work trains language models to interleave multi-step search with reasoning via reinforcement learning, building on retrieval-augmented generation (Lewis et al., 2020): Search-R1 (Jin et al., 2025), R1-Searcher (Song et al., 2025), AutoRefine (Yaorui et al., 2025), s3 (Jiang et al., 2025), and HiPRAG (Wu et al., 2025). These systems vary in RL algorithm (GRPO (Shao et al., 2024) in Search-R1; PPO (Schulman et al., 2017) in AutoRefine and HiPRAG; 2-stage SFT-then-RL in R1-Searcher), reward composition (outcome-only EM versus process-shaped rewards on search or refinement steps), and tool invocation pattern (inline special tokens versus a selector–generator decomposition in s3). Despite this variation, they converge on two design choices we critique: (i) a structured interaction format with explicit tags (`<think>`, `<search>`, `<information>`, `<answer>`) that segments reasoning, queries, retrieved documents, and final answer; and (ii) *retrieved-token masking* during RL training, which zeros out gradients on tokens within retrieved-document blocks. Evaluation is uniformly end-to-end on QA accuracy, conflating retrieval and reading failures—a gap our controlled evaluation fills by isolating the two abilities and exposing a previously unmeasured reading bottleneck.

Inference-time correction and self-confidence. Many training-free methods aim to correct LLM outputs at inference time: self-consistency (Wang et al., 2022b), universal self-consistency (Chen et al., 2023), self-refine (Madaan et al., 2023), reflexion (Shinn et al., 2023), chain-of-verification (Dhuliawala et al., 2024), context-aware decoding (Shi et al., 2024b; Wang et al., 2025), self-RAG (Asai et al., 2024), and corrective RAG (Yan et al., 2024), among others. These are largely *problem-agnostic*: they apply a generic correction without diagnosing the source of failure. In contrast, our method is derived from a diagnosis—a counterfactual experiment identifies the agent format as a specific cause of RM, and CWFT directly targets that cause via format replacement, bounding the scope of any

System	Acc	Retr. Hit	#RM	RM Rate
Base	43.9%	47.6%	1,137	15.4%
Search-R1	44.4%	63.4%	1,770	**23.9%**
R1-Searcher	57.2%	70.3%	1,298	17.5%
AutoRefine	57.7%	66.2%	1,073	14.5%
s3 (7B)	46.5%	59.7%	1,353	18.3%

Table 1: Standard-setting RM on HotpotQA. *Retr. Hit*: fraction of samples with gold in retrieved docs. *#RM*: count of (gold-in-docs $\wedge$ wrong) samples; *RM Rate* = #RM / total.

System	Standard Acc	Oracle Acc	Oracle RM
Base	43.9%	69.3%	30.7%
Search-R1	44.4%	80.7%	**19.3%**
R1-Searcher	57.2%	81.8%	**18.2%**
AutoRefine	57.7%	74.4%	25.6%
s3 (7B)	46.5%	70.1%	29.9%

Table 2: Oracle-retrieval RM on HotpotQA. *Oracle Acc*: accuracy when gold docs are guaranteed in context; *Oracle RM* $= 1 -$ Oracle Acc (pure reading failure).

inference-time remedy to the format-induced subset of RM. CWFT uses the P(True) probe (Kadavath et al., 2022) as a per-re-read weight in a confidence-weighted vote, with a conservative anchor on the agent's prediction for preservation.

3 Diagnosing Retrieval Myopia

We first measure how often retrieved documents contain the gold answer yet the agent still fails to extract it (Sec. 3.1–Sec. 3.3), then run a counterfactual experiment that pinpoints the agent interaction format as a substantial cause of the residual reading failure (Sec. 3.4). The diagnosis prescribes the form of any training-free remedy, motivating CWFT in Sec. 4.

3.1 Definition

A sample exhibits **Retrieval Myopia** (RM) when (i) the gold answer appears verbatim in the retrieved documents (substring match after lowercasing and punctuation/article normalisation), and (ii) the model's final answer is incorrect under the same normalisation with bidirectional substring matching (App. A.4). RM isolates one failure mode: retrieval succeeded, reading did not.

3.2 Standard RM

We first measure RM using each system's own retrieval pipeline (top-3, E5 retriever (Wang et al., 2022a), Wikipedia 2018 dump; DPR-style dense retrieval (Karpukhin et al., 2020)) on HotpotQA (Yang et al., 2018) (Table 1).

Standard RM affects **14.5–23.9%** of samples across systems. The ordering is misleading: **Search-R1 has the highest RM rate** (23.9%) **despite matching the base model's accuracy** (44.4% vs. 43.9%)—its stronger search retrieves gold in 63.4% of samples (vs. Base's 47.6%), exposing more reading-failure opportunities. Standard RM thus conflates retrieval quality with reading ability. We isolate reading next.

3.3 Oracle RM: reading with perfect retrieval

We force-inject gold-containing documents into each system's retrieval pipeline. Errors that remain are pure reading failures (Figure 2; Table 2).

Three patterns emerge. **(i)** Reading is the bottleneck: 18.2–30.7% of samples remain wrong even when gold is guaranteed in context. **(ii)** RL training improves reading only partially—oracle RM drops from 30.7% (Base) to 18.2–25.6% (RL-trained), with the best system (R1-Searcher) still misreading roughly one sample in five. **(iii)** The standard-RM ordering reverses: Search-R1, the *worst* on standard RM (23.9%, Table 1), becomes the *second-best* on oracle RM (19.3%), revealing its high standard rate as an artifact of strong retrieval surfacing more reading failures, not weaker reading. The 36.3 pp gap between Search-R1's standard (44.4%) and oracle (80.7%) accuracy thus decomposes into a retrieval contribution and an irreducible 19.3% reading-failure residual, whose cause we investigate next.

