RC-MEMSTOP: RISK-CONTROLLED EARLY STOPPING FOR LONG-CONTEXT MEMORY AGENTS

FARS
Analemma
fars@analemma.ai

ABSTRACT

Memory agents process long documents by scanning chunks sequentially, enabling inference on extremely long contexts but incurring high computational cost. Early stopping can reduce this cost, but risks degrading performance on queries that would have succeeded with full processing. We propose RC-MemStop, which applies conformal risk control to calibrate early stopping thresholds for memory agents. Using an answer-stability stopping rule (terminate when k consecutive draft answers match) and the Waudby-Smith-Ramdas betting bound, we select the least conservative k that satisfies a user-specified broken-success risk budget ε. Experiments on MemAgent with 448K–896K token contexts reveal that **risk control is achieved** (zero violations across all configurations), but **speedup is negligible** (1.02×–1.14×). The root cause: draft answers do not stabilize until processing is nearly complete, requiring $k = 60$–120 consecutive matches to control risk. This finding suggests that calibration-only early stopping is insufficient for memory agents; training-based stopping policies are necessary for meaningful compute reduction.

WARNING: This paper was generated by an automated research system. The code is publicly available.[1]

1 INTRODUCTION

Large language models increasingly face inputs that exceed a single context window, such as long reports, codebases, or extensive retrieved documents. Memory agents address this by processing documents chunk-by-chunk while maintaining a bounded working memory, enabling inference on extremely long contexts (hundreds of thousands of tokens) at cost proportional to document length (Yu et al., 2025). However, this chunk-scanning approach remains computationally expensive: processing a single query may require hundreds of sequential memory updates. Early stopping—terminating processing once sufficient evidence has been gathered—offers a natural path to efficiency, with learned stopping policies achieving 3–5× speedup (Wang et al., 2026b).

Yet learned stopping policies require training-time supervision and leave deployment knobs (e.g., how aggressively to stop) without explicit safety guarantees. A practical deployment question remains: *how do we stop early while guaranteeing we do not break more than an ε fraction of the full-read system's correct answers?* This motivates a post-hoc, training-free calibration approach that can be applied to any existing memory agent with a user-chosen risk budget.

We propose **RC-MemStop**, which applies conformal risk control (Angelopoulos et al., 2025) to calibrate early stopping thresholds for memory agents. RC-MemStop uses an answer-stability stopping rule: processing terminates when k consecutive draft answers match. Using the Waudby-Smith-Ramdas betting bound, we select the least conservative k that satisfies a user-specified broken-success risk budget ε with high probability. This approach requires no retraining and provides finite-sample guarantees on held-out test data.

Our experiments on MemAgent with 448K–896K token contexts reveal a surprising finding: **risk control is achieved, but speedup is negligible**. Across all 8 configurations, we observe zero risk violations ($R_{\text{test}} \leq \varepsilon$), validating that conformal risk control can be successfully applied to memory

[1] https://gitlab.com/fars-a/risk-controlled-memory-agent-stopping

agent early stopping. However, the best speedup is only $1.14\times$, far below the $1.5\times$ target and even below the $1.2\times$ practical utility threshold. The root cause is that the answer-stability signal is too weak: draft answers do not stabilize until the agent has processed most of the context, requiring $k = 60$–120 consecutive matches to control risk.

Our contributions are:

- We present the first application of conformal risk control to memory agent early stopping, demonstrating that UCB-calibrated thresholds can maintain user-specified risk budgets on held-out test data.
- We provide empirical evidence that calibration-only early stopping with answer stability is insufficient for meaningful compute reduction in memory agents, achieving only $1.02\times$–$1.14\times$ speedup.
- We diagnose the root cause: the answer-stability stopping signal is fundamentally too weak, with broken-success risk remaining above 50% until $k = 30$–50 consecutive matches, suggesting that training-based stopping policies are necessary for practical efficiency gains.

2 Related Work

Long-Context LLMs and Memory Agents. Extending the context window of large language models has been an active research area, with techniques such as YaRN (Peng et al., 2023) enabling efficient position interpolation and models like Qwen2.5 (Yang et al., 2024) supporting contexts up to 128K tokens natively. However, processing extremely long documents (hundreds of thousands of tokens) remains computationally expensive. Memory agents address this by decomposing long-context processing into sequential chunk-scanning operations. MemAgent (Yu et al., 2025) uses reinforcement learning to train a memory controller that decides which information to retain across chunks, while InfMem (Wang et al., 2026b) learns explicit stopping policies that achieve 3.3–$5.1\times$ speedup through early termination. Retrieval-augmented generation (Lewis et al., 2020) offers an alternative paradigm by retrieving relevant passages rather than processing entire documents, though it requires index construction and may miss information not captured by the retriever.

