COUNTING ON THINKING: TRACING EVIDENCE INTEGRATION IN LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Computational resources are finite, so no intelligent system can afford extended computation for every decision. Humans and animals handle frequent operations with automatic *System 1* processes and reserve costly *System 2* computation for problems that warrant it. Large language models (LLMs) can likewise spend extra inference-time computation on hard problems, yet their direct answers struggle even with counting, an elementary integration of input that humans and animals perform automatically. We ask why such an operation requires thinking in LLMs. Evidence integration is so basic that psychology and neuroscience have long used it to probe decision-making; our LLM version presents one letter per conversational turn and asks the model which of two target letters appeared more often. A running difference between the two counts solves the task optimally, weighting every letter equally, and fits this format exactly: the tokens at each turn can represent the difference accumulated over all earlier letters and update it with the new one. Direct responses nonetheless weighted evidence unevenly, with a strong recency effect, and assigned less probability to the correct answer as the task grew harder. Thinking improved performance and made the integration weights nearly uniform, yet final-query attention stayed concentrated on the ends of the sequence in both modes. Instead, reasoning trajectories showed the model revisiting the input, recounting the letters, and checking intermediate counts that the answer then read, suggesting that thinking constructs the accumulated count that direct responses lack rather than reading out one already formed. This recounting cost reasoning tokens that grew with the number of letters far more than with coherence. Nor did outcome feedback bring the computation into direct responses: under in-context reinforcement learning (ICRL), performance deteriorated over repeated games and the recency effect strengthened, yet the models grew more confident. Humans and animals amortize such elementary computations into automatic processes, whereas current LLMs still pay for them with thinking on every trial. Which operations learning can make directly available is a central question for how future models allocate computation.

1 INTRODUCTION

Computation is a limited resource, and every intelligent system must decide where to spend it. Humans and animals handle most decisions with fast, automatic processes and recruit slower, effortful computation only when a situation demands it. The distinction between *System 1* and *System 2* captures this division (Evans & Stanovich, 2013), and resource-rational accounts explain its logic: effortful computation pays off only when it improves decisions enough to justify its cost (Shenhav et al., 2013; Griffiths et al., 2015; Lieder & Griffiths, 2020), whereas computations required often can be amortized, so that their cost is paid largely during learning (Gershman & Goodman, 2014). Integrating incoming evidence is one such computation. Performed reliably by a fast pathway, it leaves effortful computation free for the problems that need it.

Large language models (LLMs) can likewise spend additional computation at inference time, and we treat their non-thinking and thinking modes as functional analogues of *System 1* and *System 2*. Their *System 1* has conspicuous gaps. LLMs solve complex problems (OpenAI et al., 2024; Guo et al., 2025), yet their direct answers have struggled with elementary letter counting, as in the widely noted failure to count the occurrences of "r" in "strawberry" (Fu et al., 2026). Thinking lets models solve

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.

harder problems: generating intermediate steps helps on arithmetic and symbolic tasks that models cannot answer directly (Wei et al., 2022; Nye et al., 2021; Lanchantin et al., 2023) and extends what a fixed transformer can compute (Merrill & Sabharwal, 2024; Li et al., 2024). The gain has a price: thinking can spend substantial computation on simple questions with little additional benefit (Chen et al., 2025). Counting letters that arrive one at a time is the textual analogue of integrating sensory evidence, and a single counter, updated once per turn, solves it exactly. Theoretical work shows how transformers can implement counting (Weiss et al., 2021) and examines how this ability depends on model capacity (Yehudai et al., 2024). We therefore ask to what extent this simple integration fails in direct responses, where the failure lies, and what thinking adds when every letter is already in the context. We consider two hypotheses. The construction hypothesis holds that thinking forms a count that direct responses lack, converting letters scattered across the input into an accumulated state through additional steps. The readout hypothesis instead assumes that the count already exists and that thinking improves how the answer uses it. The distinction matters because it locates the failure of direct responses either in forming the count or in using it, which determines what a fast pathway would have to learn.

Accumulating evidence over time is so basic to how humans and animals guide action that psychology and neuroscience have long studied it as a model of decision-making (Gold & Shadlen, 2007; Bogacz et al., 2006; Brunton et al., 2013). Behavioral integration kernels estimate from choices how strongly evidence at each moment influences the decision (Kiani et al., 2008; Keung et al., 2020; Xiong et al., 2025), and the optimal kernel depends on the task: discounting older evidence is adaptive in changing environments (Glaze et al., 2015), whereas counting across a whole sequence is optimal only with uniform integration weights, which weight every item equally (Appendix B.2). LLMs are known to use context unevenly, with recency bias in few-shot prompting (Zhao et al., 2021) and weaker use of the middle of long contexts (Liu et al., 2024). For counting, the kernel turns this unevenness into a direct test: any deviation from uniform weights marks a departure from the rule. The kernel is a functional description of how input evidence maps to the answer under the tested protocol, not an estimate of an internal accumulator. Simplicity is what makes this test possible: the fully specified counting rule lets us characterize how thinking changes temporal evidence weighting, beyond measuring whether it improves the final answer.

We apply this framework directly to LLMs: models observe letters one per conversational turn and choose which of two target letters occurred more often while ignoring distractors (Figure 1). Across seven models and a range of sequence lengths, coherences, and distractor densities, we compare direct answers with answers generated after thinking from an identical history, using integration kernels to measure how evidence at each position shapes the answer and final-query attention to record where the answer query attends at its last step. Because the kernel reflects the entire computation and final-query attention only its last step, we examine whether more uniform evidence use accompanies more uniform access to the letters at the answer.

Even on this task, direct responses depart from the rule in a structured way. They weight evidence unevenly: their integration weights skew toward the end of the sequence, a recency effect, and stay low in its middle, a lost-in-the-middle profile. Their correct-answer negative log-likelihood also rises on longer sequences. Thinking improves performance and makes the integration weights more uniform, approaching the equal weighting the rule requires, at a reasoning-token cost that grows with the number of letters far more than with coherence. Final-query attention in both modes concentrates on the ends of the sequence, even as thinking makes evidence use more uniform. Direct answers attend most to the latest letter yet do not reliably reflect the accumulated evidence. Selected reasoning trajectories illustrate how thinking processes the evidence. The model revisits the letters and writes out intermediate counts that the final query attends to, regenerating information already present in its input. These observations suggest that thinking constructs explicit intermediate counts.

LLMs adapt to new tasks through statistical learning in context, reorganizing their internal representations to fit the task structure (Brown et al., 2020; Wei et al., 2023; Park et al., 2025; Xiong et al., 2026b). Outcome feedback across repeated games might therefore teach the model to integrate the letters in its residual stream, making the integration kernels of direct responses progressively more uniform. We test this with in-context reinforcement learning (ICRL): models play ten games in one conversation with correctness feedback after each decision (Laskin et al., 2022; Monea et al., 2024). As games accumulate, correct-answer negative log-likelihood rises and integration kernels show a stronger recency effect, yet the models become more confident on average. The policy thus becomes more concentrated while continuing to favor recent evidence. Where humans and animals amortize

elementary computations into fast processes, current LLMs still recover counting through thinking, spending computation and context on an operation that a fast pathway could perform.

This work makes four contributions. We introduce a sequential counting protocol whose reference computation is fully specified, so that any departure from uniform integration weights is interpretable. Behavioral integration kernels characterize how thinking changes the temporal distribution of evidence weights, from the strong recency of direct responses toward the uniform profile specified by the rule. Comparing kernels with final-query attention dissociates evidence use from evidence access at the answer: thinking makes evidence use nearly uniform while measured final-query attention to the original letters remains concentrated on the sequence ends. Finally, an ICRL test shows that ten games with outcome feedback do not make direct evidence weighting more uniform.

2 METHODS

2.1 SEQUENTIAL LETTER-COUNTING TASK

We asked models to identify which of two target letters occurred more often in a sequence presented one letter per conversational turn (Figure 1). The instructions specified the two targets and asked the model to count them while ignoring all other letters, which served as distractors. Each game used a new target pair and sequence.

For a sequence $\ell_1, \ldots, \ell_T$ with targets A and B, let $x_i = +1$ if ℓ_i is A, $x_i = -1$ if it is B, and $x_i = 0$ for any other letter. A signed count then gives the answer:

$$z_0 = 0, \qquad z_i = z_{i-1} + x_i, \qquad a^\star = \mathrm{sign}(z_T), \tag{1}$$

with $a^\star = +1$ denoting A. Each occurrence of a target letter thus contributes the same unit weight, independent of its position in the sequence, so the optimal integration weights are uniform. This recurrence defines the optimal case of the task; it does not assume that the model internally implements a running count or any particular accumulation algorithm.

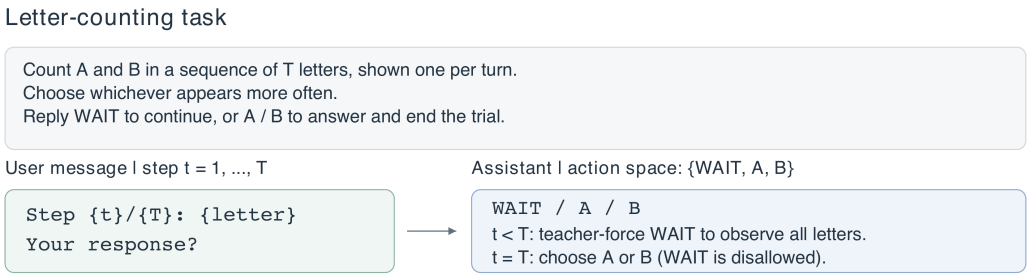

Figure 1: **The sequential letter-counting task.** A T-step trial with example target letters A and B. Letters arrive one per turn, and the model responds with `WAIT`, A, or B; at the final step, `WAIT` is disallowed. The two experiments differ in how responses are controlled. In the single-game experiment, each conversation contains one game: `WAIT` is teacher-forced at nonfinal steps so that the model observes the whole sequence, and at the final letter it answers either directly (non-thinking) or after thinking. In the ICRL experiment, each conversation contains ten non-thinking games: the model chooses its own responses, may answer before the final letter, and learns after each game only whether its answer was correct. The schematic depicts response control in the single-game experiment; ICRL games use the same message format. System instructions are shown schematically; Appendix A.3 gives the complete prompts.

2.2 EXPERIMENTAL DESIGN

The single-game experiment asked what thinking adds once all the evidence has been received. Each sequence was presented in a fresh conversation, and the response at every nonfinal step was fixed to `WAIT`, so the model observed the whole sequence. At the final letter, each trial branched into a direct answer (non-thinking) and an answer produced after a reasoning continuation (thinking), both from

an identical history. We varied one factor at a time around a baseline of 40 letters, coherence 0.10, and no distractors. Specifically, we varied sequence length (20 to 60 letters); coherence, the requested target-count difference relative to the number of target letters (0.05 to 0.20); and distractor density (0 or 0.20). Because coherence was fixed across lengths, longer sequences carried larger absolute count differences (Table 2). The repeated-game experiment asked whether experience improves direct answers. Models played ten non-thinking games of 20 letters in one conversation and could wait for further evidence or answer early. After each decision, feedback stated only whether it was correct, while earlier observations, choices, and feedback remained in context. We refer to this as the in-context reinforcement learning (ICRL) experiment (Laskin et al., 2022; Monea et al., 2024). Answers were scored by counting the two target letters across all 20 positions, in conditions with or without distractors. Complete conditions, sequence generation, and prompts are given in Appendix A.

2.3 Models

We studied seven models: Qwen3.5-4B and Qwen3.5-9B (Qwen, 2026), Gemma-4-E2B-it and Gemma-4-E4B-it (Team, 2026), Qwen3.6-35B-A3B (Qwen, 2026), Inkling-Small (Lab, 2025), and Nemotron-3.5-Lightning-30B-A3B-BF16 (NVIDIA et al., 2025). The four Qwen3.5 and Gemma-4 models provided complete action scores and form the attention cohort; the other three returned top-20 token scores and contribute to the seven-model behavioral averages. Because the Qwen3.5 and Qwen3.6 models used mode-specific sampling settings, their mode contrast also includes a change in sampling. Sample sizes, scoring, and recorded layers are specified in Appendix A.1.

2.4 Measures

We measured decision quality as the negative log-likelihood (NLL) of the correct answer, normalizing action scores over the legal actions at unit temperature. In the ICRL experiment we also report action entropy, computed from the executed sampling distribution, which measures how concentrated, and in this sense how confident, the policy is. Integration kernels regress, for each model, condition, and mode, the final score difference between the two target letters, y_t, on the signed stimulus codes of trial t,

$$y_t = b + \sum_{i=1}^{T} w_i x_{ti} + \varepsilon_t, \tag{2}$$

by ridge regression (Appendix B.2). The integration weight w_i measures how strongly a target letter at position i moves the final preference toward that letter. Kernels keep raw score scales, so their height reflects overall sensitivity to evidence and their profile its temporal distribution; under Eq. 1, the optimal profile is uniform, weighting every position equally. Final-query attention is the attention that the query producing the answer assigns to the original letter tokens. Because individual heads specialize in function (Olsson et al., 2022), we used stimulus-selective heads chosen on separate conversation blocks and shared across single-game modes, normalizing their attention over the letters (Appendix B.4). Reasoning length is the number of generated reasoning tokens. Behavioral measures average all seven models and attention measures the four complete-score models. Single-game kernels use trials eligible in both modes; ICRL kernels use only games that observed all 20 letters, whereas ICRL NLL and action entropy also include early answers (Appendix B).