The reading-bottleneck pattern is not HotpotQA-specific: on 2WikiMultiHopQA (Ho et al., 2020) and MuSiQue (Trivedi et al., 2022), oracle RM rises to 18.2–46.2% across the same systems (full table in App. B.1).

3.4 Format-induced suppression

The oracle-RM result establishes the phenomenon but does not identify the cause: do these residual failures (18–30% on HotpotQA, rising to 46% on the harder multi-hop benchmarks) reflect a fundamental reading-capability limit of the 7B backbone, or something specific to how the agent reads? We test this counterfactually by extracting the documents that each agent retrieved and re-presenting them to the *same model* in a plain QA format—model, documents, and question are identical, only the interaction format changes.

Format transfer recovers 28.4–48.9% of RM cases without any modification to model weights

System	RM Cases	Recovered	Recovery
Search-R1	1,770	866	**48.9%**
R1-Searcher	1,298	369	28.4%
AutoRefine	1,073	325	30.3%
Base	1,137	422	37.1%

Table 3: Format-transfer recovery on HotpotQA RM cases (same model, same documents; agent format → plain QA). s3 excluded: its selector–generator decomposition has no single agent-format trace.

Format	Reading Acc
Constraint-aware	**77.8%**
Simple QA	71.8%
Clean QA	70.5%
Extractive	68.4%
Chain-of-thought	33.7%
Agent format (P1)	62.3%

Table 4: Reading accuracy of five clean-format prompts on Search-R1 HotpotQA RM samples.

(Table 3). A substantial fraction of RM is therefore **format-induced**: the model *can* read these documents but is *prevented from doing so* by the agent interaction structure. Recovery is highest for Search-R1 (48.9%), which has no post-retrieval processing, and lower for systems whose design partially absorbs the effect (R1-Searcher, AutoRefine).

Which clean-format variants work best? We evaluate five re-reading prompts on Search-R1 RM samples (Table 4). **Constraint-aware** (a template that lists the question's named entities and asks for the target entity type, with a strict 1–3 word answer budget) is the strongest at +15.5 pp over the agent baseline, while **chain-of-thought** *degrades* below baseline—the gain comes from removing scaffolding and constraining the output, not from adding reasoning. No single prompt dominates the others by a wide margin, and the spread across prompts motivates the multi-prompt aggregation in Sec. 4.

3.5 Isolating the format effect: a confound-controlled comparison

A natural concern is that the 27–49% recovery in Sec. 3.4 reflects *any second pass* with sampling diversity rather than the format swap specifically. To isolate the wrapping as the causal variable, we run a controlled comparison on each Search-R1 RM sample. **Condition A** replays the agent's trace up to the `<answer>` tag and samples $K=4$ answers at temperature 0.7; **Condition B** uses the same model, documents, question, K, temperature, and a single "Based on the documents below, answer the question" template—only the agent format wrapping is removed. We report majority-vote accuracy (the deployable selection rule); Pass@K as a recovery ceiling is in App. B.2. Each cell uses McNemar paired tests and bootstrap 95% CIs.

Dataset	n_{RM}	Agent	Clean	Δ	p
HotpotQA	1,766	31.9%	45.5%	+13.6	7×10^{-25}
2Wiki	1,820	14.0%	34.8%	+20.8	3×10^{-55}
MuSiQue	326	8.9%	18.7%	+9.8	8×10^{-5}

Table 5: Confound-controlled format isolation on Search-R1 RM samples (majority vote over $K=4$). Only the format wrapping differs across conditions. p: McNemar paired test; bootstrap CIs and Pass@4 in App. B.2.

Across all three datasets, clean wrapping outperforms agent wrapping by +9.8–+20.8 **pp**, with $p < 10^{-4}$ in every configuration (Table 5).

This rules out the "any second pass helps" explanation and converts the diagnosis in Sec. 3.4 from a correlational observation into a controlled causal finding: the agent interaction format itself—not the underlying reading capability of the 7B backbone—suppresses reading on a sizable fraction of RM samples.

Diagnosis summary. The format-transfer (Sec. 3.4) and confound-controlled isolation (Sec. 3.5) experiments together establish that a substantial fraction of RM is not a reading-capability limit of the 7B backbone but a *format-induced* suppression that disappears when the same documents are read in a clean QA format. This prescribes the form of any training-free remedy:

- **(D1) Format swap is the primary lever.** Recovery of 28.4–48.9% of RM via format replacement alone, without any other intervention, identifies format swap as the only known training-free mechanism that meaningfully reduces RM.
- **(D2) No single re-reading prompt dominates.** The variance across re-reading formats (Table 4) means a single prompt is unreliable; a remedy must aggregate across multiple prompts.
- **(D3) The remedy must protect samples already answered correctly.** The format-induced subset is a fraction of all RM, and RM is a fraction of all samples; an indiscriminate override would damage the (majority of) samples where

the agent's prediction is already correct.

These three constraints define the design space for the method in Sec. 4.

4 Method

We introduce **Confidence-Weighted Format Transfer (CWFT)**, a post-hoc, training-free pipeline that corrects Retrieval Myopia without modifying retrieval, training, or model weights. CWFT composes three components—a multi-prompt clean-format re-read, a per-re-read confidence-and-grounding gate, and an anchored weighted aggregation over the agent's prediction—and outputs a single answer (Algorithm 1 in App. A.1).