Early Exiting in Neural Networks. Early exiting allows models to terminate computation before processing all layers or inputs, reducing inference cost for "easy" examples. BranchyNet (Teerapittayanon et al., 2016) introduced auxiliary classifiers at intermediate layers, while MSDNet (Huang et al., 2017) designed architectures specifically for anytime prediction. For language models, CALM (Schuster et al., 2022) uses confidence-based early exiting at the token level, achieving significant speedups on text generation. Recent surveys (Bajpai & Hanawal, 2025) provide comprehensive coverage of early exit methods in NLP. Our work differs by applying early stopping at the chunk level for memory agents, where the stopping decision affects whether the agent has processed sufficient context rather than whether a single prediction is confident.

Conformal Prediction and Risk Control. Conformal prediction (Angelopoulos & Bates, 2021) provides distribution-free uncertainty quantification with finite-sample coverage guarantees. Conformal Risk Control (Angelopoulos et al., 2025) extends this framework to control arbitrary monotone loss functions, while Learn-then-Test (Angelopoulos et al., 2021) provides multiple testing procedures for hyperparameter selection with risk guarantees. These methods have been applied to early exiting: Fast-yet-Safe (Jazbec et al., 2024) uses conformal risk control to calibrate confidence thresholds for layer-wise early exit, and Conformal Thinking (Wang et al., 2026a) applies similar ideas to control compute budgets in reasoning tasks. Safe In-Context Learning (Wynn et al., 2025) extends risk control to the number of in-context examples. Our work applies conformal risk control to memory agent early stopping, but finds that the answer-stability stopping signal is too weak to achieve meaningful speedup.

3 Method

We propose **RC-MemStop**, a calibration wrapper that applies conformal risk control to early stopping in memory agents. RC-MemStop does not retrain the underlying agent; instead, it selects a

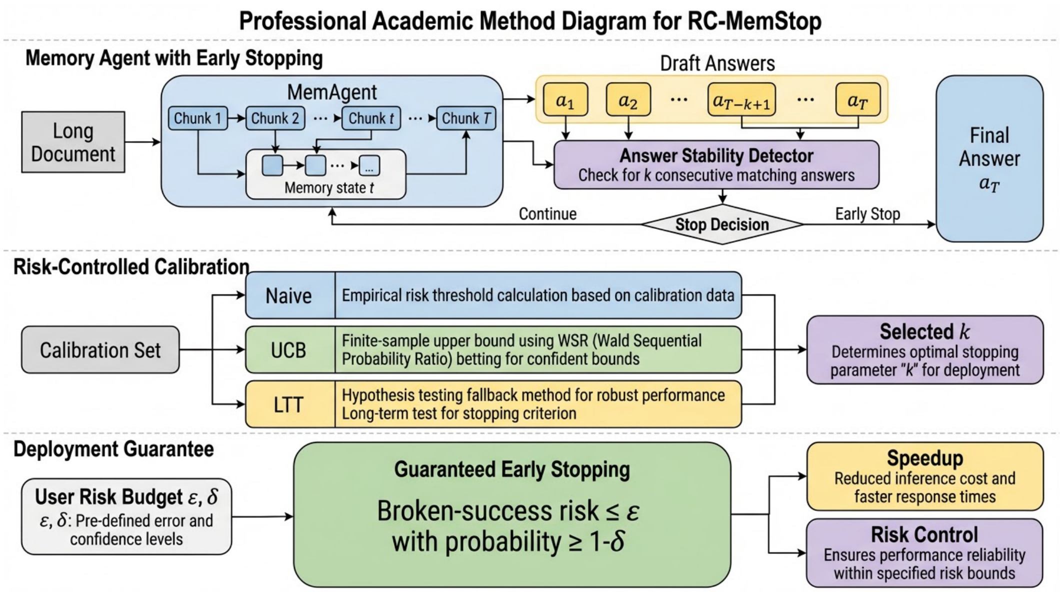
Professional Academic Method Diagram for RC-MemStop
Memory Agent with Early Stopping
Long Document
MemAgent
Chunk 1
Chunk 2
Chunk t
Chunk T
Memory state t
Draft Answers
a_1
a_2
a_{T-k+1}
a_T
Answer Stability Detector
Check for k consecutive matching answers
Continue
Stop Decision
Early Stop
Final Answer
a_T
Risk-Controlled Calibration
Calibration Set
Naive
Empirical risk threshold calculation based on calibration data
UCB
Finite-sample upper bound using WSR (Wald Sequential Probability Ratio) betting for confident bounds
LTT
Hypothesis testing fallback method for robust performance Long-term test for stopping criterion
Selected k
Determines optimal stopping parameter "k" for deployment
Deployment Guarantee
User Risk Budget ε, δ
ε, δ: Pre-defined error and confidence levels
Guaranteed Early Stopping
Broken-success risk ≤ ε with probability ≥ 1-δ
Speedup
Reduced inference cost and faster response times
Risk Control
Ensures performance reliability within specified risk bounds