3 Results

3.1 Thinking makes the uneven integration weights of direct responses nearly uniform

Direct responses performed poorly on a task that only asks which of two letters appeared more often. Without thinking, the seven-model mean of correct-answer NLL stayed near the level of an uninformed guess and exceeded it in all but the two highest coherence conditions (Figure 2a–c; error bars show model SEM). The task became harder as sequences grew longer, even though the count differences were larger. Lower coherence also made the task harder. NLL reflected both effects: it rose with sequence length and fell as coherence increased. Adding distractors did not increase NLL.

Integration kernels locate the departure from the counting rule, which weights every position equally. Non-thinking integration weights were instead uneven, with a small initial peak (primacy), little

weight through most of the sequence, and a strong skew toward its end (recency; Figure 2d; other conditions in Appendix Figure 7). Direct answers thus depended disproportionately on the final letters.

Thinking lowered correct-answer NLL at every length and coherence level (Figure 2a,c) and changed how evidence reached the answer. Thinking kernels were nearly uniform across positions, with the recency effect of direct responses largely gone, and larger overall (Figure 2d; individual-model kernels in Appendix Figure 6). Because kernel height scales with the range of the action scores, the more decisive thinking answers contribute to this increase, so the comparison rests on the profile rather than on its height. Because both modes branched from the same sequence and history, this change arose in what the model did between the last letter and its answer.

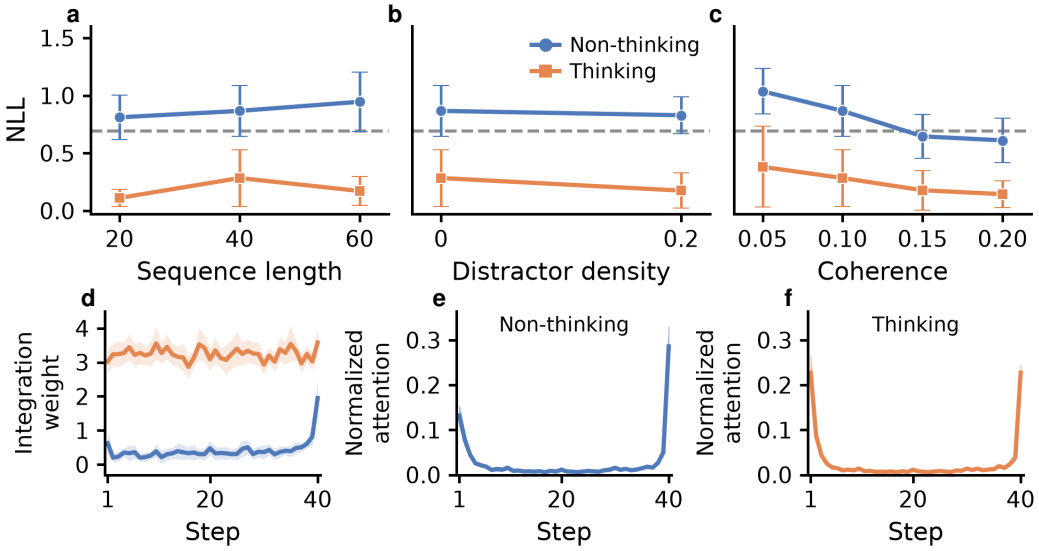

Figure 2: **Thinking lowers correct-answer NLL and makes integration weights nearly uniform, but final-query attention stays concentrated on the sequence ends.** (a–c) Correct-answer NLL by sequence length, distractor density, and coherence; dashed line: uninformed guess ($\ln 2$). (d) Integration kernels, fitted on the same trials in both modes. (e,f) Final-query stimulus attention from stimulus-selective heads in non-thinking (e) and thinking (f) modes, normalized over the original letters within each trial so that each curve sums to one; its scale is not comparable to the kernel in d. Panels d–f use 40 letters, coherence 0.10, and no distractors. Panels a–d average seven models and e,f the four complete-score models. Blue: non-thinking; orange: thinking. Means $\pm$ model SEM ($n = 7$ models in a–d; $n = 4$ in e,f). Estimator details in Appendices B.2 and B.4.

3.2 Thinking revisits the evidence and generates intermediate counts

The simplest account of what thinking adds is more even evidence access, with the final answer attending evenly to the original letters. In direct responses, final-query attention peaked weakly at the first letter and strongly at the last (Figure 2e), a profile whose ends match the primacy and recency of the non-thinking kernel, although the first-position peak may partly reflect the general tendency of LLMs to attend to early tokens (Barbero et al., 2025). The direct answer thus attended most to the latest letter, whose turn could already have held the signed count of Eq. 1 accumulated over earlier turns, yet had correct-answer NLL near the uniform-guess reference. Thinking left a similar pattern, with the same stimulus-selective heads: attention still concentrated on the first and last letters and assigned little to the middle (Figure 2f; other lengths in Appendix Figure 8), even though the behavioral weights had become more uniform. More uniform evidence use thus accompanied persistently uneven final-query attention to the original letters. This rules out more even access to the letters at the answer, but not readout, because reading out a count formed during observation would also draw attention to the latest letter; the reasoning trajectories bear on that distinction.

We selected several 20-letter trials to illustrate how thinking processes the evidence; most other trials we inspected looked similar (Appendix B.5). In a 20-letter Gemma-4-E2B-it trial, attention

during thinking moved through successive letter positions as the model counted, then swept the sequence again as it rechecked (Figure 3c). A Qwen3.5-4B trial showed the same progression without a rechecking phase (Figure 3a). In both trials, the final answer query directed part of its attention to the written counts and conclusions (Figure 3b,d). Trials from Qwen3.5-9B and Gemma-4-E4B-it show the same progression (Appendix Figure 10).

Together with the failure of direct answers that attend most to the latest letter, these trajectories suggest that thinking constructs explicit intermediate counts for the answer to consult. The kernel and final-query attention can diverge because the kernel reflects the entire computation, whereas final-query attention reflects only its last step. In these trajectories, thinking condenses evidence spread across the sequence into intermediate counts, and the final query attends to those counts. The model thus uses generated text to recreate information already present in its input.

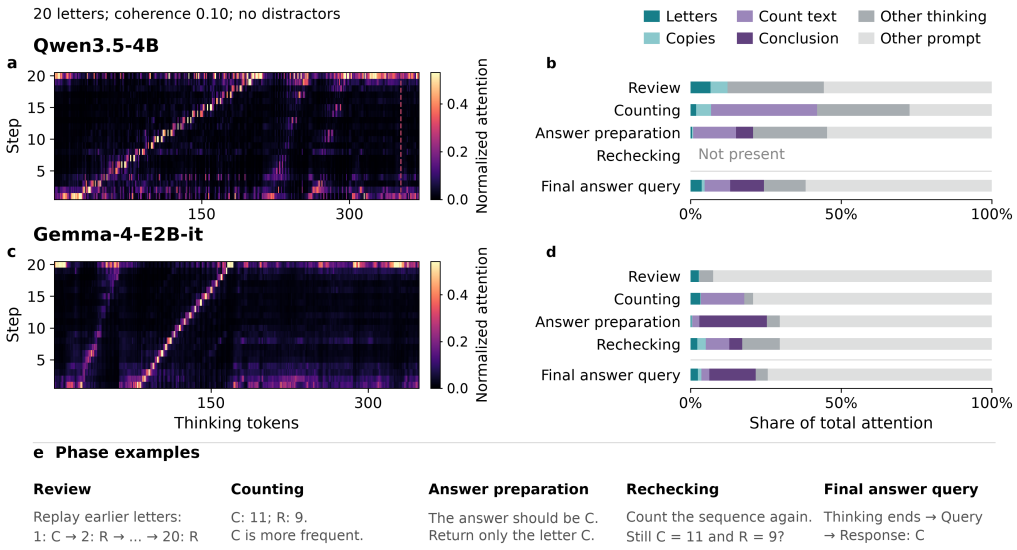

Figure 3: **Selected thinking trajectories revisit the evidence and attend to written counts at the final answer.** (a,c) Attention from each thinking token to the original letters for Qwen3.5-4B and Gemma-4-E2B-it, normalized within each column; dashed line: first expressed preference, the first statement in the generated text that names the eventual answer. (b,d) Trajectories are divided from the generated text into review, counting, answer preparation, and rechecking, with the final answer query summarized separately; bars show absolute shares of total attention to the original letters, letters copied into the reasoning, count text, conclusions, and other context. (e) Condensed example text for each phase. Both trials use 20 letters, coherence 0.10, and no distractors, with the stimulus-selective heads of Figure 2e,f (Appendix B.5).

This recounting has a computation cost that grows with the number of letters. Reasoning length grew markedly with sequence length in every model (Figure 4a), shrank more weakly as coherence rose (Figure 4c), and changed little with distractors (Figure 4b). The cost thus tracks the number of items to revisit, with a weaker adjustment to the difficulty of the decision. Within the dual-process analogy, this is the price of performing counting in *System 2*: every reasoning token is a sequential generation step, and the cost is paid anew on every trial.

3.3 In-context reinforcement learning (ICRL) does not teach direct responses to count; they grow more confident but worse

Because LLMs learn task structure statistically in context, including from outcome feedback alone (Monea et al., 2024; Xiong et al., 2026b), repeated games might teach direct responses to integrate the letters; correct-answer NLL would then fall and integration weights would grow more uniform across games. Instead, correct-answer NLL rose across games (Figure 5a) while action entropy fell

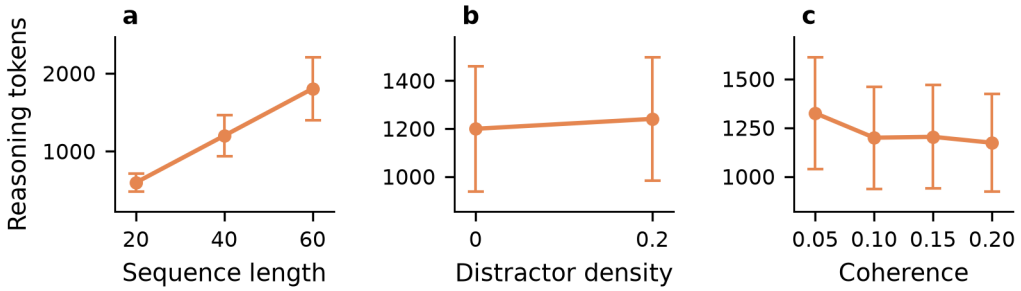

Figure 4: **Reasoning length scales with the number of letters more than with coherence.** Reasoning tokens in thinking trials, excluding answer tokens, by (a) sequence length, (b) distractor density, and (c) coherence over the tested ranges, with the other factors at the baseline. Points show equal-weight means of within-model trial means across seven models; error bars show model SEM ($n = 7$ models). Y-axis ranges differ across panels. Trials reaching the token budget are retained.

(Figure 5b). As interaction history accumulated, the models thus grew more confident while assigning less probability to the correct answer.

Evidence weighting did not move toward the counting rule as games progressed. In both initial games (Games 1–3) and final games (Games 8–10), kernels showed the recency effect of single-game direct responses of the same length, with weights low across most of the sequence and skewed toward its end (Figure 5c; Appendix Figure 7a), and the final-letter weight grew in final games (Holm-adjusted $p = 0.002$; Appendix B).

Final-query attention supported the kernel finding of a stronger recency effect in final games: the newest letter received a larger share of stimulus attention (Figure 5d). At the same time, as the conversation grew over games, a smaller fraction of final-query attention reached current-game letters (Figure 5f), with a lower fraction in final than initial games (Appendix Figure 9). Attention entropy over the full context also fell (Figure 5e), indicating more concentrated attention. The final query thus assigned less attention to the current evidence and, within it, more to the newest letter. Across ICRL games, the recency effect strengthened and integration weights remained uneven.

4 Discussion

4.1 Simple rules that direct responses do not execute

Thinking re-executes in text a computation that direct responses do not reliably perform on evidence already in context. Because the counting rule fixes what each letter should contribute to the answer, the kernel shows which letters actually did, and in direct responses the weights skewed toward the end of the sequence. This recency effect parallels those in human free recall (Greene, 1986), recency bias in few-shot prompting (Zhao et al., 2021), and the position-dependent use of long contexts (Liu et al., 2024). Scratchpads and chain-of-thought improve performance on arithmetic and symbolic problems by making intermediate computation explicit (Wei et al., 2022; Nye et al., 2021). Our task specifies the contribution of every target occurrence, allowing us to diagnose how thinking changes temporal evidence weighting relative to the exact counting rule. This connects the benefit of intermediate computation to the gap between available information and its effective use, which also arises in in-context learning and working memory (Xiong et al., 2026b;a). Nor is the gap one of capacity: transformers can express counting (Weiss et al., 2021; Yehudai et al., 2024), and interpretability work has found a production model tracking a running character count internally to predict line breaks (Gurnee et al., 2025). The relevant boundary thus separates computations that a trained model performs reliably in its direct responses from those it achieves only by unfolding additional steps, and counting, despite its simplicity, falls on the second side.