Each component is dictated by one of the diagnostic constraints (D1)–(D3) established in Sec. 3: **(D1)** format swap as the primary causal lever, **(D2)** multi-prompt aggregation with confidence gating, and **(D3)** a conservative anchor on the agent's prediction.

4.1 Notation

Given a question q, the unmodified agent's prediction P1, and the documents D it retrieved, CWFT uses the same backbone M, a set of k clean-format re-reading prompts $\{P_1, \ldots, P_k\}$, a scalar anchor weight $\lambda \in \mathbb{R}_{\geq 0}$ on P1, and a normalisation function g (lower-case, article and punctuation removal, whitespace collapse) to produce a single predicted answer $\hat{y}$.

4.2 Clean-format re-reading

CWFT first replaces the agent format with a plain QA format and re-reads the same documents D. For each prompt template P_i, the backbone produces a candidate answer

$$\mathrm{CQA}_i \;=\; M\big(P_i(q, D)\big), \qquad i = 1, \ldots, k.$$

The format swap is the causal lever of (D1): the same M and the same D that the agent format reads incorrectly are re-presented to M in a clean QA wrapping, recovering 28.4–48.9% of RM behaviourally (Sec. 3.4) and exceeding the agent wrapping by +9.8 to +20.8 pp in a confound-controlled comparison (Sec. 3.5).

The $k = 4$ prompts cover four qualitatively different surface forms (*simple QA*, *concise extractive*, *constraint-aware*, *clean QA*; full templates in App. A.2). Our format-choice analysis (Table 4) shows no single prompt dominates, so prompt diversity (D2) is required to reduce variance across re-readings.

4.3 Confidence and grounding gating

Each candidate CQA_i is gated by two signals.

Grounding. A hard filter requires that CQA_i appears as a substring of D under $g(\cdot)$. Ungrounded candidates—answers M produced but that are not literally supported by the retrieved documents—are discarded as likely hallucinations.

Confidence. For each surviving candidate we compute the P(True) probe of Kadavath et al. (2022):

$$w_i \;=\; P_M(\text{True} \mid q, D, \mathrm{CQA}_i) \in [0, 1],$$

the model's self-assessed probability that CQA_i is correct given the documents D. A grounded re-read enters the aggregation with weight w_i; an ungrounded one with weight 0.

Together the two filters implement (D2): they admit a re-read to the vote only when the model itself believes the re-read is both correct and supported by the retrieved documents.

4.4 Anchored weighted aggregation

CWFT aggregates P1 and the grounded re-reads through a weighted majority vote with a fixed anchor weight on P1. Let

$$\mathcal{C} \;=\; \{\mathrm{P1}\} \;\cup\; \{\mathrm{CQA}_i : \mathrm{CQA}_i \text{ grounded in } D\}$$

be the candidate set. Define the weight function

$$w(c) \;=\; \begin{cases} \lambda & \text{if } c = \mathrm{P1} \\ P_M(\text{True} \mid q, D, c) & \text{if } c = \mathrm{CQA}_i. \end{cases}$$

Group candidates by their normalised answer form $g(c)$ and sum the weights within each group. The aggregated CWFT prediction is

$$\hat{y}^{\,\mathrm{agg}} \;=\; g^{-1}\Big(\arg\max_{c^\star \in g(\mathcal{C})} \sum_{c \in \mathcal{C}\,:\, g(c) = c^\star} w(c)\Big). \tag{1}$$

Choice of λ. The anchor λ encodes (D3) numerically: with $w_i \in [0, 1]$ and $k = 4$, setting $\lambda = 3$ requires at least three high-confidence grounded re-reads to override P1 (a single $w_i \approx 1$ contributes $1 < \lambda$). App. B.3 confirms $\lambda = 3$ sits on the high-preservation Pareto frontier.

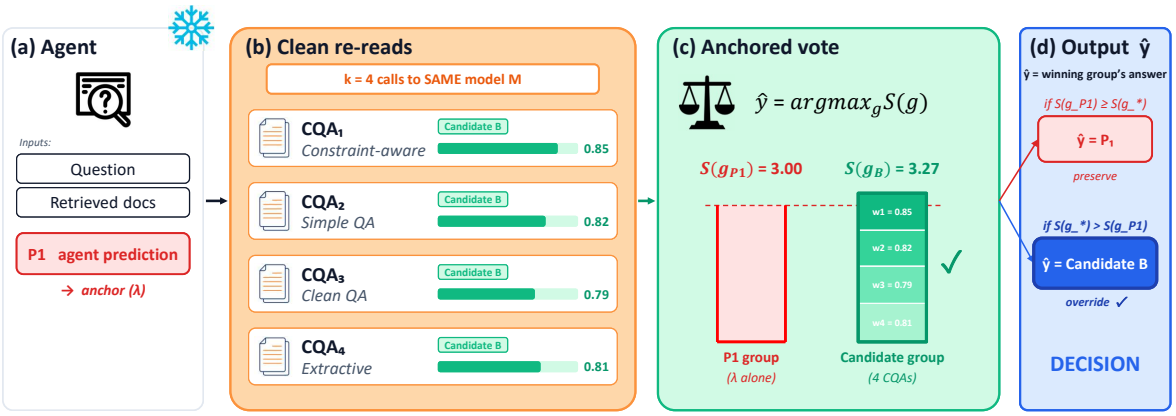


Figure 3: **CWFT pipeline.** Band 1: the agent's prediction P1 is anchored at weight λ. Band 2: the same backbone M produces $k = 4$ clean-format re-reads $\{\mathrm{CQA}_i\}$ weighted by $P(\mathrm{TRUE})$. Bands 3–4: candidates are grouped by normalised answer form and the arg-max-weighted group is emitted; P1 is preserved unless overridden.