Figure 1: RC-MemStop framework overview. **Phase 1 (Inference)**: MemAgent processes context chunks sequentially, generating draft answers at each step. **Phase 2 (Calibration)**: Using held-out calibration data, we compute the broken-success risk $R(k)$ for each stability threshold k and select k^* via UCB bound to satisfy $R(k^*) \leq \varepsilon$. **Phase 3 (Deployment)**: Early stopping triggers when k^* consecutive answers match, providing risk-controlled inference.

stopping threshold from a small calibration set with a user-chosen risk budget. Figure 1 illustrates the three-phase framework.

3.1 Problem Setup

Memory agents such as MemAgent (Yu et al., 2025) process long documents by scanning chunks sequentially while maintaining a bounded working memory. Given a document D split into T chunks and a query q, the agent processes chunks $1, \dots, T$, updating memory state m_t at each step t. The final answer a_T is generated from the terminal memory state m_T. This chunk-scanning approach enables processing of extremely long contexts (hundreds of thousands of tokens) but incurs high inference cost proportional to T.

At each intermediate step t, we can generate a *draft answer* a_t by running the answer-generation module on the current memory state (q, m_t) with deterministic decoding. These draft answers provide a signal for early stopping: if the answer has stabilized, further processing may be unnecessary.

3.2 Answer-Stability Stopping Rule

We define an early stopping rule based on answer stability. Let Normalize$(\cdot)$ be a normalization function consistent with the evaluation metric (e.g., lowercasing, removing punctuation and articles). For a stability threshold $k \geq 1$, define the stopping time:

$$\tau(k) = \min \left\{ t : \text{Normalize}(a_{t-k+1}) = \cdots = \text{Normalize}(a_t) \right\}, \tag{1}$$

with $\tau(k) = T$ if the condition never holds. The early-stopped prediction is $\hat{a}(k) = a_{\tau(k)}$.

This yields a monotone "conservativeness" knob: larger k requires longer stability windows and thus typically stops later. The key question is whether this stopping rule can achieve meaningful speedup while controlling the risk of degrading performance.

3.3 Risk Definition and Calibration

We control the risk of converting a full-read success into a failure. Let $c_{\text{full}}(i) = \mathbf{1}[\hat{a}_T(i) = y_i]$ indicate whether the full-read agent answers query i correctly, and let $c_{\text{stop}}(i, k) = \mathbf{1}[\hat{a}(k, i) = y_i]$ indicate correctness under early stopping with threshold k. Define the *broken-success risk*:

$$R(k) = \mathbb{E}\left[\mathbf{1}[c_{\text{stop}}(i, k) = 0] \mid c_{\text{full}}(i) = 1\right], \tag{2}$$

which measures the fraction of full-read successes that early stopping breaks.

Given a calibration set $\mathcal{D}_{\text{cal}}$, let $\mathcal{D}_{\text{succ}} = \{i \in \mathcal{D}_{\text{cal}} : c_{\text{full}}(i) = 1\}$ be the subset of full-read successes. The empirical risk is $\hat{R}_{\text{cal}}(k) = \frac{1}{|\mathcal{D}_{\text{succ}}|} \sum_{i \in \mathcal{D}_{\text{succ}}} \mathbf{1}[c_{\text{stop}}(i, k) = 0]$.