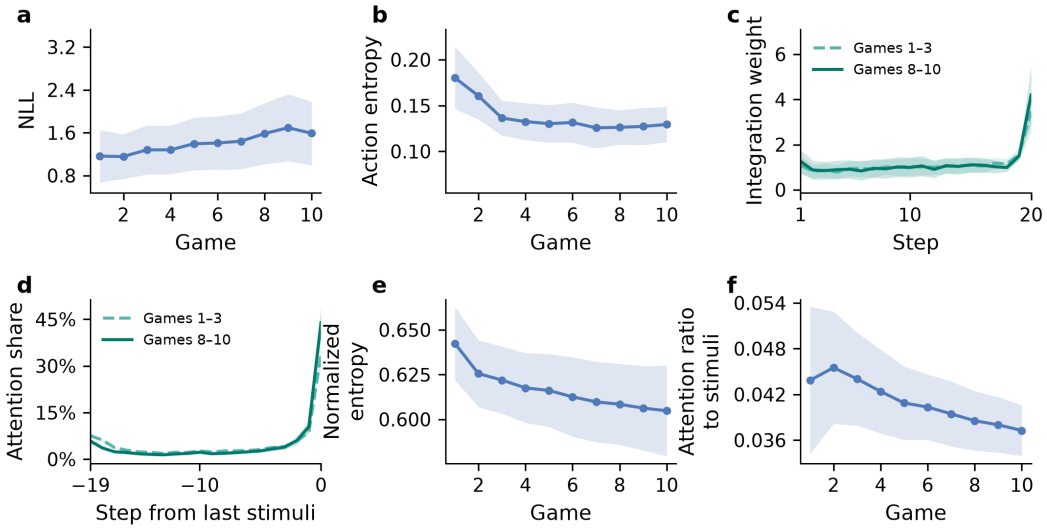

Figure 5: **ICRL does not improve direct evidence integration.** (a) Correct-answer NLL and (b) action entropy by game; (c) integration kernels for initial games (Games 1–3) and final games (Games 8–10). (d) Final-query attention share by lag from the last observed letter (0 = newest; negative values are earlier letters); (e) normalized attention entropy over the full context; (f) fraction of final-query attention on current-game letters. All panels average the two ICRL conditions, with and without distractors. Panels a–c average seven models and d–f the four complete-score models; d,e use stimulus-selective heads and f all heads in the recorded layers. Dashed light teal: initial games; solid dark teal: final games. Bands show model SEM ($n = 7$ models in a–c; $n = 4$ in d–f). Statistical tests in Appendices B and B.4.

4.2 Thinking changes what the final answer reads

We asked what thinking adds when every letter is already in the context. The evidence favors construction. Thinking made evidence use nearly uniform while the answer's attention to the original letters stayed concentrated on the ends of the sequence. In the selected trajectories, the generated text held counts built by revisiting the input: models constructed counts, compared alternatives, and sometimes checked intermediate conclusions before the final query attended to them. The ordered sweep through the letters resembles forward replay, the reactivation of recent experience in its original order that supports planning in the hippocampus (Schuck & Niv, 2019; Momennejad et al., 2018; Mattar & Lengyel, 2022). Unlike neural replay, this sweep leaves a written trace: the generated text serves as a working record that makes accumulated evidence available to subsequent processing, as scratchpads and self-notes do for other algorithmic tasks (Nye et al., 2021; Lanchantin et al., 2023), and the final query attends to a compact summary of earlier evidence. Two interventions, in the spirit of causal methods in interpretability (Geiger et al., 2025; Lindsey et al., 2025), can test this account. Supplying accurate intermediate counts alongside the original sequence, against length-matched uninformative text, would measure how much of the benefit lies in converting distributed evidence into an accessible summary; perturbing the generated counts, or disrupting attention to them, would test whether these intermediate states causally shape NLL and the kernel.

4.3 Implications for amortization and computation cost

Resource-rational accounts treat additional computation as worthwhile only when the expected improvement in decision quality outweighs its cost in time and resources (Shenhav et al., 2013; Griffiths et al., 2015; Lieder & Griffiths, 2020). By that standard, a simple and familiar operation such as counting should be performed reliably with little additional computation. Instead, thinking recovered the operation at a computation cost that grew with the number of letters (Figure 4) and is paid anew on every trial, consistent with the broader observation that scaling thinking can spend substantial computation on simple questions (Chen et al., 2025). Amortization describes the alternative: computations that are required frequently can be learned so that their cost is paid largely

during learning rather than on every decision (Gershman & Goodman, 2014). In the brain, replay also serves this function, consolidating offline computation into fast, direct processing (McClelland et al., 1995; Barry & Love, 2022); the forward replay performed during thinking leaves the model's weights unchanged, so each trial must repeat the recount. A reliable *System 1* matters for exactly this reason. Performing frequently repeated operations directly saves reasoning tokens and thinking time, leaving extended reasoning for cases where it offers sufficient benefit. If thinking helps by re-encoding evidence into an accessible summary rather than by rereading the input, latent or compressed reasoning should be judged by whether it preserves that summary, and distillation by whether it transfers the construction into the direct pathway rather than only the answers. Experience did not supply such a pathway: across repeated games, ICRL made direct responses more confident but worse, sharpening a policy that still weighted the latest letters most. Systems that accumulate outcome feedback in context may likewise become more decisive without becoming more accurate. Whether an operation that currently requires unfolding can become directly available through learning is the amortization question proper. Answering it requires manipulating learning itself, for example by training or distilling a direct integration policy (Yu et al., 2024; Deng et al., 2024) and comparing, before and after training, direct performance, temporal kernels, the reasoning required to reach a given performance, and generalization to unseen sequences and lengths. Compact working states that update accumulated evidence, retrieval mechanisms that preserve relevant observations, and latent or recurrent computation (Geiping et al., 2025) are candidate implementations, to be evaluated jointly on decision quality, temporal evidence weighting, token use, and measured response time.

4.4 Scope and limitations

Letters arrived one per turn and the single-game experiment teacher-forced `WAIT` at nonfinal steps, so the task probes integration across conversational turns; a single-prompt format could yield a different profile. The mode comparison is empirical. Because Qwen3.5 spans two sampling configurations and cross-model summaries pool complete and top-20 scores, these results describe the tested cohort rather than full-model probabilities. The tested models use at most 12B active parameters per token (Table 1), so whether frontier-scale models show the same direct-response gap is untested. Whether the direct-response gap originates in pretraining, post-training, or architecture, for instance in position encodings that index tokens rather than selected content (Golovneva et al., 2024), and where the boundary between direct and unfolded computation lies for other operations, remains open. A kernel with a recency effect shows a functional temporal bias whose source in input selection, retention, or readout remains open. The intermediate-count account rests on descriptive measurements: attention weights do not by themselves establish causal influence (Jain & Wallace, 2019), the trajectories were selected for illustration and do not estimate how often the process occurs, and the readout account, in which thinking improves access to a count already formed during observation, is not excluded, because we did not test whether the hidden states of direct responses carry a running count across turns; linear probes at each turn and the interventions proposed above would separate these alternatives. Reasoning tokens count sequential generation steps, a measure distinct from FLOPs or latency. In ICRL, NLL and action entropy come from different distributions, so "more confident" refers to a more concentrated sampling policy rather than to calibration; task experience, feedback, and context growth also varied together. Because feedback reported only correctness, the experiment tests experience with outcome feedback and leaves open whether the computation performed during thinking can be internalized. Finally, we ran no matched human experiment; the cognitive literature supplies diagnostic methods and candidate processes for this task, with no human benchmark.

4.5 Conclusion

Direct responses fail at a computation as simple as deciding which of two letters appeared more often: their integration weights are uneven, skewed toward the end of the sequence and low in its middle, and they do worse as the task becomes harder. Thinking makes evidence weighting nearly uniform while final-query attention to the original letters stays concentrated on the ends of the sequence; in selected trajectories, the model recounts the letters in text and the answer reads those counts, at a cost that grows with the number of letters. ICRL across repeated games makes direct responses more confident but not better. Humans and animals amortize such elementary computations into fast processes, whereas current LLMs pay for them with thinking. These findings pose a clear challenge for future models: determining which elementary computations can be amortized through learning, sparing effortful thinking for problems that need it.

AI USE STATEMENT

We used generative AI tools to polish the manuscript and assist with writing code for the task and data analysis. We did not use generative AI tools for any other aspect of this work. The authors reviewed all AI-assisted work and checked the code to verify that it performed the intended analyses correctly. The authors take full responsibility for the final content of this work, including all AI-assisted text, claims, and artifacts.

REPRODUCIBILITY STATEMENT

We document the experimental procedures in Section 2 and Appendix A, including model checkpoints, experimental conditions, sample sizes, stimulus generation, complete prompts, and generation settings. Appendix B specifies the behavioral measures, score-coverage and trial-eligibility rules, integration-kernel estimation, and bootstrap procedures, including resampling seeds and multiple-comparison corrections. Appendices B.4 and B.5 describe attention recording, stimulus-selective head selection, normalization, and the analysis of illustrative thinking trajectories. Appendix B.6 documents deterministic seed derivation, trial matching across models and modes, and dataset validation.

REFERENCES

Federico Barbero, Álvaro Arroyo, Xiangming Gu, Christos Perivolaropoulos, Michael Bronstein, Petar Veličković, and Razvan Pascanu. Why do LLMs attend to the first token?, May 2025. URL http://arxiv.org/abs/2504.02732. arXiv:2504.02732 [cs].

Daniel N Barry and Bradley C Love. A neural network account of memory replay and knowledge consolidation. *Cerebral Cortex*, pp. bhac054, February 2022. ISSN 1047-3211, 1460-2199. doi: 10.1093/cercor/bhac054. URL https://academic.oup.com/cercor/advance-article/doi/10.1093/cercor/bhac054/6537049.

Rafal Bogacz, Eric Brown, Jeff Moehlis, Philip Holmes, and Jonathan D. Cohen. The physics of optimal decision making: A formal analysis of models of performance in two-alternative forced-choice tasks. *Psychological Review*, 113(4):700–765, 2006. ISSN 1939-1471, 0033-295X. doi: 10.1037/0033-295X.113.4.700. URL https://doi.apa.org/doi/10.1037/0033-295X.113.4.700.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html?utm_source=transaction&utm_medium=email&utm_campaign=linkedin_newsletter.

Bingni W. Brunton, Matthew M. Botvinick, and Carlos D. Brody. Rats and Humans Can Optimally Accumulate Evidence for Decision-Making. *Science*, 340(6128):95–98, April 2013. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.1233912. URL https://www.science.org/doi/10.1126/science.1233912.

Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, Rui Wang, Zhaopeng Tu, Haitao Mi, and Dong Yu. Do NOT Think That Much for 2+3=? On the Overthinking of o1-Like LLMs, February 2025. URL http://arxiv.org/abs/2412.21187. arXiv:2412.21187 [cs.CL].

Yuntian Deng, Yejin Choi, and Stuart Shieber. From Explicit CoT to Implicit CoT: Learning to Internalize CoT Step by Step, May 2024. URL http://arxiv.org/abs/2405.14838. arXiv:2405.14838 [cs.CL].

Jonathan St. B. T. Evans and Keith E. Stanovich. Dual-Process Theories of Higher Cognition: Advancing the Debate. *Perspectives on Psychological Science*, 8(3):223–241, May 2013. ISSN 1745-6916. doi: 10.1177/1745691612460685. URL https://doi.org/10.1177/1745691612460685.

Tairan Fu, Raquel Ferrando, Javier Conde, Carlos Arriaga, and Pedro Reviriego. When Do Large Language Models (LLMs) Struggle to Count Letters? *ACM Transactions on Intelligent Systems and Technology*, 17(5):117:1–117:17, July 2026. ISSN 2157-6904. doi: 10.1145/3818606. URL https://dl.acm.org/doi/10.1145/3818606.

Atticus Geiger, Duligur Ibeling, Amir Zur, Maheep Chaudhary, Sonakshi Chauhan, Jing Huang, Aryaman Arora, Zhengxuan Wu, Noah Goodman, Christopher Potts, and Thomas Icard. Causal Abstraction: A Theoretical Foundation for Mechanistic Interpretability. *Journal of Machine Learning Research*, 26(83):1–64, 2025. ISSN 1533-7928. URL http://jmlr.org/papers/v26/23-0058.html.

Jonas Geiping, Sean McLeish, Neel Jain, John Kirchenbauer, Siddharth Singh, Brian R. Bartoldson, Bhavya Kailkhura, Abhinav Bhatele, and Tom Goldstein. Scaling up Test-Time Compute with Latent Reasoning: A Recurrent Depth Approach, February 2025. URL http://arxiv.org/abs/2502.05171. arXiv:2502.05171 [cs].

Samuel Gershman and Noah Goodman. Amortized Inference in Probabilistic Reasoning. *Proceedings of the Annual Meeting of the Cognitive Science Society*, 36(36), 2014. URL https://escholarship.org/uc/item/34j1h7k5.

Christopher M Glaze, Joseph W Kable, and Joshua I Gold. Normative evidence accumulation in unpredictable environments. *eLife*, 4:e08825, August 2015. ISSN 2050-084X. doi: 10.7554/eLife.08825. URL https://doi.org/10.7554/eLife.08825.

Joshua I. Gold and Michael N. Shadlen. The Neural Basis of Decision Making. *Annual Review of Neuroscience*, 30(1):535–574, July 2007. ISSN 0147-006X, 1545-4126. doi: 10.1146/annurev.neuro.29.051605.113038. URL https://www.annualreviews.org/doi/10.1146/annurev.neuro.29.051605.113038.