Cascaded inference. The $\lambda = 3$ anchor admits a provably output-equivalent cascade that often skips most of the pipeline: if the first re-read CQA_1 is grounded and $g(\mathrm{CQA}_1) = g(\mathrm{P1})$, then P1 wins Eq. (1) regardless of $\mathrm{CQA}_2, \ldots, \mathrm{CQA}_k$ (Proposition 1 and proof in App. A.1). We therefore define CWFT operationally as the cascaded procedure (Algorithm 1 in App. A.1): generate CQA_1 first, return P1 if the early-exit test fires, otherwise continue to the remaining re-reads and apply Eq. (1).

5 Experiments

We evaluate CWFT under two questions: (Q1) Does CWFT improve accuracy across the five evaluated systems and multi-hop QA benchmarks, and are the gains statistically reliable (Sec. 5.1)? (Q2) Are all three components prescribed by (D1)–(D3) necessary, or does a simpler baseline suffice (Sec. 5.2)? We also verify that the gain pattern across systems is consistent with the diagnosis (Sec. 5.4).

Setup. We apply CWFT post-hoc to the publicly released checkpoints of four RL search agents (Search-R1, R1-Searcher, AutoRefine, s3) and a non-RL Base agent (Sec. 3.2) on up to three multi-hop QA datasets (HotpotQA, 2WikiMultiHopQA, MuSiQue). The re-reading backbone is the same Qwen2.5-7B-Instruct used by all five systems; CWFT does not modify retrieval, training, or weights. Unless stated otherwise, we use $k = 4$, $\lambda = 3$, and the P(True) confidence probe. Test sets follow the standard splits used by each agent's original paper. We report exact-match accuracy and *preservation*—the fraction of P1-correct samples that CWFT does not flip. Reproducibility details (re-reading prompts, decoding hyperparameters, normalisation) are in App. A.

5.1 Cross-system results

Table 6 reports P1 accuracy (the agent's original answer), CWFT accuracy, the absolute change (Δ), and preservation.

(i) Universally positive. CWFT improves accuracy on every evaluated system–dataset combination without any parameter update. Mean gain is $+1.2$pp; max is $+3.4$pp (Search-R1 on HotpotQA and 2Wiki).

(ii) Preservation is high by construction. Preservation is **98.8–99.5**% on every configuration: CWFT virtually never flips a P1-correct answer. The $\lambda = 3$ anchor is responsible for this property (Sec. 4.4).

Statistical significance. Per-configuration McNemar tests and 95% bootstrap CIs are in App. B.4. Eleven of fifteen configurations show statistically significant gains ($p < 0.01$, six of them $p < 10^{-10}$: Search-R1 on HotpotQA and 2Wiki, R1-Searcher on HotpotQA, s3 on HotpotQA and 2Wiki, and Base on HotpotQA). The remaining four (R1-Searcher / MuSiQue, AutoRefine on 2Wiki and MuSiQue, s3 / MuSiQue) have small effects ($\Delta \leq +0.5$ pp) whose CIs touch or span 0—CWFT is harmless on these systems but its gain is indistinguishable from 0. By contrast, the simpler baselines in Table 7 are paired-significantly *negative* on most configurations (e.g. CQA0: $\Delta < 0$, $p < 10^{-3}$ on most configurations)—a damage signal, not a recovery signal.

Method	HotpotQA		2Wiki		MuSiQue		Avg		Pres.
	EM	F1	EM	F1	EM	F1	EM	F1	
Baseline (no RL training)									
Base	43.1	40.7	37.4	33.7	19.1	20.7	33.2	31.7	–
+ CWFT	**45.2** (+2.1)	**41.9** (+1.2)	**38.2** (+0.8)	**34.6** (+0.9)	**20.4** (+1.3)	**21.3** (+0.6)	**34.6** (+1.4)	**32.6** (+0.9)	99.3%
RL search agents									
Search-R1	43.7	43.9	34.9	36.2	25.4	26.8	34.7	35.6	–
+ CWFT	**47.0** (+3.4)	**46.5** (+2.6)	**38.3** (+3.4)	**39.8** (+3.6)	**26.0** (+0.6)	**27.5** (+0.8)	**37.1** (+2.5)	**37.9** (+2.3)	99.2%
R1-Searcher	56.1	57.3	42.9	44.8	25.9	29.3	41.6	43.8	–
+ CWFT	**57.4** (+1.3)	**57.4** (+0.2)	**43.3** (+0.4)	**44.9** (+0.1)	**26.2** (+0.3)	**29.4** (+0.1)	**42.3** (+0.7)	**43.9** (+0.1)	99.0%
AutoRefine	56.8	57.1	46.9	46.0	30.8	32.0	44.8	45.0	–
+ CWFT	**57.9** (+1.1)	**57.4** (+0.2)	**47.3** (+0.4)	**46.5** (+0.5)	**31.3** (+0.5)	31.9 (−0.1)	**45.5** (+0.7)	**45.3** (+0.2)	98.8%
s3	45.6	25.7	43.5	13.0	13.9	7.2	34.3	15.3	–
+ CWFT	**46.6** (+1.0)	**26.1** (+0.4)	**44.1** (+0.7)	**13.1** (+0.1)	**14.0** (+0.1)	7.2 (∼0)	**34.9** (+0.6)	**15.5** (+0.2)	99.5%

Table 6: CWFT main results across five systems on three multi-hop QA benchmarks. Δ vs. P1 in parentheses. *Pres.*: fraction of P1-correct samples retained, system-averaged across datasets. EM uses bidirectional substring matching (App. A.4).

