Selective Memory Materialization for Efficient Multi-Turn Medical Consultation

Anonymous ACL submission

Abstract

Multi-turn medical consultation accumulates long histories, but exposing the full interaction trace at every round increases prompt cost and latency. We study whether memory exposure can instead be allocated dynamically at inference time. We propose CASM, a lightweight controller that maintains a deterministic compact state view, escalates to the full consultation history only under uncertainty, and preserves a prefix-stable prompt layout. The method leaves the consultation backbone, memory system, and serving stack unchanged, isolating memory exposure as the intervention. In a controlled consultation setting, CASM improves the quality–latency operating point over always exposing full history while supporting round-level analysis of when fallback is actually needed. The broader takeaway is that full-history materialization is not a fixed requirement in consultation agents, but an inference-time decision that can be evaluated directly through paired task and systems metrics.

1 Introduction

Large language models (LLMs) have made multi-turn medical consultation a realistic testbed for studying sequential information acquisition and differential diagnosis (Valizadeh and Parde, 2022; Tu et al., 2024; Li et al., 2024; Chen et al., 2024). In a typical trajectory, an agent receives self-reported symptoms, asks follow-up questions, updates disease beliefs, and finally outputs a diagnosis. The interaction history grows round by round, yet the system must keep making intermediate decisions under limited consultation budgets.

Recent consultation frameworks have focused primarily on improving decision quality, for example through better inquiry planning, diagnosis calibration, or multi-agent coordination (Liu et al., 2022; Yan et al., 2023; Hou et al., 2023; Zhao et al., 2024; Hu et al., 2024; Jia et al., 2025). However, they typically leave one systems question under-explored: *does every intermediate round really require full-history materialization?* In many cases, the agent's current decision may depend mostly on a compact summary of confirmed positive symptoms, negative symptoms, asked/unasked status, and the current disease ranking. Repeatedly rematerializing the entire consultation history can therefore be wasteful, especially when prompt prefill dominates latency.

If consultation is treated only as a reasoning benchmark, this systems issue is easy to overlook. In serving stacks, repeated context materialization directly affects deployment cost. Long prompts consume tokens, increase prefill cost, reduce cache reuse, and make latency more sensitive to prompt layout and prefix stability (Kwon et al., 2023). The practical question is whether the agent should pay the cost of full history on every round.

Beyond consultation, recent agent-memory work has begun to treat memory as an active design space rather than a passive transcript buffer, studying efficient lifelong memory, evolving memory skills, and experience-driven skill improvement (Zhang et al., 2026; Liu et al., 2026; Liang et al., 2025; Yang et al., 2026; Ye et al., 2025; Yang et al., 2025; Yu et al., 2025). We take a narrower angle: instead of building a new memory system, we ask whether an existing consultation agent can avoid unnecessary full-history exposure through selective materialization at inference time.

This paper proposes a focused answer. We introduce CASM, a lightweight inference-time controller for multi-turn consultation agents. CASM maintains two dynamic views over the same consultation state: a *compact* view that serializes the current state into a stable structured representation, and a *full-history* view that exposes the complete interaction trace. Stable task instructions and disease-specific knowledge remain shared across both views. At each round, the agent first reasons

with the compact view and escalates to the full-history view only when uncertainty is sufficiently high.

CASM is simple to test. It can be inserted into an existing consultation pipeline without retraining the policy or diagnosis backbone. It still yields explicit task–system trade-off measurements and an inspectable routing decision. If compact memory is sufficient for easy rounds and full history is used mainly on uncertain ones, the routing behavior itself becomes evidence. In this paper, memory exposure is treated as a policy decision rather than a fixed prompt template.

Contributions.

1. **Problem formulation.** We recast full-history exposure in multi-turn consultation as a per-round allocation policy.
2. **Method.** We propose CASM, a deterministic dual-view interface with uncertainty-triggered fallback and a prefix-stable prompt layout.
3. **Mechanism-oriented evaluation.** We evaluate the method with matched task/system metrics together with replay and negative-control analyses so that selective fallback can be tested as a mechanism rather than presented as a prompt tweak.