Olga Golovneva, Tianlu Wang, Jason Weston, and Sainbayar Sukhbaatar. Contextual Position Encoding: Learning to Count What's Important, May 2024. URL http://arxiv.org/abs/2405.18719. arXiv:2405.18719 [cs] version: 2.

Robert L. Greene. Sources of recency effects in free recall. *Psychological Bulletin*, 99(2):221–228, 1986. ISSN 1939-1455. doi: 10.1037/0033-2909.99.2.221.

Thomas L. Griffiths, Falk Lieder, and Noah D. Goodman. Rational Use of Cognitive Resources: Levels of Analysis Between the Computational and the Algorithmic. *Topics in Cognitive Science*, 7(2):217–229, 2015. ISSN 1756-8765. doi: 10.1111/tops.12142. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/tops.12142.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, September 2025. ISSN 1476-4687. doi: 10.1038/s41586-025-09422-z. URL https://www.nature.com/articles/s41586-025-09422-z.

Wes Gurnee, Emmanuel Ameisen, Isaac Kauvar, Julius Tarng, Adam Pearce, Chris Olah, and Joshua Batson. When Models Manipulate Manifolds: The Geometry of a Counting Task. *Transformer Circuits Thread*, 2025. URL https://transformer-circuits.pub/2025/linebreaks/index.html.

Sarthak Jain and Byron C. Wallace. Attention is not Explanation. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter*

of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 3543–3556, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1357. URL https://aclanthology.org/N19-1357/.

Waitsang Keung, Todd A. Hagen, and Robert C. Wilson. A divisive model of evidence accumulation explains uneven weighting of evidence over time. *Nature Communications*, 11(1):2160, May 2020. ISSN 2041-1723. doi: 10.1038/s41467-020-15630-0. URL https://www.nature.com/articles/s41467-020-15630-0. Number: 1.

Roozbeh Kiani, Timothy D. Hanks, and Michael N. Shadlen. Bounded Integration in Parietal Cortex Underlies Decisions Even When Viewing Duration Is Dictated by the Environment. *Journal of Neuroscience*, 28(12):3017–3029, March 2008. ISSN 0270-6474, 1529-2401. doi: 10.1523/JNEUROSCI.4761-07.2008. URL https://www.jneurosci.org/content/28/12/3017.

Thinking Machines Lab. Inkling: Our Open-Weights Model, July 2025. URL https://thinkingmachines.ai/news/introducing-inkling/.

Jack Lanchantin, Shubham Toshniwal, Jason Weston, arthur szlam, and Sainbayar Sukhbaatar. Learning to Reason and Memorize with Self-Notes. In *Advances in Neural Information Processing Systems*, volume 36, pp. 11891–11911. Curran Associates, Inc., 2023. doi: 10.52202/075280-0520. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/274d0146144643ee2459a602123c60ff-Abstract-Conference.html.

Michael Laskin, Luyu Wang, Junhyuk Oh, Emilio Parisotto, Stephen Spencer, Richie Steigerwald, D. J. Strouse, Steven Stenberg Hansen, Angelos Filos, Ethan Brooks, Maxime Gazeau, Himanshu Sahni, Satinder Singh, and Volodymyr Mnih. In-context Reinforcement Learning with Algorithm Distillation. September 2022. URL https://openreview.net/forum?id=hy0a5MMPUv.

Zhiyuan Li, Hong Liu, Denny Zhou, and Tengyu Ma. Chain of Thought Empowers Transformers to Solve Inherently Serial Problems, September 2024. URL http://arxiv.org/abs/2402.12875. arXiv:2402.12875 [cs].

Falk Lieder and Thomas L. Griffiths. Resource-rational analysis: Understanding human cognition as the optimal use of limited computational resources. *Behavioral and Brain Sciences*, 43:e1, 2020. ISSN 0140-525X, 1469-1825. doi: 10.1017/S0140525X1900061X. URL https://www.cambridge.org/core/product/identifier/S0140525X1900061X/type/journal_article.

Jack Lindsey, Wes Gurnee, Emmanuel Ameisen, Brian Chen, Adam Pearce, Nicholas L. Turner, Craig Citro, David Abrahams, Shan Carter, Basil Hosmer, Jonathan Marcus, Michael Sklar, Adly Templeton, Trenton Bricken, Callum McDougall, Hoagy Cunningham, Thomas Henighan, Adam Jermyn, Andy Jones, Andrew Persic, Zhenyi Qi, T. Ben Thompson, Sam Zimmerman, Kelley Rivoire, Thomas Conerly, Chris Olah, and Joshua Batson*‡ Affiliations Anthropic Published March 27. On the Biology of a Large Language Model, 2025. URL https://transformer-circuits.pub/2025/attribution-graphs/biology.html.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9/.

Marcelo G. Mattar and Máté Lengyel. Planning in the brain. *Neuron*, 110(6):914–934, March 2022. ISSN 08966273. doi: 10.1016/j.neuron.2021.12.018. URL https://linkinghub.elsevier.com/retrieve/pii/S0896627321010357.

James L. McClelland, Bruce L. McNaughton, and Randall C. O'Reilly. Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, 102(3):419–457, 1995. ISSN 1939-1471. doi: 10.1037/0033-295X.102.3.419. Place: US.

William Merrill and Ashish Sabharwal. The Expressive Power of Transformers with Chain of Thought, April 2024. URL http://arxiv.org/abs/2310.07923. arXiv:2310.07923 [cs.LG].

Ida Momennejad, A Ross Otto, Nathaniel D Daw, and Kenneth A Norman. Offline replay supports planning in human reinforcement learning. *eLife*, 7:e32548, December 2018. ISSN 2050-084X. doi: 10.7554/eLife.32548. URL https://doi.org/10.7554/eLife.32548.

Giovanni Monea, Antoine Bosselut, Kianté Brantley, and Yoav Artzi. LLMs Are In-Context Reinforcement Learners, October 2024. URL http://arxiv.org/abs/2410.05362. arXiv:2410.05362 [cs].

NVIDIA, Aaron Blakeman, Aaron Grattafiori, Aarti Basant, Abhibha Gupta, Abhinav Khattar, Adi Renduchintala, Aditya Vavre, Akanksha Shukla, Akhiad Bercovich, Aleksander Ficek, Aleksandr Shaposhnikov, Alex Kondratenko, Alexander Bukharin, Alexandre Milesi, Ali Taghibakhshi, Alisa Liu, Amelia Barton, Ameya Sunil Mahabaleshwarkar, Amir Klein, Amit Zuker, Amnon Geifman, Amy Shen, Anahita Bhiwandiwalla, Andrew Tao, Anjulie Agrusa, Ankur Verma, Ann Guan, Anubhav Mandarwal, Arham Mehta, Ashwath Aithal, Ashwin Poojary, Asif Ahamed, Asit Mishra, Asma Kuriparambil Thekkumpate, Ayush Dattagupta, Banghua Zhu, Bardiya Sadeghi, Barnaby Simkin, Ben Lanir, Benedikt Schifferer, Besmira Nushi, Bilal Kartal, Bita Darvish Rouhani, Boris Ginsburg, Brandon Norick, Brandon Soubasis, Branislav Kisacanin, Brian Yu, Bryan Catanzaro, Carlo del Mundo, Chantal Hwang, Charles Wang, Cheng-Ping Hsieh, Chenghao Zhang, Chenhan Yu, Chetan Mungekar, Chintan Patel, Chris Alexiuk, Christopher Parisien, Collin Neale, Cyril Meurillon, Damon Mosk-Aoyama, Dan Su, Dane Corneil, Daniel Afrimi, Daniel Lo, Daniel Rohrer, Daniel Serebrenik, Daria Gitman, Daria Levy, Darko Stosic, David Mosallanezhad, Deepak Narayanan, Dhruv Nathawani, Dima Rekesh, Dina Yared, Divyanshu Kakwani, Dong Ahn, Duncan Riach, Dusan Stosic, Edgar Minasyan, Edward Lin, Eileen Long, Eileen Peters Long, Elad Segal, Elena Lantz, Ellie Evans, Elliott Ning, Eric Chung, Eric Harper, Eric Tramel, Erick Galinkin, Erik Pounds, Evan Briones, Evelina Bakhturina, Evgeny Tsykunov, Faisal Ladhak, Fay Wang, Fei Jia, Felipe Soares, Feng Chen, Ferenc Galko, Frank Sun, Frankie Siino, Gal Hubara Agam, Ganesh Ajjanagadde, Gantavya Bhatt, Gargi Prasad, George Armstrong, Gerald Shen, Gorkem Batmaz, Grigor Nalbandyan, Haifeng Qian, Harsh Sharma, Hayley Ross, Helen Ngo, Herbert Hum, Herman Sahota, Hexin Wang, Himanshu Soni, Hiren Upadhyay, Huizi Mao, Huy C. Nguyen, Huy Q. Nguyen, Iain Cunningham, Ido Galil, Ido Shahaf, Igor Gitman, Ilya Loshchilov, Itamar Schen, Itay Levy, Ivan Moshkov, Izik Golan, Izzy Putterman, Jan Kautz, Jane Polak Scowcroft, Jared Casper, Jatin Mitra, Jeffrey Glick, Jenny Chen, Jesse Oliver, Jian Zhang, Jiaqi Zeng, Jie Lou, Jimmy Zhang, Jinhang Choi, Jining Huang, Joey Conway, Joey Guman, John Kamalu, Johnny Greco, Jonathan Cohen, Joseph Jennings, Joyjit Daw, Julien Veron Vialard, Junkeun Yi, Jupinder Parmar, Kai Xu, Kan Zhu, Kari Briski, Katherine Cheung, Katherine Luna, Keith Wyss, Keshav Santhanam, Kevin Shih, Kezhi Kong, Khushi Bhardwaj, Kirthi Shankar, Krishna C. Puvvada, Krzysztof Pawelec, Kumar Anik, Lawrence McAfee, Laya Sleiman, Leon Derczynski, Li Ding, Lizzie Wei, Lucas Liebenwein, Luis Vega, Maanu Grover, Maarten Van Segbroeck, Maer Rodrigues de Melo, Mahdi Nazemi, Makesh Narsimhan Sreedhar, Manoj Kilaru, Maor Ashkenazi, Marc Romeijn, Marcin Chochowski, Mark Cai, Markus Kliegl, Maryam Moosaei, Matt Kulka, Matvei Novikov, Mehrzad Samadi, Melissa Corpuz, Mengru Wang, Meredith Price, Michael Andersch, Michael Boone, Michael Evans, Miguel Martinez, Mikail Khona, Mike Chrzanowski, Minseok Lee, Mohammad Dabbah, Mohammad Shoeybi, Mostofa Patwary, Nabin Mulepati, Najeeb Nabwani, Natalie Hereth, Nave Assaf, Negar Habibi, Neta Zmora, Netanel Haber, Nicola Sessions, Nidhi Bhatia, Nikhil Jukar, Nikki Pope, Nikolai Ludwig, Nima Tajbakhsh, Nir Ailon, Nirmal Juluru, Nishant Sharma, Oleksii Hrinchuk, Oleksii Kuchaiev, Olivier Delalleau, Oluwatobi Olabiyi, Omer Ullman Argov, Omri Puny, Oren Tropp, Ouye Xie, Parth Chadha, Pasha Shamis, Paul Gibbons, Pavlo Molchanov, Pawel Morkisz, Peter Dykas, Peter Jin, Pinky Xu, Piotr Januszewski, Pranav Prashant Thombre, Prasoon Varshney, Pritam Gundecha, Przemek Tredak, Qing Miao, Qiyu Wan, Rabeeh Karimi Mahabadi, Rachit Garg, Ran El-Yaniv, Ran Zilberstein, Rasoul Shafipour, Rich Harang, Rick Izzo, Rima Shahbazyan, Rishabh Garg, Ritika Borkar, Ritu Gala, Riyad Islam, Robert Hesse, Roger Waleffe, Rohit Watve, Roi Koren, Ruoxi Zhang, Russell Hewett, Russell J. Hewett, Ryan Prenger, Ryan Timbrook, Sadegh Mahdavi, Sahil Modi, Samuel Kriman, Sangkug Lim, Sanjay Kariyappa, Sanjeev Satheesh, Saori Kaji, Satish Pasumarthi, Saurav Muralidharan, Sean Narentharen, Sean Narenthiran, Seonmyeong Bak, Sergey Kashirsky, Seth Poulos, Shahar Mor, Shanmugam Ramasamy, Shantanu Acharya, Shaona Ghosh, Sharath Turuvekere Sreenivas, Shelby

Thomas, Shiqing Fan, Shreya Gopal, Shrimai Prabhumoye, Shubham Pachori, Shubham Toshniwal, Shuoyang Ding, Siddharth Singh, Simeng Sun, Smita Ithape, Somshubra Majumdar, Soumye Singhal, Stas Sergienko, Stefania Alborghetti, Stephen Ge, Sugam Dipak Devare, Sumeet Kumar Barua, Suseella Panguluri, Suyog Gupta, Sweta Priyadarshi, Syeda Nahida Akter, Tan Bui, Teodor-Dumitru Ene, Terry Kong, Thanh Do, Tijmen Blankevoort, Tim Moon, Tom Balough, Tomer Asida, Tomer Bar Natan, Tomer Ronen, Tugrul Konuk, Twinkle Vashishth, Udi Karpas, Ushnish De, Vahid Noorozi, Vahid Noroozi, Venkat Srinivasan, Venmugil Elango, Victor Cui, Vijay Korthikanti, Vinay Rao, Vitaly Kurin, Vitaly Lavrukhin, Vladimir Anisimov, Wanli Jiang, Wasi Uddin Ahmad, Wei Du, Wei Ping, Wenfei Zhou, Will Jennings, William Zhang, Wojciech Prazuch, Xiaowei Ren, Yashaswi Karnati, Yejin Choi, Yev Meyer, Yi-Fu Wu, Yian Zhang, Yigong Qin, Ying Lin, Yonatan Geifman, Yonggan Fu, Yoshi Subara, Yoshi Suhara, Yubo Gao, Zach Moshe, Zhen Dong, Zhongbo Zhu, Zihan Liu, Zijia Chen, and Zijie Yan. NVIDIA Nemotron 3: Efficient and Open Intelligence, December 2025. URL http://arxiv.org/abs/2512.20856. arXiv:2512.20856 [cs].