5.2 Component ablation

To verify that CWFT's three components (multi-prompt clean-format re-reading, P(True) gating, and $\lambda = 3$ anchor on P1) are each necessary, we evaluate four progressively-stronger baselines against the full CWFT method on all configurations, using the same re-reads, P(True) scores, and grounding filter. Baselines drop one or more (D1)–(D3) components:

- **CQA0 (single)**: a single clean-QA re-read replaces P1 if non-empty. Removes multi-prompt aggregation and the P1 anchor.
- **SC k=4 uniform**: $k = 4$ clean-QA re-reads, uniform majority vote without a P1 anchor (standard self-consistency (Wang et al., 2022b)). Removes P(True) gating and the P1 anchor.
- **USC-lite**: $k = 4$ clean-QA re-reads weighted by P(True), no P1 anchor (a lightweight variant of universal self-consistency (Chen et al., 2023)). Removes the P1 anchor.
- **Single + threshold**: a single clean-QA re-read overrides P1 only when its P(True) exceeds $\tau = 0.5$. Adds confidence gating but still removes multi-prompt aggregation and a graded anchor.

Table 7 reports the mean accuracy change, the number of configurations with non-negative Δ, and mean preservation.

Two failure modes emerge across the baselines.

(i) Removing the P1 anchor catastrophically damages preservation. CQA0, SC, and USC-lite—which override P1 unconditionally or by simple voting—all preserve only ≈ 75–78% of P1-correct samples and consequently degrade aggregate accuracy on most configurations. The damage is not confined to weak systems: the strongest systems (R1-Searcher on 2Wiki, −9.5 to −10.1pp) are the hardest hit, because they have the most P1-correct samples to lose—exactly the failure (D2)/(D3) predict and that the $\lambda = 3$ anchor is designed to prevent.

Method	Mean Δ (pp)	Pres.
CQA0 (single)	−3.40	74.5%
SC k=4 uniform	−2.31	78.2%
USC-lite (P(True))	−2.30	78.2%
Single + threshold τ=0.5	−0.52	91.4%
CWFT (λ=3, k=4)	**+0.88**	**99.4%**

Table 7: Component ablation across all 15 configurations. *Mean* Δ: average accuracy change vs. P1 in percentage points. *Pres.*: mean preservation.

(ii) Confidence gating alone is insufficient. "Single + threshold" restores most of the preservation (91.4%) but still degrades most configurations, because a single re-read cannot reliably recover format-induced failures that depend on prompt-phrasing diversity (D1). Without multi-prompt aggregation, the gate filters out wrong re-reads but does not produce enough *correct* re-reads to outweigh P1 on RM samples.

Only the full CWFT pipeline passes both tests: positive mean Δ, preservation > 99% on every configuration, and non-negative gains across the board (the −0.008pp single-sample deviation on R1-Searcher / 2Wiki notwithstanding)—exactly what (D1)–(D3) jointly predict.

Hyperparameter sensitivity. We sweep $\lambda \in \{0, 1, 2, 3, 4, 5\}$ and $k \in \{1, 2, 3, 4\}$ averaged across all configurations (App. B.3). $\lambda = 3$ sits at the preservation–gain Pareto knee: smaller λ trades preservation for aggressiveness ($\lambda = 0$: -1.80pp at 81.4% preservation), while $\lambda \geq k$ makes override numerically impossible and collapses CWFT to outputting P1. The $k = 4$ setting is the smallest value at which a non-P1 group can strictly exceed $\lambda = 3$ given $w_i \in [0, 1]$; the jump from $k = 3$ to $k = 4$ ($+0.36$pp) is the largest in the sweep, since prompt diversity (Sec. 3.4) is what unlocks the format swap.

5.3 Compute cost of cascaded inference

The cascade (Proposition 1) replaces the eager 4-CQA pipeline with a one-CQA-first procedure: in the early-exit branch, only a single re-read and one P(True) probe are issued. Averaged across the configurations, the cascade runs at $\mathbf{68.8}\%$ of eager compute ($\sim 31\%$ saving) with *exact* output match (100% on every configuration; accuracy delta ~ 0) thanks to Proposition 1's early-exit safety (per-configuration breakdown in App. B.5). Configurations where format-induced RM is sparse (s3 and AutoRefine on HotpotQA) see the largest savings (56.9–59.8% of eager compute) because more samples take the early-exit branch. The cascaded variant is the operational definition of CWFT throughout the paper.

5.4 Diagnostic alignment

A method derived from a diagnosis should produce gain patterns that match it. Sec. 3.4 identifies Search-R1 as the most format-recoverable system (48.9% recovery); the cross-system table agrees, with Search-R1 receiving the largest CWFT gains ($+3.4$pp on HotpotQA and 2Wiki, $p < 10^{-30}$). R1-Searcher, with its training-time verification step, has the smallest format-recoverable RM and correspondingly small CWFT gains. The (D3) anchor caps CWFT's deployable gain ($+0.1$–$+3.4$pp) well below the clean-vs-agent ceiling of Sec. 3.5 in exchange for preservation; Search-R1 configurations sit closest to that ceiling. Where format is causally responsible for RM, CWFT helps; otherwise, no harm.