2 Task Setup and Base Consultation Agent

We instantiate our study on top of a public multi-agent consultation framework (Jia et al., 2025). The base workflow decomposes consultation into four coordinated roles: a diagnosis module that estimates disease confidence, a policy module that proposes candidate inquiry actions, an inquiry module that selects the next symptom to ask about, and a patient simulator that returns the response. The system operates over a shared memory containing both static disease-related knowledge and a dynamic consultation trace.

This setting fits the memory-materialization question well. The dynamic consultation trace grows every round, but the agent also maintains structured state variables that already summarize much of the clinically relevant information: confirmed positive symptoms, confirmed negative symptoms, symptoms that have already been queried, turn budget, and the current disease-confidence distribution. The base agent already exposes two views of the same interaction: a *trajectory view* and a *state view*. CASM uses them as a selective interface.

This framework is a good substrate because intermediate decisions are explicit, memory is updated round by round, and the task has bounded actions and clinically meaningful outcomes. We keep the diagnosis backbone, inquiry workflow, patient simulator, and dataset protocol fixed; the only intervention is how state is serialized, when compact reasoning escalates to full history, and how the prompt is ordered for reuse. This keeps the paper focused on memory exposure rather than on inventing a new consultation policy.

3 Method

3.1 Design Principles

Before describing the controller, we make four design principles explicit. **P1: surgical intervention.** The method should sit on top of an existing consultation agent without retraining its core modules. **P2: deterministic compact state.** If the compact view is itself generated by another unconstrained LLM, then any observed gain or failure becomes difficult to attribute. **P3: transparent escalation.** The trigger should be simple enough that readers can tell when it routes to full history and when it does not. **P4: systems-visible evidence.** Every design choice should leave an observable footprint in latency, prompt length, prefill cost, or escalation behavior. These principles keep the intervention analyzable and reproducible, and separate it from training-heavy agent redesign.

3.2 Control Objective

CASM treats memory exposure as a per-round control problem. Let $v_t \in \{\texttt{compact}, \texttt{full}\}$ denote the memory view selected at round t, and let $c(v_t)$ denote its systems cost, measured through prompt length, prefill work, and latency. The always-full baseline fixes $v_t = \texttt{full}$ for all rounds, while the compact-only baseline fixes $v_t = \texttt{compact}$.

Our goal is to learn a lightweight decision rule that minimizes cumulative systems cost without sacrificing final consultation quality:

$$\min_{\pi_{\text{view}}} \mathbb{E}\Big[\sum_{t=1}^{T} c(v_t)\Big] \quad \text{s.t.} \quad \Delta Q(\pi_{\text{view}}, \pi_{\text{full}}) \leq \epsilon, \tag{1}$$

where $Q(\cdot)$ denotes final task quality (e.g., diagnostic accuracy) and π_{full} is the always-full infer-