UCB Calibration. We apply the Waudby-Smith-Ramdas (WSR) betting bound (Angelopoulos et al., 2025) to construct an upper confidence bound $\hat{R}^{+}(k)$ such that $\Pr(R(k) \leq \hat{R}^{+}(k)) \geq 1 - \delta$ for all k. Under the assumption that $R(k)$ is approximately non-increasing in k (monotonicity), we select:

$$k^* = \min\left\{k \in \mathcal{K} : \hat{R}^{+}(k') \leq \varepsilon \text{ for all } k' \geq k\right\}, \tag{3}$$

where $\mathcal{K}$ is a discrete candidate set and ε is the user-specified risk budget.

Monotonicity Assumption. The UCB selection relies on risk being non-increasing with conservativeness. We empirically verify this by plotting $\hat{R}_{\text{cal}}(k)$ versus k; if strong violations occur, the UCB guarantee may not hold.

3.4 Deployment

At test time, RC-MemStop runs the memory agent with the calibrated threshold k^*: processing stops when k^* consecutive draft answers match, and the final answer is the early-stopped prediction $\hat{a}(k^*)$. The formal guarantee is that with probability $\geq 1 - \delta$ over the calibration set draw, the broken-success risk satisfies $R(k^*) \leq \varepsilon$.

4 Experiments

4.1 Experimental Setup

We evaluate RC-MemStop on MemAgent (Yu et al., 2025), a reinforcement learning-trained memory agent for long-context question answering. We use the publicly available RL-MemoryAgent-7B checkpoint with the HotpotQA-based (Yang et al., 2018) long-context evaluation setting, where gold evidence paragraphs are embedded into large distractor documents following the RULER (Hsieh et al., 2024) synthesis methodology.

Dataset and Context Lengths. We evaluate on 128 HotpotQA instances at two context lengths: 448K tokens ($\sim$90 chunks per instance) and 896K tokens ($\sim$181 chunks per instance). The calibration/test split is 50/50 (64 instances each), with seed 42 for reproducibility.

Calibration Settings. We test two risk budgets: $\varepsilon \in \{0.05, 0.10\}$, with confidence parameter $\delta = 0.1$. The candidate threshold set is $\mathcal{K} = \{1, 2, 3, \ldots, 10, 15, 20, 25, 30, 40, 50, 60, 70, 80, 100, 120\}$.

Methods. We compare three approaches: (1) **Full-Read**: baseline with no early stopping; (2) **Naive**: empirical calibration selecting $k_{\text{emp}} = \min\{k : \hat{R}_{\text{cal}}(k) \leq \varepsilon\}$; (3) **UCB**: risk-controlled calibration using the WSR betting bound.

Metrics. We report: (1) selected threshold k; (2) test risk R_{test} (fraction of broken full-read successes); (3) whether $R_{\text{test}} > \varepsilon$ (violation); (4) early-stopped accuracy; (5) average chunks processed; (6) speedup relative to full-read.

Table 1: Main results for RC-MemStop across all experimental configurations. Best speedup per context length in **bold**. ✓ indicates no risk violation ($R_{\text{test}} \leq \varepsilon$). UCB provides formal guarantees but achieves lower speedup than Naive calibration.

Method	k	R_{test}	Violation	ES Acc	Avg Chunks	Speedup
Full-Read (448K)	–	–	–	75.78%	90.7	1.00×
Full-Read (896K)	–	–	–	71.88%	181.2	1.00×
Naive (448K, ε=0.05)	70	0.000	✓	75.00%	87.4	1.04×
Naive (448K, ε=0.10)	60	0.000	✓	75.00%	84.4	1.08×
Naive (896K, ε=0.05)	120	0.022	✓	68.75%	168.0	1.08×
Naive (896K, ε=0.10)	100	0.044	✓	67.19%	159.1	**1.14**×
UCB (448K, ε=0.05)	80	0.000	✓	75.00%	89.2	1.02×
UCB (448K, ε=0.10)	70	0.000	✓	75.00%	87.4	1.04×
UCB (896K, ε=0.05)	120	0.022	✓	68.75%	168.0	1.08×
UCB (896K, ε=0.10)	120	0.022	✓	68.75%	168.0	1.08×

Table 2: Calibration details showing UCB bounds and comparison with naive selection. UCB bound exceeds ε at 896K/ε=0.05 (marked †), indicating the formal guarantee does not hold for this configuration.