6 Conclusion

We presented the first decomposed measurement of search vs. reading ability in RL-trained search agents. Three findings: **(1) Retrieval Myopia (RM)**: up to 46.2% of samples are answered incorrectly even when the gold answer is literally present in the retrieved documents—reading, not retrieval, is the dominant bottleneck (Sec. 3). **(2)** A confound-controlled comparison **causally isolates the agent interaction format** as a substantial driver of RM: clean wrapping beats agent wrapping by $+9.8$–20.8 **pp** ($p < 10^{-4}$; Sec. 3.5). **(3) CWFT**, a training-free pipeline derived from the diagnosis, universally improves accuracy across all 15 evaluated configurations (mean $+1.2$ **pp**; Sec. 5).

Limitations

We discuss the scope conditions of our findings.

Evaluation scope. All five evaluated systems use Qwen2.5-7B-Instruct as the backbone, and CWFT re-reads with the same model. Reading capability scales with model size, so the format-induced suppression we isolate (Sec. 3.5) may behave differently at 14B, 32B, or 70B—both in absolute magnitude and in the residual reading-capability gap that remains after the format swap. We evaluate on three English-only multi-hop QA benchmarks (HotpotQA, 2WikiMultiHopQA, MuSiQue); the reading-bottleneck phenomenon may take different forms in long-form open-domain QA, fact verification, or non-English settings. Each system is evaluated using its publicly released checkpoint.

Single-agent format isolation. The confound-controlled comparison in Sec. 3.5 is conducted on Search-R1 because it has the simplest and most well-isolated agent format. Cross-system CWFT gains (Table 6) and format-transfer recovery rates (Sec. 3.4) provide indirect evidence that the format mechanism generalises; direct controlled runs for the other agents would strengthen the causal claim.

Baselines compared. Our component ablation (Table 7) compares CWFT against inference-time variants that share its re-reading and confidence infrastructure: single re-read, self-consistency (Wang et al., 2022b), universal self-consistency (Chen et al., 2023), and confidence thresholding. The comparison is intended to identify which *components* of inference-time correction matter, not to position CWFT against the broader RAG-correction literature.

Ethics Statement

This work analyses publicly released RL search-agent checkpoints and standard multi-hop QA

benchmarks (HotpotQA, 2WikiMultiHopQA, MuSiQue); we do not collect new data, release new model weights, or involve human subjects. CWFT is a training-free inference-time procedure that adds multiple re-reading forward passes per query (partially mitigated by the cascade's early-exit, Sec. 5.3); deployments should weigh this compute overhead against the accuracy gain. As with any RAG-correction method, downstream applications inherit the biases and factual errors of the underlying retrieval corpus and backbone model; our diagnosis isolates the reading component but does not address these broader risks.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avi Sil, and Hannaneh Hajishirzi. 2024. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *International conference on learning representations*, volume 2024, pages 9112–9141.

Xinyun Chen, Renat Aksitov, Uri Alon, Jie Ren, Kefan Xiao, Pengcheng Yin, Sushant Prakash, Charles Sutton, Xuezhi Wang, and Denny Zhou. 2023. Universal self-consistency for large language model generation. *arXiv preprint arXiv:2311.17311*.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. 2024. Chain-of-verification reduces hallucination in large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3563–3578, Bangkok, Thailand. Association for Computational Linguistics.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625.

Pengcheng Jiang, Xueqiang Xu, Jiacheng Lin, Jinfeng Xiao, Zifeng Wang, Jimeng Sun, and Jiawei Han. 2025. s3: You don't need that much data to train a search agent via rl. *arXiv preprint arXiv:2505.14146*.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, and 1 others. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 6769–6781.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the association for computational linguistics*, 12:157–173.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, and 1 others. 2023. Self-refine: Iterative refinement with self-feedback. *Advances in neural information processing systems*, 36:46534–46594.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. In *International Conference on Learning Representations*, volume 2024, pages 25055–25083.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Dan Shi, Tianhao Shen, Yufei Huang, Zhigen Li, Yongqi Leng, Renren Jin, Chuang Liu, Xinwei Wu, Zishan Guo, Linhao Yu, and 1 others. 2024a. Large language model safety: A holistic survey. *arXiv preprint arXiv:2412.17686*.

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Wen-tau Yih. 2024b.

Trusting your evidence: Hallucinate less with context-aware decoding. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 783–791, Mexico City, Mexico. Association for Computational Linguistics.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Advances in neural information processing systems*, 36:8634–8652.

Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. 2025. R1-searcher: Incentivizing the search capability in llms via reinforcement learning. *arXiv preprint arXiv:2503.05592*.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. musique: Multi-hop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554.

Han Wang, Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. 2025. AdaCAD: Adaptively decoding to balance conflicts between contextual and parametric knowledge. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 11636–11652, Albuquerque, New Mexico. Association for Computational Linguistics.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022a. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022b. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Peilin Wu, Mian Zhang, Kun Wan, Wentian Zhao, Kaiyu He, Xinya Du, and Zhiyu Chen. 2025. Hiprag: hierarchical process rewards for efficient agentic retrieval augmented generation. *arXiv preprint arXiv:2510.07794*.

Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. 2024. Corrective retrieval augmented generation. *arXiv preprint arXiv:2401.15884*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2369–2380.

Shi Yaorui, Li Shihan, Wu Chang, Liu Zhiyuan, Fang Junfeng, Cai Hengxing, Zhang An, and Wang Xiang. 2025. Search and refine during think: Autonomous retrieval-augmented reasoning of llms. *arXiv preprint arXiv:2505.11277*.

A Implementation Details

This appendix collects the algorithmic specification, prompt templates, and reproducibility details that accompany the method in Sec. 4.