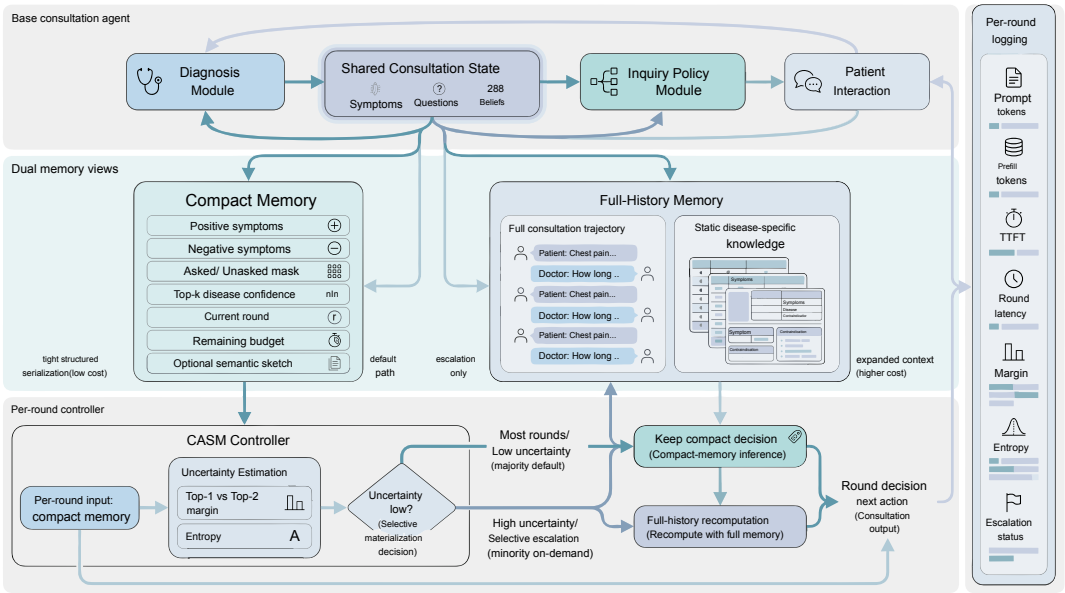

Figure 1: Overview of CASM. The base consultation pipeline is kept fixed while memory exposure becomes the inference-time decision variable. The controller switches between a compact structured state view and the full interaction trace, and logs both task and systems signals for each round.

ence policy. CASM does not solve this optimization with a separate learned router. Instead, it uses uncertainty as a simple and inspectable proxy for whether the extra memory cost is warranted.

3.3 Two Memory Views

Full-history view. The full-history view materializes the complete consultation trace, including the original sequence of past inquiries and responses. This view is the most expressive dynamic representation, but also the most expensive to repeatedly serialize into prompts.

Compact memory. The compact view is a deterministic structured serialization of the current consultation state. In our current implementation, it includes:

- confirmed positive symptoms,
- confirmed negative symptoms,
- asked/unasked symptom mask,
- top-k disease confidence scores,
- current round index and remaining budget,
- an optional semantic sketch inherited from a prior state-augmentation prototype.

We study three codec variants: `slots`, `slots_topk`, and `slots_topk_d1`. The first uses only symptom-status slots; the second augments them with top-k disease beliefs; the third further adds the semantic sketch.

We intentionally keep this serializer deterministic rather than using an additional LLM summarizer. This avoids introducing a second generative bottleneck whose own failures would be difficult to disentangle from the memory-control question. As a result, when CASM succeeds or fails, the effect can be attributed more cleanly to view selection rather than to lossy paraphrasing.

3.4 Uncertainty-Triggered Escalation

At round t, let p_t be the diagnosis distribution predicted under compact memory. CASM first runs the round with compact memory and computes uncertainty statistics from p_t. We consider the top-1/top-2 confidence margin

$$m_t = p_t^{(1)} - p_t^{(2)}, \tag{2}$$

and the entropy

$$H_t = -\sum_i p_t(i) \log p_t(i). \tag{3}$$

The controller escalates to full memory when uncertainty is high:

$$\texttt{use_full}_t = \mathbb{I}[m_t < \delta_m \ \vee \ H_t > \delta_H]. \tag{4}$$

We evaluate three trigger families: `margin`, `entropy`, and `margin+entropy`. This yields three operating modes:

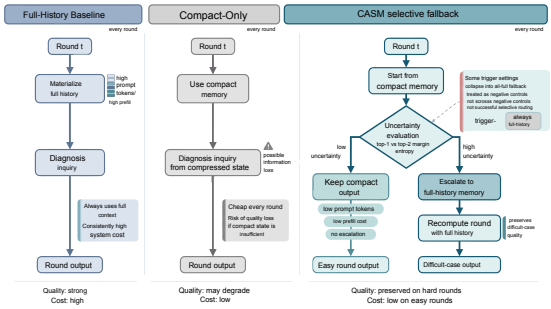

Figure 2: Round-level decision flow under three policies. Always-full inference materializes the full interaction trace on every round, compact-only never does, and CASM starts from the compact view before falling back to full history only when the uncertainty trigger fires.

- **full**: always use full memory;
- **compact**: always use compact memory;
- **casm**: use compact memory by default and escalate only when triggered.

This design is intentionally conservative. We prefer a small family of transparent triggers over a more powerful but opaque controller, because the paper's main value lies in establishing a clear mechanism: some rounds deserve full-history materialization and others do not. Figure 2 visualizes this decision process as a three-way comparison among always-full, compact-only, and selective fallback.