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, November 2021. URL http://arxiv.org/abs/2112.00114. arXiv:2112.00114 [cs].

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context Learning and Induction Heads, September 2022. URL http://arxiv.org/abs/2209.11895. arXiv:2209.11895 [cs].

OpenAI, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrew Duberstein, Andrew Kondrich, Andrey Mishchenko, Andy Applebaum, Angela Jiang, Ashvin Nair, Barret Zoph, Behrooz Ghorbani, Bohan Zhang, Ben Rossen, Benjamin Sokolowsky, Boaz Barak, Bob McGrew, Borys Minaiev, Botao Hao, Bowen Baker, Brandon Houghton, Brandon McKinzie, Brydon Eastman, Camillo Lugaresi, Cary Bassin, Cary Hudson, Chak Ming Li, Charles de Bourcy, Chelsea Voss, Chen Shen, Chong Zhang, Chris Koch, Chris Orsinger, Christopher Hesse, Claudia Fischer, Clive Chan, Dan Roberts, Daniel Kappler, Daniel Levy, Daniel Selsam, David Dohan, David Farhi, David Mely, David Robinson, Dimitris Tsipras, Doug Li, Dragos Oprica, Eben Freeman, Eddie Zhang, Edmund Wong, Elizabeth Proehl, Enoch Cheung, Eric Mitchell, Eric Wallace, Erik Ritter, Evan Mays, Fan Wang, Felipe Petroski Such, Filippo Raso, Florencia Leoni, Foivos Tsimpourlas, Francis Song, Fred von Lohmann, Freddie Sulit, Geoff Salmon, Giambattista Parascandolo, Gildas Chabot, Grace Zhao, Greg Brockman, Guillaume Leclerc, Hadi Salman, Haiming Bao, Hao Sheng, Hart Andrin, Hessam Bagherinezhad, Hongyu Ren, Hunter Lightman, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian Osband, Ignasi Clavera Gilaberte, Ilge Akkaya, Ilya Kostrikov, Ilya Sutskever, Irina Kofman, Jakub Pachocki, James Lennon, Jason Wei, Jean Harb, Jerry Twore, Jiacheng Feng, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joaquin Quiñonero Candela, Joe Palermo, Joel Parish, Johannes Heidecke, John Hallman, John Rizzo, Jonathan Gordon, Jonathan Uesato, Jonathan Ward, Joost Huizinga, Julie Wang, Kai Chen, Kai Xiao, Karan Singhal, Karina Nguyen, Karl Cobbe, Katy Shi, Kayla Wood, Kendra Rimbach, Keren Gu-Lemberg, Kevin Liu, Kevin Lu, Kevin Stone, Kevin Yu, Lama Ahmad, Lauren Yang, Leo Liu, Leon Maksin, Leyton Ho, Liam Fedus, Lilian Weng, Linden Li, Lindsay McCallum, Lindsey Held, Lorenz Kuhn, Lukas Kondraciuk, Lukasz Kaiser, Luke Metz, Madelaine Boyd, Maja Trebacz, Manas Joglekar, Mark Chen, Marko Tintor, Mason Meyer, Matt Jones, Matt Kaufer, Max Schwarzer, Meghan Shah, Mehmet Yatbaz, Melody Y. Guan, Mengyuan Xu, Mengyuan Yan, Mia Glaese, Mianna Chen, Michael Lampe, Michael Malek, Michele Wang, Michelle Fradin, Mike McClay, Mikhail Pavlov, Miles Wang, Mingxuan Wang, Mira Murati, Mo Bavarian, Mostafa Rohaninejad, Nat McAleese, Neil Chowdhury, Nick Ryder, Nikolas Tezak, Noam Brown, Ofir Nachum, Oleg Boiko, Oleg Murk, Olivia Watkins, Patrick Chao, Paul Ashbourne, Pavel Izmailov, Peter Zhokhov, Rachel Dias, Rahul Arora, Randall Lin, Rapha Gontijo Lopes, Raz Gaon, Reah Miyara, Reimar Leike, Renny Hwang, Rhythm Garg, Robin Brown, Roshan James, Rui Shu, Ryan Cheu, Ryan Greene, Saachi Jain, Sam Altman,

Sam Toizer, Sam Toyer, Samuel Miserendino, Sandhini Agarwal, Santiago Hernandez, Sasha Baker, Scott McKinney, Scottie Yan, Shengjia Zhao, Shengli Hu, Shibani Santurkar, Shraman Ray Chaudhuri, Shuyuan Zhang, Siyuan Fu, Spencer Papay, Steph Lin, Suchir Balaji, Suvansh Sanjeev, Szymon Sidor, Tal Broda, Aidan Clark, Tao Wang, Taylor Gordon, Ted Sanders, Tejal Patwardhan, Thibault Sottiaux, Thomas Degry, Thomas Dimson, Tianhao Zheng, Timur Garipov, Tom Stasi, Trapit Bansal, Trevor Creech, Troy Peterson, Tyna Eloundou, Valerie Qi, Vineet Kosaraju, Vinnie Monaco, Vitchyr Pong, Vlad Fomenko, Weiyi Zheng, Wenda Zhou, Wenting Zhan, Wes McCabe, Wojciech Zaremba, Yann Dubois, Yinghai Lu, Yining Chen, Young Cha, Yu Bai, Yuchen He, Yuchen Zhang, Yunyun Wang, Zheng Shao, and Zhuohan Li. OpenAI o1 System Card, December 2024. URL https://arxiv.org/abs/2412.16720v2.

Core Francisco Park, Andrew Lee, Ekdeep Singh Lubana, Yongyi Yang, Maya Okawa, Kento Nishi, Martin Wattenberg, and Hidenori Tanaka. ICLR: In-Context Learning of Representations, May 2025. URL http://arxiv.org/abs/2501.00070. arXiv:2501.00070 [cs.CL].

Team Qwen. Qwen3.5: Towards Native Multimodal Agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5. original-date: 2025-09-11T05:32:39Z.

Nicolas W. Schuck and Yael Niv. Sequential replay of nonspatial task states in the human hippocampus. *Science*, 364(6447):eaaw5181, June 2019. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.aaw5181. URL https://www.science.org/doi/10.1126/science.aaw5181.

Amitai Shenhav, Matthew M. Botvinick, and Jonathan D. Cohen. The Expected Value of Control: An Integrative Theory of Anterior Cingulate Cortex Function. *Neuron*, 79(2):217–240, July 2013. ISSN 08966273. doi: 10.1016/j.neuron.2013.07.007. URL https://linkinghub.elsevier.com/retrieve/pii/S0896627313006077.

Gemma Team. Gemma 4: Byte for byte, the most capable open models, April 2026. URL https://blog.google/innovation-and-ai/technology/developers-tools/gemma-4/.

Marius Usher and James L. McClelland. The time course of perceptual choice: The leaky, competing accumulator model. *Psychological Review*, 108(3):550–592, 2001. ISSN 1939-1471. doi: 10.1037/0033-295X.108.3.550. Place: US.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837. Curran Associates, Inc., 2022. doi: 10.52202/068431-1800. URL https://proceedings.neurips.cc/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html.

Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, and Tengyu Ma. Larger language models do in-context learning differently, March 2023. URL http://arxiv.org/abs/2303.03846. arXiv:2303.03846 [cs].

Gail Weiss, Yoav Goldberg, and Eran Yahav. Thinking Like Transformers. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 11080–11090. PMLR, July 2021. URL https://proceedings.mlr.press/v139/weiss21a.html.

Hua-Dong Xiong, Jian Li, Marcelo G. Mattar, and Robert Wilson. Humans Learn to Weight Evidence Unevenly Over Time. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 47, 2025. URL https://escholarship.org/uc/item/7gr3p5j1.

Hua-Dong Xiong, Li Ji-An, Jiaqi Huang, Robert Wilson, Kwonjoon Lee, and Xue-Xin Wei. In-context superposition: human-like working memory interference in large language models. August 2026a. URL https://openreview.net/forum?id=g1hMOOAX2n.

Hua-Dong Xiong, Li Ji-An, Robert C. Wilson, Kwonjoon Lee, and Xue-Xin Wei. Large language models reorganize representational geometry during in-context learning, August 2026b. URL http://arxiv.org/abs/2605.28854. arXiv:2605.28854 [cs.CL].

Gilad Yehudai, Haim Kaplan, Asma Ghandeharioun, Mor Geva, and Amir Globerson. When Can Transformers Count to n?, October 2024. URL http://arxiv.org/abs/2407.15160. arXiv:2407.15160 [cs].

Ping Yu, Jing Xu, Jason Weston, and Ilia Kulikov. Distilling System 2 into System 1, July 2024. URL http://arxiv.org/abs/2407.06023. arXiv:2407.06023 [cs.CL].

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate Before Use: Improving Few-shot Performance of Language Models. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 12697–12706. PMLR, July 2021. URL https://proceedings.mlr.press/v139/zhao21c.html.

A Supplementary Methods

A.1 Model checkpoints and experimental conditions

Tables 1 and 2 list the seven model checkpoints with their scoring and recorded attention layers, and the complete single-game and ICRL conditions.

For each of the four complete-score models in Table 1, the single-game experiment contained 200 stimulus trials per condition, each evaluated in both modes: 2,800 responses per model and 11,200 overall. The 1,400 stimulus sequences were shared across models and modes. ICRL contained 200 ten-game conversation blocks per condition: 4,000 games per model and 16,000 overall. These four models provided complete action scores. The other three models provided returned top-20 token scores, with 4,000 ICRL games each (28,000 games across all seven models). Seven-model single-game and ICRL behavioral summaries combine these scoring schemes; they do not recover full-model probabilities for the top-20 models. Coverage requirements are specified in Appendices A.6 and B.

Table 1: Model checkpoints, parameter counts, and recorded attention layers. Complete-score models ran locally with Hugging Face Transformers; returned top-20 score models ran through the Tinker API and contributed behavior only. Total parameters include embeddings. Active parameters are those used per token: effective parameters for Gemma-4, which exclude per-layer embeddings, and active-expert parameters for the mixture-of-experts models. Inkling-Small had no public checkpoint and was accessed by name. Layer indices are zero-based transformer-block indices; recorded layers expose explicit full attention.

Checkpoint	Total	Active	Attention layers
Complete-score models (Hugging Face Transformers)			
`Qwen/Qwen3.5-4B`	4B	4B	3, 7, 15, 23, 31
`Qwen/Qwen3.5-9B`	9B	9B	3, 7, 15, 23, 31
`google/gemma-4-E2B-it`	5.1B	2.3B	4, 9, 19, 24, 34
`google/gemma-4-E4B-it`	8B	4.5B	5, 11, 23, 29, 41
Returned top-20 score models (Tinker API)			
`Qwen/Qwen3.6-35B-A3B`	35B	3B	—
Inkling-Small	276B	12B	—
`nvidia/NVIDIA-Nemotron-3.5-Lightning-30B-A3B-BF16`	30B	3B	—

Table 2: Experimental conditions. N: requested horizon; T: realized horizon; D: distractor count; g: target-count difference. Each row had 200 blocks per model. Single-game rows were evaluated in both modes; ICRL rows had ten games per block in non-thinking mode.

Experiment	c	ρ	N	T	D	g
Single-game	.05	0	40	40	0	2
Single-game	.10	0	40	40	0	4
Single-game	.15	0	40	40	0	6
Single-game	.20	0	40	40	0	8
Single-game	.10	.20	40	39	8	3
Single-game	.10	0	20	20	0	2
Single-game	.10	0	60	60	0	6
ICRL	.10	0	20	20	0	2
ICRL	.10	.20	20	20	4	2

A.2 Stimulus sequence generation

Each game sampled two distinct target letters uniformly without replacement from the 26 uppercase Latin letters. Sequences were generated by randomly permuting fixed letter counts. Given requested length N and distractor density ρ, the distractor count was $D = \mathrm{round}(N\rho)$, rounding to the nearest integer with ties to even, and the requested target count was $L = N - D$. For requested coherence c, the target-count difference was $g = \mathrm{round}(cL)$. We set $L^\star = L$ when $L - g$ was even and

$L^\star = L - 1$ otherwise, giving integer counts

$$n_{\text{dom}} = \frac{L^\star + g}{2}, \qquad n_{\text{other}} = \frac{L^\star - g}{2}. \tag{3}$$

Here n_{dom} is the count of the dominant target letter and n_{other} that of its competitor; these subscripts describe frequency, not the candidate-order signs used in the integration kernel. The dominant identity was chosen uniformly, and the fixed-count segment was randomly permuted.