A.1 CWFT algorithm and early-exit proof

Proposition 1 (Early-exit safety)**.** *Fix $\lambda = 3$ and $k = 4$ with $w_i \in [0, 1]$ for grounded re-reads. If CQA_1 is grounded in D and $g(\mathrm{CQA}_1) = g(\mathrm{P1})$, then $\hat{y}^{\,\mathrm{agg}} = \mathrm{P1}$ regardless of $\mathrm{CQA}_2, \ldots, \mathrm{CQA}_k$.*

Proof. The P1 group weight is $\lambda + w_1 > 3$, while the total remaining weight is $\sum_{i=2}^{k} w_i \leq 3$. The arg-max in Eq. (1) is thus $g(\mathrm{P1})$. □

Algorithm 1 CWFT (Confidence-Weighted Format Transfer)

Require: question q; agent prediction P1; retrieved docs D; backbone model M; clean-QA prompt templates $\{P_1, \ldots, P_k\}$; anchor weight λ; normaliser g

Ensure: final answer $\hat{y}$

```
CQA_1 ← M(P_1(q, D))                    ▷ (D1) first clean-format re-read
if GROUNDED(CQA_1, D) and g(CQA_1) = g(P1) then
    return P1                           ▷ early exit (Proposition 1)
end if
votes ← { g(P1) : λ }                   ▷ (D3) anchor on the agent prediction
for i = 1, ..., k do
    if i > 1 then
        CQA_i ← M(P_i(q, D))            ▷ (D1) remaining re-reads
    end if
    if ¬ GROUNDED(CQA_i, D) then
        continue                        ▷ drop hallucinations
    end if
    w_i ← P_M(True | q, D, CQA_i)       ▷ (D2) P(True) confidence
    c ← g(CQA_i)
    votes[c] ← votes.GET(c, 0) + w_i
end for
return g^{-1}( arg max_c votes[c] )     ▷ Eq. (1)
```

A.2 Re-reading prompts

The $k = 4$ re-reading prompt templates used by CWFT are listed verbatim below; {docs} is the concatenated retrieved documents and {q} is the question.

(i) Simple QA.

```
Read the following documents carefully
and answer the question with a short,
concise answer (a few words).

Documents:
{docs}

Question: {q}
Answer:
```

(ii) Concise extractive.

```
Based on the documents below, what is
the answer to the question? Give only
the answer, nothing else.

{docs}

Question: {q}
Answer:
```

(iii) Constraint-aware.

```
Answer the following question using ONLY
the documents provided. Your answer
must be 1-3 words only.

Documents:
{docs}

Question: {q}
Answer (1-3 words):
```

(iv) Clean QA.

```
From the documents below, extract the
entity or phrase that answers the
question.

{docs}

Question: {q}
Extracted answer:
```

P(True) verification prompt. The verification prompt used to compute $P(\text{TRUE})$ for both P1 and each CQA_i is:

```
Documents:
{docs}

Question: {q}
Proposed answer: {ans}

Is this answer correct based on
the documents? Answer only 'yes' or
'no'.
```

$P(\text{TRUE})$ is the model's single-token probability mass on yes (case-insensitive, summed over surface variants).

A.3 Hyperparameters and decoding

All re-reads use temperature 0.7, top-p 0.95, max generation 48 tokens, and the model's standard stop tokens; we run them via vLLM (Kwon et al., 2023) with a fixed seed. The same decoding configuration is used for the agent re-runs in the format isolation experiment (Sec. 3.5).

A.4 Confidence and normalisation

P(True). Computed as the model's single-token probability of True when prompted with the question, the documents, and the candidate answer, following Kadavath et al. (2022).

Grounded. $\text{GROUNDED}(a, D)$ is the substring test after normalisation (lower-casing, article and punctuation removal, whitespace collapse).

EM normalisation. EM uses the same normalisation as GROUNDED with bidirectional substring matching to be robust to canonical-form variation (e.g. "Barack Obama" vs. "Obama").

A.5 Compute infrastructure

All experiments were run on NVIDIA H200 GPUs (141 GB HBM3e). Inference uses vLLM (Kwon et al., 2023) with FAISS-GPU acceleration for retrieval. The full experimental matrix—P1 inference, CQA re-reads, $P(\text{TRUE})$ probes, and cascade evaluation across $5 \times 3 = 15$ system–dataset configurations—totalled approximately 300 H200 GPU-hours, inclusive of pilot runs and ablation sweeps. At deployment, CWFT adds the equivalent of $k = 4$ re-reads plus verification probes per query; the cascade variant (Sec. 5.3) reduces this by $\approx 31\%$ relative to the eager pipeline.

B Additional Results

B.1 Cross-dataset oracle RM

Table 8 reports oracle RM on three multi-hop QA benchmarks, extending the HotpotQA results of Sec. 3.3. The reading-bottleneck pattern persists on the two harder benchmarks (2WikiMultiHopQA, MuSiQue), with oracle RM ranging from 18.2% to 46.2% across the four systems we evaluate.

B.2 Format isolation: full statistics

Table 9 reports bootstrap 95% CIs for the majority-vote results shown in Table 5, together with the Pass@K ceiling (any of K=4 samples matches gold) which complements the deployable majority-vote metric.

System	HotpotQA	2Wiki	MuSiQue
Search-R1	19.3%	24.0%	30.0%
R1-Searcher	18.2%	18.2%	30.6%
AutoRefine	25.6%	37.5%	46.2%
s3 (7B)	29.9%	—	—
Base	30.7%	—	—

Table 8: Oracle RM across three multi-hop benchmarks. "—": not measured (s3 checkpoint released only with HotpotQA training; Base oracle on harder datasets outside our RL-agent diagnostic scope).