3.5 Prefix-Stable Prompt Layout

CASM is motivated by token and prefill efficiency, so prompt organization matters. We therefore place stable instructions and reusable policy scaffolding before dynamic memory fields whenever possible. This keeps the prompt prefix stable across rounds and makes prompt-side systems measurements easier to interpret. The same ordering is compatible with prefix-caching backends and paged serving engines (Kwon et al., 2023), but our main claim does not depend on a cache-specific implementation.

3.6 Instrumentation and Evidence Protocol

Each case/round emits structured logs that include the case id, round id, dataset, model, memory mode, whether full memory was used, whether the round escalated, confidence margin, entropy, prompt tokens, prefill tokens when available, TTFT, and round latency. These logs are required for the mechanism-focused analyses. In particular, it lets us analyze:

- whether escalation concentrates on later rounds,
- whether difficult cases trigger more fallback,
- whether compact/full disagreement correlates with useful escalation.

4 Experimental Setup

4.1 Benchmarks and Models

We use public medical consultation benchmarks centered on DXY, GMD, and CMD, following prior work on automatic consultation and multi-agent diagnosis (Chen et al., 2023b; Zhao et al., 2024; Chen et al., 2024; Jia et al., 2025). Our main development backbone is a 7B instruction-tuned model to keep threshold sweeps and per-round logging tractable; a larger backbone is reserved for confirmation after the control policy is frozen.

4.2 Baselines and Comparison Lines

To avoid overloading the story with too many variants, the main paper uses the following externally visible names:

- FULL-HISTORY BASELINE: original full-history inference;
- COMPACT-ONLY: compact memory only;
- CASM: standard selective-fallback configuration;
- $\text{CASM}_{\text{CONS}}$: a more conservative selective-fallback configuration;
- CASM_{AGG}: a more aggressive efficiency-oriented configuration.

The main text avoids framework-specific naming because the contribution is the selective-materialization policy, not a renamed consultation baseline.

4.3 Metrics

We report final diagnostic accuracy (**Acc**) and average consultation turns (**Avg.n**), together with end-to-end latency, TTFT, prompt tokens, and escalation rate. When backend instrumentation permits, we additionally inspect prefill-oriented measurements. Together these metrics show the accuracy–efficiency trade-off of each method.

4.4 Evaluation Protocol

Thresholds are tuned on DXY development data, frozen, and then transferred to DXY test; only after that is the same setup evaluated on GMD and

CMD with minimal retuning. All primary task and latency measurements are collected on one designated experiment server so that cross-machine variance does not contaminate the core tradeoff claims.

Two discipline rules are especially important. First, matched-split evidence decides ranking; exploratory rows only reveal regime shape. Second, latency-facing numbers are read only under repeated same-hardware measurement. The paper would rather claim less than mix exploratory and frozen evidence into one table.

4.5 Hypotheses and Success Conditions

The paper is organized around three hypotheses. **H1**: compact-only inference is a meaningful lower bound but not a sufficient policy. **H2**: selective fallback recovers most of that gap while still saving memory-related cost relative to FULL-HISTORY BASELINE. **H3**: a successful controller remains genuinely selective rather than collapsing into almost-always-full routing. These hypotheses also define the reading standard: *strong* means lower latency or prefill cost with no meaningful accuracy drop; *acceptable* means a smaller systems gain with clear selectivity; *informative* means compact-only fails but controlled fallback still recovers part of the loss.

5 Main Results

Table 1 isolates four completed runs on the same development slice. Three points stand out. First, CASM occupies the desired selective middle regime: it improves accuracy from 0.875 to 0.906 while reducing TTFT and end-to-end latency relative to FULL-HISTORY BASELINE, and it does so with an escalation rate of 0.577 rather than near-uniform full-history use. Second, $\text{CASM}_{\text{CONS}}$ and CASM_{AGG} expose the tradeoff frontier around that point, showing that the controller is not a one-off prompt variant but a tunable allocation policy. Third, the small prompt-token reduction suggests that the latency gain should be read as an operating-point improvement, not yet as evidence for a specific cache-level mechanism.