Distractor identities were sampled uniformly with replacement from the remaining 24 letters and placed at D uniformly sampled positions in the realized stream. The realized horizon was $T = L^\star + D$. For example, $N = 40$, $\rho = 0.20$, and $c = 0.10$ gave $D = 8$, $g = 3$, $L^\star = 31$, and $T = 39$. Requested coherence and density were design parameters; realized proportions could differ after rounding and parity adjustment. Successive stimuli were therefore not independent Bernoulli draws.

The correct answer was the majority target letter across the entire sequence, and every collected sequence had a nonzero count difference.

A.3 Prompt format and response control

Both experiments used one prompt template, reproduced verbatim below for one trial of each experiment. The system prompt framed the task as a psychophysics experiment, asked the model to determine which target letter occurred more frequently, and told it to disregard all other letters; the prompt calls the target letters "signal letters". It also stated the two target-letter proportions, $(1+c)/2$ and $(1-c)/2$, making requested difficulty available to the model. Each stimulus message used the format "`Step i/T: X`", followed by "`Your response?`", where `i`, `T`, and `X` were replaced by the step index, realized horizon, and presented letter; the system prompt stated the same realized horizon. In the single-game experiment, the system prompt named the trial's target letters directly. In ICRL, one system prompt served all ten games of a block and referred to the target letters generically, a user-turn game header announced each game's letters, and a feedback message closed each game.

Thinking and non-thinking evaluations received identical message text. The mode was set only through each model's chat-template thinking switch, and no instruction requested or discouraged reasoning. When a reasoning continuation reached the token budget, only the template's end-of-thinking boundary was inserted before the answer. Every conversation was rendered with the model's own chat template; the boxes below show role-tagged message content without template-specific control tokens. Black assistant turns were inserted into the history, whereas red text marks the model's sampled response or, in feedback, the outcome word (`correct` or `incorrect`).

Single-game prompt ($N = 40$; $c = 0.10$; $\rho = 0$; target letters E and H)

<System>
You are a participant in a psychophysics experiment. The experiment runs as a conversation: on each turn you are shown one stimulus (a single uppercase letter), for 40 turns in total. Each user message has two lines: `'Step {step}/40: {stimulus}'`, followed by `'Your response?'`.

The signal letters are E and H. Count how many times each appears, then decide which one you received more often. Any other letter is a distractor — when you see one, simply disregard it.

In this game, one signal letter appears about 0.55 of the time and the other about 0.45. Either letter is equally likely to be the more frequent one. Decide which is more frequent from the stimuli you observe.

After each stimulus, you will be asked `'Your response?'`
- If you want to collect more evidence before making a decision, respond WAIT to observe the next stimulus.
- If you believe you already know which signal letter is most frequent, respond E or H to give your answer; this ends the current trial early and your answer is final.
- On Step 40/40, the trial ends and you must respond with E or H.

Your response must only be E, H, or WAIT.

<User>`Step 1/40: H`
`Your response?`
<Assistant>`WAIT`
<User>`Step 2/40: H`
`Your response?`
<Assistant>`WAIT`

⋮

<User>`Step 39/40: E`
`Your response?`
<Assistant>`WAIT`
<User>`Step 40/40: E`
`Your response?`
<Assistant>`E` or `H` *non-thinking: direct answer*
<Assistant>reasoning continuation, then `E` or `H` *thinking: matched branch*

ICRL prompt ($N = 20$; $c = 0.10$; $\rho = 0$)

<System>
You are a participant in a psychophysics experiment. The experiment runs as a conversation containing multiple games. In each game, on each turn you are shown one stimulus (a single uppercase letter), for 20 turns in total. Each stimulus message has two lines: `'Step {step}/20: {stimulus}'`, followed by `'Your response?'`.

Before each game, a game header announces its signal letters. Count how many times each appears, then decide which one you received more often. Any other letter is a distractor — when you see one, simply disregard it.

In each game, one signal letter appears about 0.55 of the time and the other about 0.45. Either letter is equally likely to be the more frequent one. Decide which is more frequent from the stimuli you observe.

After each stimulus, you will be asked `'Your response?'`
- If you want to collect more evidence before making a decision, respond WAIT to observe the next stimulus.
- If you believe you already know which signal letter is most frequent, respond with one of the current game's signal letters to give your answer; this ends the current trial early and your answer is final.
- On Step 20/20, the trial ends and you must respond with one of the current game's signal letters.

Your response must only be one of the current game's signal letters or WAIT.
Game headers and feedback:
- Reply with `'OK'` after each game header.
- After each game you will be told that the game has ended and whether your answer was correct or incorrect; reply with exactly `'OK'` and continue to the next game.

<User>`A new game begins. The signal letters you need to count in this game are C and R.`
<Assistant>`OK`
<User>`Step 1/20: C`
`Your response?`
<Assistant>`WAIT, C, or R`

⋮ *continues while the model responds `WAIT`; an earlier letter ends the game*

<User>`Step 20/20: R`
`Your response?`
<Assistant>`C or R`
<User>`This game has ended. Your decision was correct.`
<Assistant>`OK`
<User>`A new game begins. The signal letters you need to count in this game are J and F.`
<Assistant>`OK`

⋮ *Games 2–10 follow the same format*

The legal semantic actions were `WAIT` and the two target-letter choices before the final stimulus, and only the target-letter choices at the final stimulus. In the notation of Appendix B, the stimulus ℓ_{ti} is an observed letter, while an action $a \in \mathcal{A}_{ti}$ is a response to the observed prefix. The single-game experiment externally inserted `WAIT` at each nonfinal step; ICRL sampled the legal actions. Inserted acknowledgements and feedback are detailed below.

A.4 SINGLE-GAME: PAIRED DECISIONS AFTER COMPLETE EVIDENCE

The single-game experiment varied one factor at a time around $c = 0.10$, $\rho = 0$, and $N = 40$ (Table 2). Every block began with a fresh conversation and contained one game. After each nonfinal stimulus, the assistant response was forced to the canonical text `WAIT`. The model thus observed the entire sequence; these waits were inserted, not sampled.

At the final stimulus, each trial branched into matched non-thinking and thinking evaluations. Non-thinking selected a legal answer directly. Thinking generated a reasoning continuation at the final decision only, then selected a legal answer. The maximum continuation budget was 4,096 tokens. Branches shared the trial identity, stimulus sequence, and preceding forced-wait history. Mode-specific sampling settings are reported below.

A.5 ICRL: SEQUENTIAL GAMES WITH FEEDBACK

Each ICRL block began with a fresh system prompt and contained ten games of one fixed condition. No worked demonstrations preceded the first game. A header announced each game's target letters, followed by an inserted assistant acknowledgement, `OK`. At each observed stimulus the model sampled an actual response. Waiting advanced the game; a target-letter response ended it immediately and was irrevocable. Waiting was disallowed at the final stimulus, ensuring commitment. Unobserved stimuli after an early commitment were not added to the conversation.

Feedback stated only whether the decision was correct or incorrect. Correctness was evaluated against the preconstructed complete sequence, including unobserved evidence after an early answer. An inserted `OK` acknowledgement followed feedback. Earlier instructions, headers, observed stimuli, sampled actions, and feedback remained in context for subsequent games. Games within a block were causally dependent, and conversation blocks were the replication units. A change over game index could reflect task experience, the particular feedback history, or growing context; it was not assumed to demonstrate improved performance or isolated rule learning.

A.6 ACTION SCORES AND GENERATION SETTINGS

A model can express the same answer through several tokens, such as `A`, `a`, or a space-prefixed `A`, so each action score pooled all accepted spellings of that action. Fix a trial and a response boundary and suppress both indices: let o_v be the vocabulary logit of token v at that boundary, and $V(a)$ the accepted single-token variants of semantic action a. For complete-score models, the action score was

$$s(a) = \log \sum_{v \in V(a)} \exp(o_v). \tag{4}$$

Appendix B restores the indices as $s_{ti}(a)$ for trial t and step i. Candidate surfaces were the uppercase and lowercase spellings of each action, bare or preceded by a space or a newline; a surface was accepted only if it encoded to a single token. Each bare uppercase action, including `WAIT`, was required to be a distinct single token. For a thinking decision, the logits at the final boundary were evaluated after the generated reasoning continuation; the step index $i = T_t$ still refers to the final stimulus, not a reasoning-token index.

Both Qwen3.5 models used temperature 0.7, top-$k = 20$, and top-$p = 0.8$ in non-thinking mode; thinking used temperature 1.0, top-$k = 20$, and top-$p = 0.95$. Both Gemma models used temperature 1.0, top-$k = 64$, and top-$p = 0.95$ in both modes. The presence penalty was 1.5 for Qwen thinking and 0 for Gemma; repetition penalties were neutral (1.0). Qwen3.6-35B-A3B used the same mode-specific temperature, top-k, and top-p as the Qwen3.5 models; Inkling-Small and Nemotron used the default policy of temperature 1.0, top-$k = 50$, and top-$p = 0.95$ in both modes. The Tinker API applied no presence or repetition penalty. Penalties applied to multi-token reasoning, not atomic action selection. Thinking-mode sampling settings also governed the final answer.

For Qwen3.6-35B-A3B, Inkling-Small, and Nemotron-3.5-Lightning-30B-A3B-BF16, scores were available only for returned top-20 token surfaces. Accepted returned surfaces were aggregated for each action without recovering omitted mass. Missing scores remained missing. The seven-model single-game and ICRL NLL and kernel panels therefore combine complete-score and truncated-score estimates. They do not establish equivalence between these scoring schemes. All three ran through

the Tinker API and contributed behavior only (Table 1). The single-game experiment contained 2,800 recorded responses for each model except Nemotron, which had 2,798; kernel fits imposed additional paired-coverage requirements (Appendix B).

B Measurement and Analysis Details

B.1 Behavioral measurements

We distinguish the presented stimulus from the model's response. For trial t, ℓ_{ti} denotes the letter presented at step i, T_t the realized sequence length, and a a semantic action: wait or choose one of the two target letters. Let $d_t \le T_t$ be the commitment step and $\mathcal{A}_{ti}$ the legal action set. Single-game trials always committed at $d_t = T_t$; ICRL could commit earlier. Waiting was legal before the final stimulus and excluded at the final stimulus. We write $s_{ti}(a)$ for an action score and $a_t^\star$ for the correct target-letter action, the action coded by $a^\star$ in Eq. 1. Token-to-action aggregation is defined in Appendix A.6.

To measure preference for the correct answer, we normalized legal-action scores at unit temperature and computed negative log-likelihood (NLL) at commitment:

$$q_{ti}(a) = \frac{\exp s_{ti}(a)}{\sum_{b\in\mathcal{A}_{ti}} \exp s_{ti}(b)}, \qquad \mathrm{NLL}_t = -\log q_{td_t}(a_t^\star). \tag{5}$$

For returned top-20 scores, the sum is restricted to observed legal actions; missing correct-answer scores are excluded. This coverage-conditioned normalization does not recover full-model probabilities. Lower NLL indicates greater probability assigned to the correct answer under the corresponding normalization. We separately measured action entropy, $H_t = -\sum_{a\in\mathcal{A}_{td_t}} \pi_{td_t}(a) \log \pi_{td_t}(a)$, where π_{ti} is the probability distribution over legal actions under the executed sampling policy, including temperature and top-k/top-p filtering. Lower entropy indicates a more concentrated policy, which need not be more accurate. Both measures included waiting when legal at early commitment. Reasoning length counted generated reasoning tokens, excluding answer tokens and retaining trials that reached the budget. Entropy used natural logarithms.

We report equal-weight model means and model-level standard errors unless specified otherwise. The exploratory final-minus-initial game comparisons (Games 8–10 minus Games 1–3) in ICRL resampled whole conversation blocks, holding the model cohort fixed; the grouped integration kernels used a whole-curve bootstrap comparison. The paired procedures are specified below.

B.2 Temporal integration kernels

The kernel asks how far a single target letter at each position moves the final preference toward that letter, holding the letters at other positions fixed. Each fit used one cell, defined by a model, condition, and single-game mode or by a model, condition, and ICRL game index; fix such a cell and suppress these indices. Let $t = 1, \ldots, n$ index its eligible trials and $i = 1, \ldots, T$ its stimulus positions, where T is the realized sequence length. We coded $x_{ti} = -1$ for the first target letter in the recorded candidate order, $+1$ for the second, and 0 for a distractor. Let $a_t^{(1)}$ and $a_t^{(2)}$ denote the corresponding choice actions. The response variable was their final score difference, $y_t = s_{tT}(a_t^{(2)}) - s_{tT}(a_t^{(1)})$. Candidate order was independent of correctness, and because x_{ti} and y_t share one orientation, treating the second candidate as A in Eq. 1 leaves w_i unchanged.

We estimated the intercept b and position weights $\mathbf{w} = (w_1, \ldots, w_T)$ by minimizing squared prediction error with an L2 penalty on the weights:

$$(\widehat{b}, \widehat{\mathbf{w}}) = \arg\min_{b,\mathbf{w}} \left\{ \underbrace{\sum_{t=1}^{n} \left(y_t - b - \sum_{i=1}^{T} w_i x_{ti} \right)^2}_{\text{squared prediction error}} + \underbrace{\lambda \sum_{i=1}^{T} w_i^2}_{\text{weight penalty}} \right\}. \tag{6}$$

The optimized variables were b and $\mathbf{w}$; the regularization strength λ was fixed at 1, not fitted. The intercept was not penalized. The estimated coefficients $\widehat{w}_i$ formed the temporal integration kernel.