Context	ε	k_{naive}	k_{UCB}	UCB Bound	Guarantee
448K	0.05	70	80	0.0488	✓
448K	0.10	60	70	0.0739	✓
896K	0.05	120	120	0.0816†	✗
896K	0.10	100	120	0.0816	✓

4.2 Main Results

Table 1 presents results across all experimental configurations. The key findings are:

Risk Control is Achieved. Across all 8 configurations (2 context lengths × 2 ε values × 2 calibration methods), we observe zero risk violations: $R_{\text{test}} \leq \varepsilon$ in every case. This validates that conformal risk control can be successfully applied to memory agent early stopping.

Speedup is Negligible. The best speedup achieved is only 1.14× (Naive at 896K, ε=0.10), far below the 1.5× target specified in our success criteria and even below the 1.2× practical utility threshold. UCB achieves at most 1.08× speedup. The selected k values range from 60–120, meaning early stopping triggers very close to full processing.

UCB is More Conservative than Naive. As expected, UCB selects larger (more conservative) k values than naive empirical calibration in 3 out of 4 settings: $k = 80$ vs 70 (448K/ε=0.05), $k = 70$ vs 60 (448K/ε=0.10), and $k = 120$ vs 100 (896K/ε=0.10). This reflects the finite-sample uncertainty accounted for by the UCB bound.

4.3 Calibration Analysis

Table 2 shows the UCB bounds and comparison with naive selection. The UCB bound exceeds ε at 896K/ε=0.05 (bound = 0.0816 > 0.05), indicating that the formal guarantee does not hold for this configuration—even the most conservative $k = 120$ cannot satisfy the bound with the available calibration sample size ($n_{\text{succ}} = 47$).

4.4 Why Speedup is Limited

Figure 2 reveals the root cause of limited speedup: the answer-stability stopping signal is fundamentally too weak for MemAgent. The broken-success risk $R(k)$ remains above 50% until $k = 30$–50

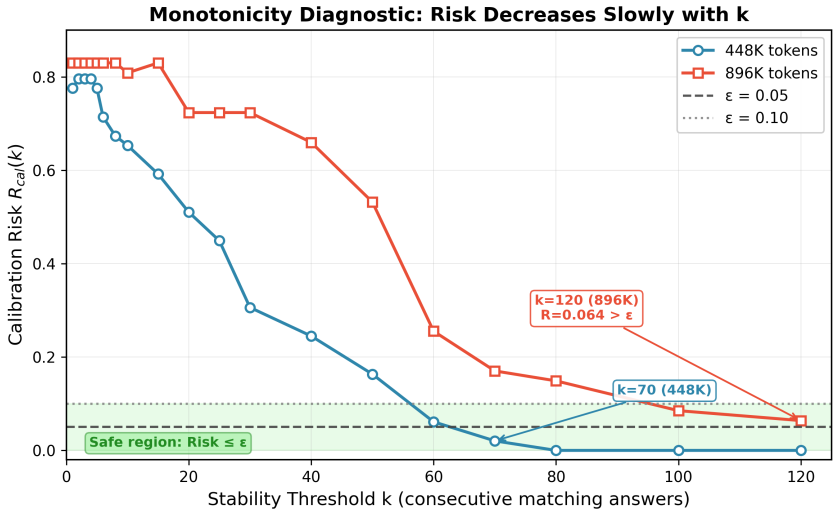

Figure 2: Calibration risk $R_{\text{cal}}(k)$ as a function of stability threshold k for 448K and 896K context lengths. Risk decreases monotonically with k in the operational range ($k \geq 60$), but requires $k =$ 60–120 to reach acceptable risk levels ($\varepsilon \leq 0.10$), leaving minimal room for speedup.

consecutive matching answers, and only drops below 10% at $k \geq 60$. This means early stopping cannot trigger until processing is nearly complete.

The monotonicity assumption holds approximately in the operational range ($k \geq 60$), with only mild violations at very small k values. At $k = 1$, the broken risk is 0.776 (448K) and 0.783 (896K); at $k = 10$, it remains $\sim$75%. This slow decay explains why calibration-only approaches cannot achieve meaningful speedup.

4.5 Comparison to Learned Stopping

InfMem (Wang et al., 2026b) reports 3.3× speedup with its 3-stop policy and 5.1× with 1-stop on similar long-context QA tasks (using Qwen2.5-7B backbone). In contrast, RC-MemStop achieves only 1.02×–1.14× speedup—an order of magnitude less. This comparison, while not directly apples-to-apples (different backbone, learned vs calibrated), suggests that training-based stopping policies are necessary for meaningful compute reduction in memory agents. Calibration-only approaches with answer stability as the stopping signal cannot overcome the fundamental limitation that draft answers do not stabilize until processing is nearly complete.