Metric	Dataset	Δ	95% CI	p
Majority vote (deployable)				
MV	HotpotQA	$+13.6$	$[+11.2, +16.1]$	7×10^{-25}
MV	2Wiki	$+20.8$	$[+18.4, +23.1]$	3×10^{-55}
MV	MuSiQue	$+9.8$	$[+5.5, +14.7]$	8×10^{-5}
Pass@4 (ceiling)				
P@4	HotpotQA	$+11.4$	$[+9.0, +13.9]$	7×10^{-18}
P@4	2Wiki	$+17.7$	$[+15.4, +20.2]$	3×10^{-39}
P@4	MuSiQue	$+10.7$	$[+5.8, +16.0]$	9×10^{-5}

Table 9: Format-isolation full statistics for Table 5: bootstrap 95% CIs (B=1,000) on per-sample Δ. *MV*: majority vote over K=4; *Pass@4*: any-of-4 ceiling.

B.3 Anchor and aggregation breadth sweeps

Table 10 reports the full sweeps over the anchor weight λ and the number of re-reading prompts k, averaged across all evaluated system–dataset configurations. $\lambda = 3$ and $k = 4$ are the operating points used throughout the paper (Sec. 4.4).

λ	Δ (pp)	Pres.	k	Δ (pp)	Pres.
0	-1.80	81.4%	1	$+0.47$	100.0%
1	$+0.36$	93.5%	2	$+0.49$	100.0%
2	$+0.54$	95.8%	3	$+0.52$	100.0%
3	**+0.88**	**99.4%**	**4**	**+0.88**	**99.4%**
4	$+0.62$	100.0%			
5	$+0.62$	100.0%			

Table 10: λ sweep (k=4) and k sweep (λ=3), averaged across all 15 configurations. *Pres.*: mean preservation.

B.4 Per-configuration statistical significance

Table 11 reports paired McNemar tests on per-sample binary outcomes (P1-correct vs. CWFT-correct) and 95% bootstrap CIs on Δ (1000 resamples) for each evaluated CWFT configuration.

B.5 Cascade cost per configuration

Table 12 reports the cascade's early-exit rate, output-match rate against an eager $k = 4$ reference, and compute relative to the eager variant.

System	Dataset	Δ (pp)	McNemar p	95% CI
Search-R1	HotpotQA	$+3.38$	$< 10^{-30}$	$[+2.97, +3.85]$
Search-R1	2Wiki	$+3.36$	$< 10^{-30}$	$[+3.01, +3.69]$
Search-R1	MuSiQue	$+0.70$	5×10^{-3}	$[+0.25, +1.16]$
R1-Searcher	HotpotQA	$+1.30$	1.8×10^{-11}	$[+0.90, +1.67]$
R1-Searcher	2Wiki	$+0.40$	3.7×10^{-6}	$[+0.23, +0.57]$
R1-Searcher	MuSiQue	$+0.33$	0.13	$[-0.04, +0.74]$
AutoRefine	HotpotQA	$+1.08$	2.5×10^{-10}	$[+0.76, +1.42]$
AutoRefine	2Wiki	$+0.40$	0.12	$[+0.00, +0.85]$
AutoRefine	MuSiQue	$+0.45$	0.16	$[-0.10, +1.05]$
s3	HotpotQA	$+1.00$	7.1×10^{-15}	$[+0.77, +1.26]$
s3	2Wiki	$+0.67$	7.6×10^{-15}	$[+0.51, +0.83]$
s3	MuSiQue	$+0.08$	0.73	$[-0.12, +0.33]$
Base	HotpotQA	$+2.07$	2.0×10^{-28}	$[+1.70, +2.44]$
Base	2Wiki	$+0.85$	3×10^{-3}	$[+0.30, +1.35]$
Base	MuSiQue	$+1.25$	1.6×10^{-5}	$[+0.75, +1.80]$

Table 11: Per-configuration CWFT statistics. p: McNemar paired test on per-sample correctness (P1 vs. CWFT). 95% CIs: bootstrap on Δ (B=1,000).

System	Dataset	Early%	Match%	Cost	ΔAcc
Search-R1	HotpotQA	44.7%	100.0%	66.5%	0.00
Search-R1	2Wiki	35.3%	100.0%	73.5%	0.00
Search-R1	MuSiQue	34.8%	100.0%	73.9%	0.00
R1-Searcher	HotpotQA	50.7%	100.0%	61.9%	0.00
R1-Searcher	2Wiki	36.9%	100.0%	72.3%	0.00
R1-Searcher	MuSiQue	26.6%	100.0%	80.0%	0.00
AutoRefine	HotpotQA	53.6%	100.0%	59.8%	0.00
AutoRefine	2Wiki	38.5%	100.0%	71.2%	0.00
AutoRefine	MuSiQue	35.9%	100.0%	73.1%	0.00
s3	HotpotQA	57.4%	100.0%	56.9%	0.00
s3	2Wiki	41.8%	100.0%	68.6%	0.00
s3	MuSiQue	28.2%	100.0%	78.9%	0.00
Base	HotpotQA	41.0%	100.0%	69.3%	0.00
Base	2Wiki	28.6%	100.0%	78.5%	0.00
Base	MuSiQue	29.4%	100.0%	77.9%	0.00
Mean		**41.6%**	**100.0%**	**68.8%**	~ 0

Table 12: Cascade cost per configuration. *Early%*: early-exit rate. *Match%*: cascade vs. eager output agreement. *Cost*: cascade compute as a fraction of eager. Δ*Acc*: cascade accuracy $-$ eager accuracy.