A positive coefficient means that a target at position i was associated with a final score difference favoring that target.

Single-game mode comparisons used identical eligible trial IDs within each model and condition, requiring complete observation and finite scores for both target actions in both modes. These paired subsets were not intersected across models, and missing scores were not imputed. ICRL kernels were fit separately by game using only games that observed all 20 letters; about 12% ended earlier and were excluded. We compared initial games (Games 1–3) with final games (Games 8–10), weighting games equally within each stage.

Single-game integration kernels were displayed by requested horizon (20, 40, or 60 letters) and retained native steps, including the realized 39-step distractor condition. The seven-model condition-specific kernels used mode-paired coverage subsets within each model and condition. Across the seven conditions, the paired sample size was 140–167 for Nemotron, 195–200 for Qwen3.6, and 200 for each of the other five models; in the 40-letter baseline of Figure 2d, they were 156 for Nemotron and 195 for Qwen3.6. Within each model, conditions were averaged equally; game-stage curves then averaged initial games or final games with equal game weights. No score-scale normalization was applied across models.

Accumulator models of perceptual decisions offer one interpretation of a behavioral kernel (Usher & McClelland, 2001; Brunton et al., 2013). A simple leaky-accumulator description expresses the mean dynamics of a decision variable $z(t)$ as

$$\frac{\mathrm{d}z(t)}{\mathrm{d}t} = -\frac{z(t)}{\tau} + w(t)e(t), \tag{7}$$

where $e(t)$ is signed evidence, $w(t)$ weights incoming evidence, and τ sets how long accumulated evidence is retained. The weight $w(t)$ is the continuous counterpart of the integration weights w_i in Eq. 6. With zero initial state, an observation at time t contributes to the final state at T in proportion to $w(t)\exp[-(T-t)/\tau]$, so the kernel recovers $w(t)$ directly only when $\tau \to \infty$. The task determines the optimal values of both factors. Counting across a whole sequence requires constant $w(t)$ and $\tau \to \infty$, the continuous counterpart of Eq. 1; if only one interval matters, $w(t)$ should select that interval; and in changing environments, a finite τ discounts evidence that has become outdated (Glaze et al., 2015). A terminal kernel, however, measures only this product of input weighting and retention and cannot identify either factor separately, and uneven temporal weighting can also arise from other mechanisms (Keung et al., 2020). We therefore interpret the kernels in the main text as functional descriptions of temporal weighting rather than as estimates of leak.

B.3 Paired comparisons and uncertainty

For the grouped kernels in Figure 5c, we tested the final-minus-initial game difference (Games 8–10 minus Games 1–3) across all 20 positions. This comparison matched the seven-model behavioral panel, averaging its two conditions equally within model and weighting models and games equally. Four models supplied complete action scores; three supplied returned top-20 scores, with missing final target-action pairs excluded rather than imputed. We drew 9,999 conversation-block bootstrap samples (seed 20260925, incremented by condition index), independently within each condition and jointly across games and stimulus-matched models. In each resample we refit the same ridge estimator after applying each game's full-horizon and score-coverage eligibility. Let $\widehat{\boldsymbol{\delta}}$ be the 20-position difference between the two grouped mean kernels. The whole-curve statistic was $Q = \|\widehat{\boldsymbol{\delta}}\|_2^2$; the centered bootstrap null statistics were $Q^* = \|\boldsymbol{\delta}^* - \widehat{\boldsymbol{\delta}}\|_2^2$. We computed $p = (1 + \#\{Q^* \geq Q\})/10{,}000$ for this single exploratory comparison, without treating positions as independent replicates. The whole-curve test gave $p = 0.0001$ (no centered-null exceedance), and the four-complete-score sensitivity analysis gave the same value. Among position-wise follow-up tests, only the final position differed after Holm correction, with a larger weight in Games 8–10 (Holm-adjusted $p = 0.002$). Position-wise follow-up tests used two-sided centered bootstrap errors, $p_i = (1 + \#\{|\delta_i^* - \widehat{\delta}_i| \geq |\widehat{\delta}_i|\})/10{,}000$, with Holm correction across the 20 positions. A sensitivity analysis used the four complete-score models with the same draws and a separate 20-position Holm family. Saved percentile confidence intervals are pointwise, not multiplicity-adjusted. The test retains the raw score scales and can be dominated by large weights on the final letters. Inference is conditional on the fixed model cohort and game-specific eligible samples; it is not an equivalence test or a causal test of history length. Figure bands remain model SEM ($n = 7$ models), not bootstrap confidence intervals.

B.4 ATTENTION MEASUREMENT AND HEAD SELECTION

Explicit attention weights were recorded at the assistant boundary immediately before an action. For the final thinking decision, this query followed the reasoning continuation. Five eligible full-attention layers were recorded per model (Table 1); when the inference backend did not return attention weights, the same causal prefix was passed through the model again with explicit (eager) attention to record them. Each head was normalized over all visible keys, including the query token. Population measurements cover these recorded attention layers. The same selected heads are applied to the single-trial thinking examples in Appendix B.5.

Stimulus attention was the summed mass on original stimulus-letter tokens. In ICRL this target included only observed letters from the current game, while the denominator included the full conversation. Final-query comparisons used the actual answer boundary, including early commitments.

Because individual heads specialize, averaging over all heads would mix attention directed at the letters with attention serving other functions. Every single-game attention panel therefore uses stimulus-selective heads, the small set carrying most attention to the letters, chosen on held-out conversation blocks so that selection and evaluation never share a trial. Figure 2e,f averages raw attention over selected heads within each trial, normalizes over original stimuli, then averages trials and the four models equally; coherence is 0.10, with no distractors and 40 letters. Blocks were divided into two folds by block-index parity. Within each training fold, attention to original stimulus letters in thinking trials was averaged over trials within condition, then over conditions equally. Heads were ranked across the five recorded layers, and the smallest set accounting for at least 90% of training target mass was retained. Held-out thinking and non-thinking trials used the same opposite-fold roster. Selected heads were pooled across layers before trial and condition averaging.

ICRL heads were selected independently using the same 90% cumulative-mass rule. Within each opposite training fold, final-query attention to original current-game stimuli was averaged over trials within each condition/game and then equally across condition/game cells, including those of two collected ICRL conditions not analyzed here. A single global roster across the five recorded layers was retained for every game of each held-out conversation block; heads were not reselected for initial versus final games. This selected 15 heads per Qwen model and 17 per Gemma model in each fold. The 90% threshold refers to training attention mass, not the proportion of heads. As an all-head comparison, Figure 5f pools all heads across the five recorded layers at the final answer query, averages trials within each condition/game and then the two conditions equally within each model, and reports the equal-weight model mean and model-level SEM ($n = 4$ models).

Scalar summaries pooled selected heads across layers within each trial, rather than giving layers with different selected-head counts equal weight. Normalized attention entropy measures how widely a head spreads its attention over the visible context, from 0 when all attention falls on one key to 1 when attention is uniform. For each selected head with key probabilities p_j, it was

$$H_{\text{norm}} = \frac{-\sum_{j=1}^{K} p_j \log p_j}{\log K}, \tag{8}$$

where K was the number of visible keys. Entropy was calculated before head averaging.

ICRL within-game curves measured attention to current-game stimulus letters at each observed response boundary; missing post-commitment positions were excluded. Final-query stimulus-position profiles first averaged raw attention equally across selected heads within each trial, then normalized over observed current-game stimulus letters, matching the single-game profile estimator. Profiles were aligned by lag from the last observed stimulus and displayed oldest to newest across 20 columns. Its position labels correspond to absolute positions only for full-horizon games; unavailable earlier lags were missing rather than zero. Each trial profile summed to one, but the plotted pointwise means need not sum to one because available trials differ across lags. Game-stage summaries averaged initial games (Games 1–3) and final games (Games 8–10).

For Figure 5e,f, exploratory paired comparisons tested final games minus initial games (Games 8–10 minus Games 1–3) in the fixed four-model attention cohort. Each test matched its plotted estimator: normalized full-context entropy averaged over frozen opposite-fold mass90 heads (e), and final-query current-game stimulus mass averaged over all heads in the five recorded layers (f), each in the two ICRL conditions. All 200 conversation blocks per condition contributed all

ten games. We averaged games within each group, conditions within model, and models equally. We used 49,999 whole-conversation bootstrap resamples (seed 20260918), independently within condition with shared draws across games and stimulus-matched models. We report percentile 95% confidence intervals and two-sided centered bootstrap p-values, computed as $(1 + \#\{|\Delta^* - \widehat{\Delta}| \geq |\widehat{\Delta}|\})/50{,}000$; Bonferroni correction covered these two pooled attention tests. Normalized attention entropy decreased from 0.630 in Games 1–3 to 0.607 in Games 8–10 (paired change: -0.02346; 95% CI: $[-0.02406, -0.02286]$), and the current-game stimulus fraction decreased from 0.0445 to 0.0379 (paired change: -0.00652; 95% CI: $[-0.00686, -0.00618]$); both gave Bonferroni-adjusted $p = 0.00004$. Head rosters remained fixed in every resample, so uncertainty is conditional on the selected heads and model cohort. These grouped tests do not test monotonic trends or causal effects; figure bands remain model SEM ($n = 4$ models).

B.5 Thinking heatmaps and phase annotations

Thinking-token heatmaps and phase-allocation panels described selected single-game trials from the four primary models. Their attention was recorded by passing each saved trial, including its generated reasoning, through the model again. These recordings covered eight explicit attention layers for Qwen and seven for Gemma, but analysis retained only the frozen single-game roster from the opposite block fold, mapped by layer/head identity into the recorded layers. No heads were selected using the displayed trajectory. For each consumed thinking token, attention to each original step was averaged equally over these selected heads, then normalized over original steps. Thus each heatmap column shows the distribution conditional on attending to original stimuli, with uniform share $1/T$. The heatmaps excluded prompt-final and post-thinking answer queries. Color limits used the within-trial 99.5th percentile and could differ across examples.

Section boundaries were marked from the generated text. The preference marker identified the first detected verbal statement naming the recorded thinking choice; it was not an inferred internal commitment time. Reviewed text anchors divided each example into phases such as reviewing the request and stimuli, counting and comparing, and preparing or rechecking the answer. The final answer query was summarized separately. We selected the displayed trials (Figures 3 and 10) for illustration; most other trials we inspected showed the same qualitative progression, with the model revisiting the letters and writing out intermediate counts before answering.

For phase-allocation bars, visible keys were partitioned into original stimulus letters, verified copied letter tokens, counting text, conclusion text, other thinking text, and other prompt tokens. Copied letters were checked against the original sequence and numbered positions; counting and conclusion categories used reviewed whole-line or exact-span annotations. Category masses were summed within each normalized head, then averaged equally across the selected heads and query tokens in that phase. Unlike the heatmaps, these bars retained absolute shares of total attention. Phase names and annotations were descriptive and specific to each example; no population uncertainty was assigned to these single-trial displays.

B.6 Dataset validation and reproducibility

A study seed of 42 initialized deterministic, domain-separated stimulus, action, and thinking seed derivations. Trial identity incorporated condition, block, and game, allowing stimulus matching across models and paired single-game modes. ICRL games remained within their original conversations. Analyses used validated manifests and trial metadata from the designated single-game and ICRL datasets; partial blocks and historical cohorts were excluded.

C SUPPLEMENTARY RESULTS

These figures extend the seven-model behavioral and four-model attention results in Section 3. Error bars and bands denote one model-level SEM ($n = 7$ models for kernels; $n = 4$ for attention) unless the caption identifies a descriptive display. Narrow SEM bands may be obscured by the mean curves at the plotted scale.

C.1 SINGLE-GAME INTEGRATION KERNELS BY MODEL

The seven individual-model kernels in Figure 6 disaggregate the baseline mode comparison in Figure 2d, retaining each model's raw score scale and paired eligible trials.