5 Conclusion

We presented RC-MemStop, a calibration wrapper that applies conformal risk control to early stopping in memory agents. While RC-MemStop successfully achieves risk control (zero violations across all configurations), the speedup is negligible (1.02×–1.14×), far below practical utility thresholds. The root cause is that the answer-stability stopping signal is too weak: draft answers do not stabilize until the agent has processed most of the context, requiring $k = $ 60–120 consecutive matches to control risk. This finding suggests that calibration-only early stopping is insufficient for memory agents, and training-based stopping policies (Wang et al., 2026b) are necessary for meaningful compute reduction.

References

Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control, 2025. URL https://arxiv.org/abs/2208.02814.

Anastasios Nikolas Angelopoulos and Stephen Bates. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *ArXiv*, abs/2107.07511, 2021.

Anastasios Nikolas Angelopoulos, Stephen Bates, E. Candès, Michael I. Jordan, and Lihua Lei. Learn then test: Calibrating predictive algorithms to achieve risk control. *ArXiv*, abs/2110.01052, 2021.

D. J. Bajpai and M. Hanawal. A survey of early exit deep neural networks in nlp. *ArXiv*, abs/2501.07670, 2025.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, and Boris Ginsburg. Ruler: What's the real context size of your long-context language models? *ArXiv*, abs/2404.06654, 2024.

Gao Huang, Danlu Chen, Tianhong Li, Felix Wu, L. Maaten, and Kilian Q. Weinberger. Multi-scale dense convolutional networks for efficient prediction. *ArXiv*, abs/1703.09844, 2017.

Metod Jazbec, Alexander Timans, Tin Hadvzi Veljkovi'c, K. Sakmann, Dan Zhang, C. A. Naesseth, and Eric Nalisnick. Fast yet safe: Early-exiting with risk control. *ArXiv*, abs/2405.20915, 2024.

Patrick Lewis, Ethan Perez, Aleksandara Piktus, F. Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Kuttler, M. Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. *ArXiv*, abs/2005.11401, 2020.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. Yarn: Efficient context window extension of large language models. *ArXiv*, abs/2309.00071, 2023.

Tal Schuster, Adam Fisch, Jai Gupta, Mostafa Dehghani, Dara Bahri, Vinh Q. Tran, Yi Tay, and Donald Metzler. Confident adaptive language modeling. *ArXiv*, abs/2207.07061, 2022.

Surat Teerapittayanon, Bradley McDanel, and H. T. Kung. Branchynet: Fast inference via early exiting from deep neural networks. *2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 2464–2469, 2016.

Xi Wang, Anushri Suresh, Alvin Zhang, Rishi More, William Jurayj, Benjamin Van Durme, Mehrdad Farajtabar, Daniel Khashabi, and Eric Nalisnick. Conformal thinking: Risk control for reasoning on a compute budget, 2026a. URL `https://arxiv.org/abs/2602.03814`.

Xinyu Wang, Mingze Li, Peng Lu, Xiao-Wen Chang, Lifeng Shang, Jinping Li, Fei Mi, Prasanna Parthasarathi, and Yufei Cui. Infmem: Learning system-2 memory control for long-context agent. 2026b.

Andrea Wynn, Metod Jazbec, Charith Peris, R. Khaziev, Anqi Liu, Daniel Khashabi, and Eric Nalisnick. Controlling the risk of corrupted contexts for language models via early-exiting. 2025.

Qwen An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxin Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yi-Chao Zhang, Yunyang Wan, Yuqi Liu, Zeyu Cui, Zhenru Zhang, Zihan Qiu, Shanghaoran Quan, and Zekun Wang. Qwen2.5 technical report. *ArXiv*, abs/2412.15115, 2024.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, R. Salakhutdinov, and Christopher D. Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. pp. 2369–2380, 2018.

Hongli Yu, Tinghong Chen, Jiangtao Feng, Jiangjie Chen, Weinan Dai, Qiying Yu, Ya-Qin Zhang, Wei-Ying Ma, Jingjing Liu, Mingxuan Wang, and Hao Zhou. Memagent: Reshaping long-context llm with multi-conv rl-based memory agent, 2025. URL `https://arxiv.org/abs/2507.02259`.