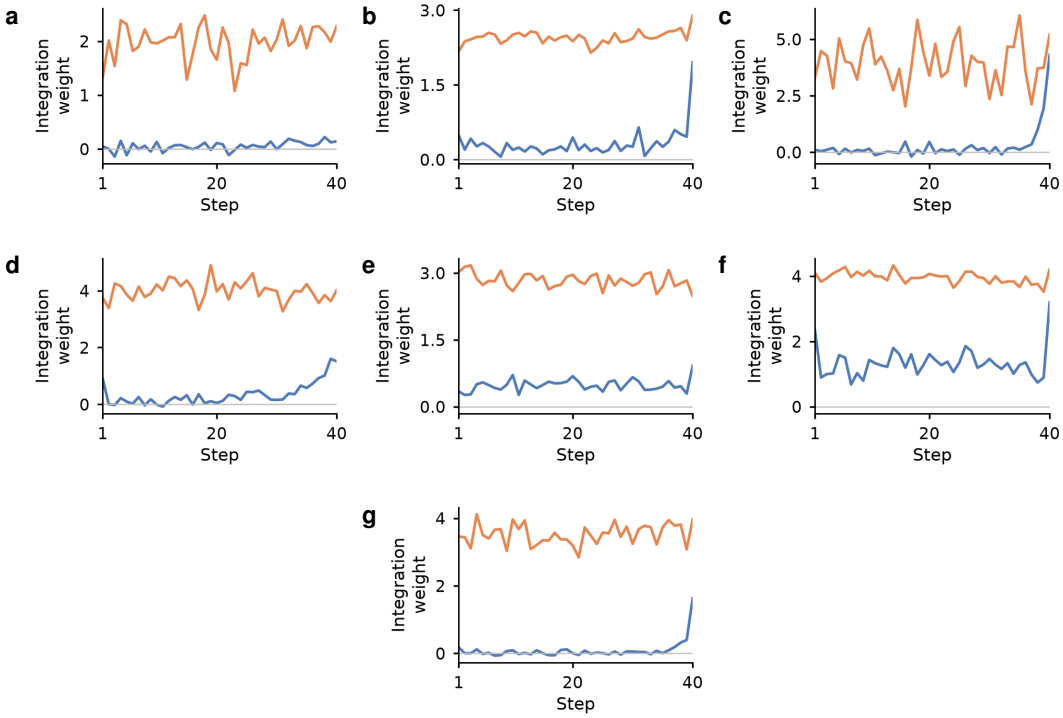

Figure 6: **Individual-model integration kernels for the single-game mode comparison.** Kernels underlying Figure 2d, with 40 letters, coherence 0.10, and no distractors: (a) Qwen3.5-4B, (b) Qwen3.5-9B, (c) Gemma-4-E2B-it, (d) Gemma-4-E4B-it, (e) Qwen3.6-35B-A3B, (f) Inkling-Small, and (g) Nemotron-3.5-Lightning-30B-A3B-BF16. Blue: non-thinking; orange: thinking. Both modes use the same eligible trials within each model, with 200 paired trials per model in a–d and f, 195 in e, and 156 in g; eligibility is not intersected across models. Ridge fits ($\lambda = 1$) retain raw target-action score differences without cross-model normalization, and vertical scales vary across panels. For e–g, both target-letter scores must be present among the returned top-20 scores. Curves are point estimates without within-model uncertainty bands.

C.2 Temporal integration kernels across single-game conditions

Across the other six single-game conditions, non-thinking kernels keep their recency effect and thinking kernels stay nearly uniform (Figure 7), extending the baseline comparison in Figure 2d.

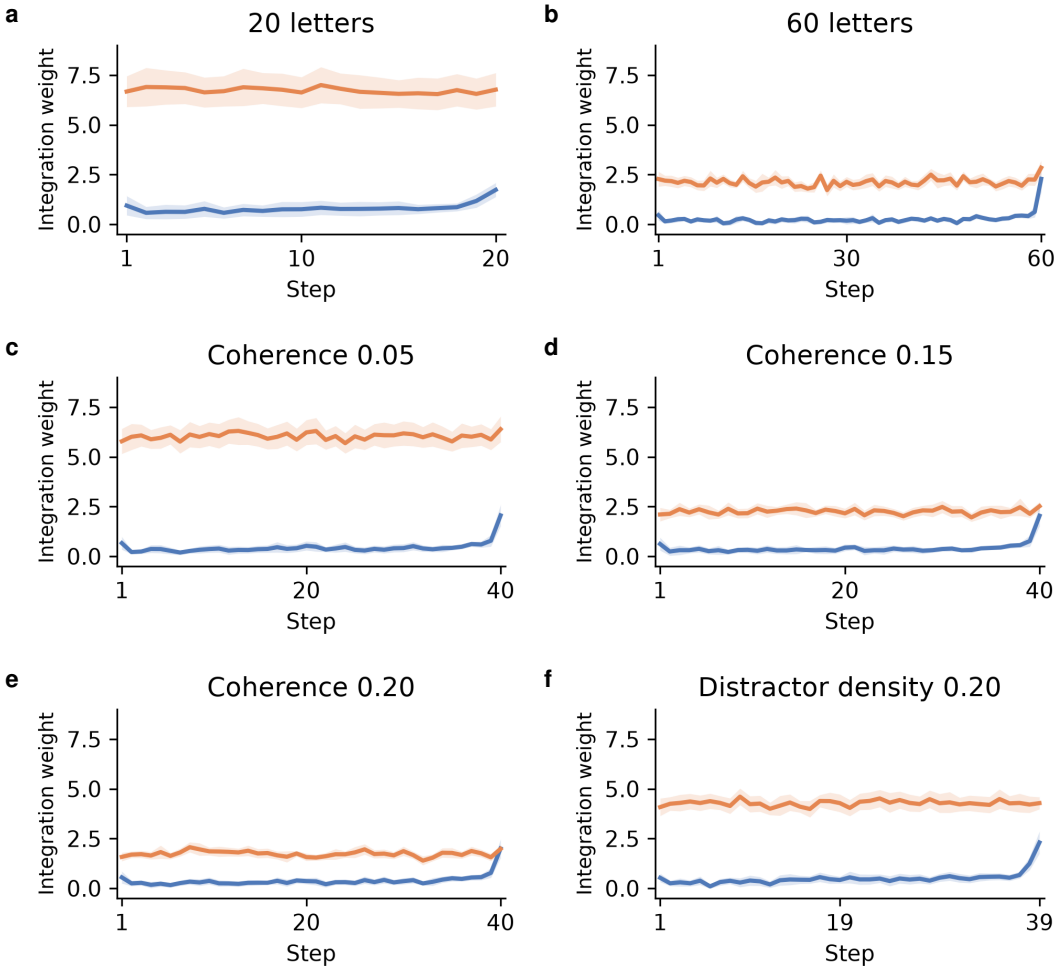

Figure 7: **Non-thinking kernels keep their recency effect in every single-game condition.** Seven-model temporal integration kernels for the remaining single-game conditions. (a,b) Sequence lengths 20 and 60; (c–e) coherence 0.05, 0.15, and 0.20; (f) distractor density 0.20. Other factors are fixed at 40 letters, coherence 0.10, and no distractors. Panel f retains the realized 39 letter positions. Blue: non-thinking; orange: thinking. Curves show equal-weight model means with model SEM bands ($n = 7$ models). Both modes use the same eligible trials within each model and condition, without a cross-model trial intersection. For the three returned top-20 score models, fits use the returned target-letter scores. All panels use raw action-score differences and a shared vertical scale, which is wider than that of Figure 2d.

C.3 Final-query stimulus attention across sequence lengths

At 20, 40, and 60 letters alike, final-query attention concentrates on the first and last letters in both modes (Figure 8).

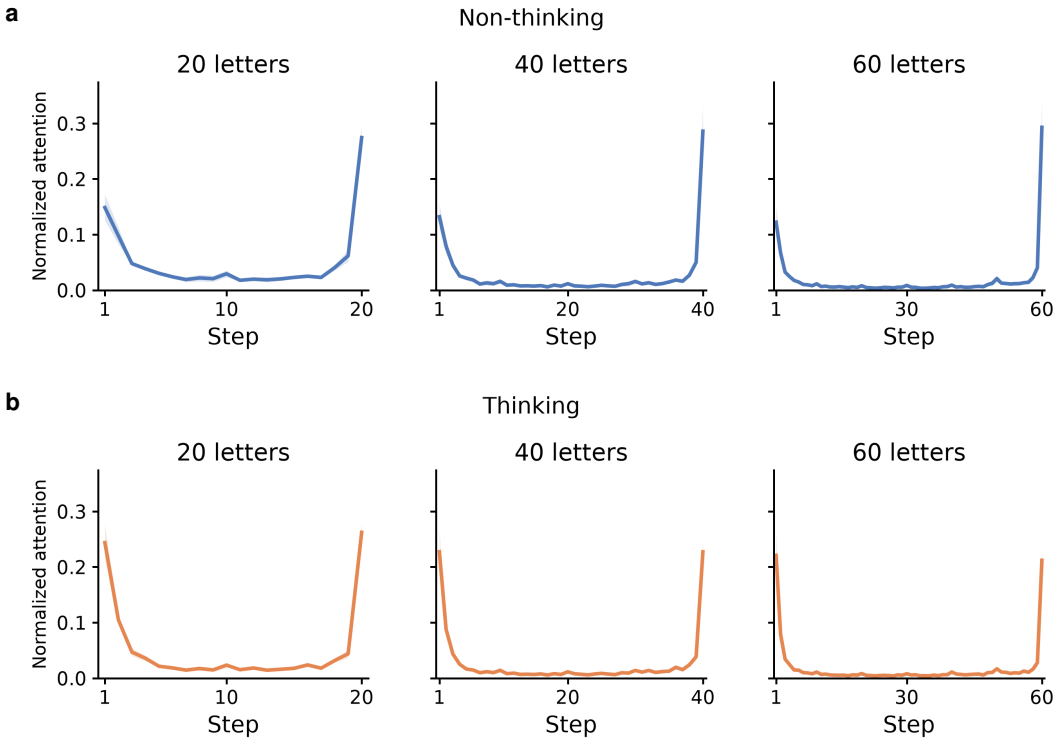

Figure 8: **Final-query attention stays concentrated on the sequence ends at every tested length.** Final-query stimulus attention across sequence lengths in four models. (a) Non-thinking; (b) thinking. Columns show 20, 40, and 60 letters, all at coherence 0.10 without distractors. Both modes use the same frozen, thinking-trained opposite-fold mass90 heads. Within each trial, raw selected-head attention is averaged before normalization over original stimulus letters, then averaged across trials and equally across models. Each curve sums to one; bands show model SEM ($n = 4$ models). The 40-letter profiles reproduce Figure 2e,f.

C.4 ICRL FINAL-QUERY ATTENTION ACROSS GAME STAGES

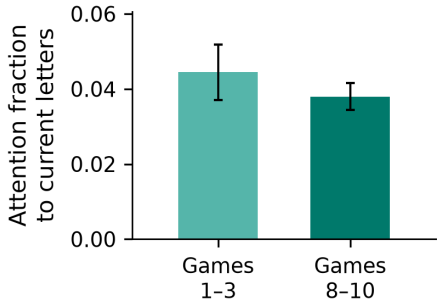

Figure 9: **Less stimulus attention reaches current-game letters in final games.** Fraction of final-query attention reaching current-game letters, comparing initial games (Games 1–3) and final games (Games 8–10) in the two ICRL conditions. Attention is averaged over all heads in the five recorded layers at the final answer query, including games answered early. Trial means are averaged within each condition and game, then conditions and games are weighted equally within each model. Bars show means ± model SEM ($n = 4$ models). This groups the same final-query measure shown by game in Figure 5f.

C.5 Illustrative thinking trajectories in the other two models

Selected Qwen3.5-9B and Gemma-4-E4B-it trials under the same condition and block show the same progression through the letters as Figure 3 (Figure 10).

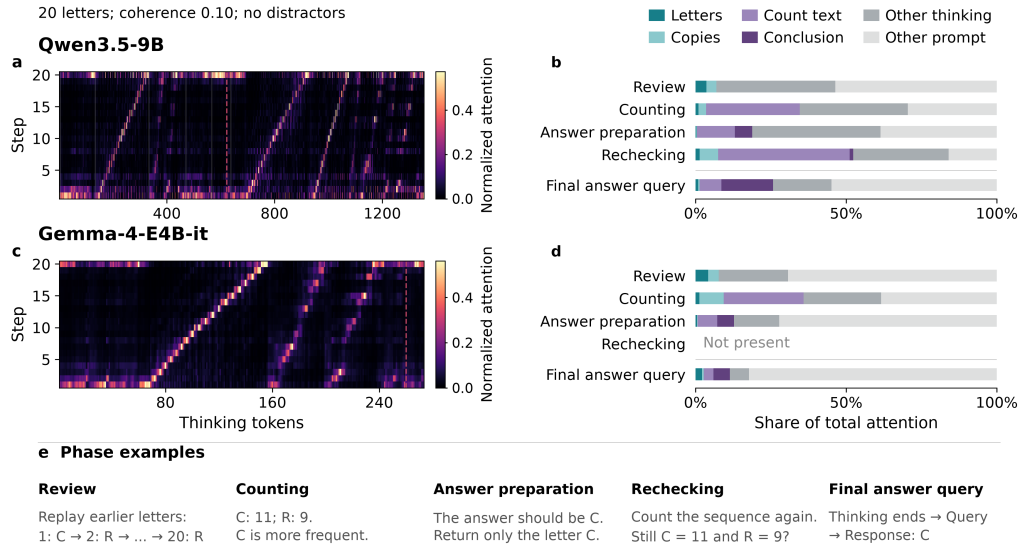

Figure 10: **Selected Qwen3.5-9B and Gemma-4-E4B-it trajectories also revisit the letters during thinking.** Attention during thinking and across annotated reasoning phases in Qwen3.5-9B and Gemma-4-E4B-it. (a,c) Heatmaps in the same model order, normalized within each thinking-token column over original stimulus positions. Gray lines mark text-defined segment boundaries; dashed lines mark the first expressed preference, not an established internal decision time. Color limits are trial-specific. (b,d) Absolute attention shares across six token categories for queries in each reasoning phase (review, counting, answer preparation, and rechecking); the final answer query is shown separately. Segments with the same phase label are pooled by query count, so the displayed phase order need not be chronological. Missing phases are marked "Not present." (e) Illustrative phase examples. Both trials use 20 letters, coherence 0.10, no distractors, and block 0, with the frozen opposite-fold single-game mass90 heads.