Two ratios sharpen this reading. CASM reduces prompt tokens by only 1.58% relative to FULL-HISTORY BASELINE, yet lowers end-to-end latency by 5.54% while keeping the same average number of consultation rounds. This pattern is unlikely to come from simple truncation

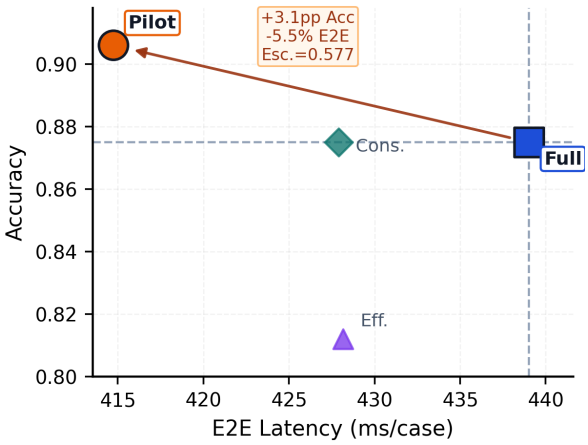

Figure 3: Accuracy–latency Pareto view of the matched DXY development slice. CASM moves up and left relative to the always-full anchor, while remaining selective: its escalation rate is 0.577 rather than collapsing into uniform full-history exposure.

alone. The controller is not deleting most context up front; it is changing *when* expensive full-history exposure is paid.

CASM is interesting only if it buys quality with *partial* full-history use. If a strong row were achieved only by collapsing into almost-always-full inference, then selective materialization would not be established. This slice therefore suggests a middle regime between always-full inference and overly compressed routing. The claim is intentionally narrow: on the development slice, selective materialization produces a measurable operating point that is neither "always expose everything" nor "compress everything".

The secondary CASM rows sharpen this interpretation. $\text{CASM}_{\text{CONS}}$ preserves the baseline accuracy while giving back part of the latency gain, whereas CASM_{AGG} spends less full-history budget but loses too much diagnostic quality. The pattern matters because it traces a continuous quality–cost frontier around CASM rather than producing a single favorable row. The controller behaves like a tunable allocation policy, not an unexplained prompt accident.

6 Mechanism Diagnostics

A selective controller should not be judged by one table alone. We therefore evaluate CASM along three mechanism axes: round position, case difficulty, and compact/full disagreement replay. Round-level escalation tests whether full-history budget is concentrated on a minority of clini-

Method	Split	Acc	Avg.n	Prompt Tokens	TTFT (ms)	E2E Latency (ms/case)	Escalation Rate
FULL-HISTORY BASELINE	DXY dev32	**0.875**	4.3	2149.74	101.07	439.02	1.000
CASM (slots_topk_d1, margin, m=0.08)	DXY dev32	**0.906**	**4.3**	**2115.72**	**96.87**	**414.72**	**0.577**
CASM$_{\text{CONS}}$ (slots_topk_d1, margin, m=0.10)	DXY dev32	0.875	4.4	2129.42	97.11	427.89	0.709
CASM$_{\text{AGG}}$ (slots_topk, margin, m=0.05)	DXY dev32	0.812	4.1	2074.90	105.39	428.16	0.438

Table 1: Matched comparison on a DXY development slice ($N = 32$ cases). The most important contrast is between the always-full anchor and CASM: the latter improves accuracy while reducing latency and keeping full-history materialization partial rather than uniform.

cally consequential turns rather than spread diffusely across the trajectory. Difficulty buckets test whether harder consultations consume more full-history exposure than easy ones. Replay compares compact-view and full-view predictions on the same saved state to determine whether disagreement, uncertainty, and fallback align. Together these diagnostics test whether a task–latency gain comes from selective materialization rather than prompt shortening or nearly always-full routing.

These diagnostics also constrain how failure should be interpreted. If escalation is spent uniformly across rounds, or if replay reveals little compact/full disagreement before fallback, then the apparent gain would more plausibly come from prompt layout effects than from a meaningful controller. By contrast, sparse escalation concentrated on disagreement-heavy or difficult states would support the paper's core allocation view: full history is not a constant requirement, but a conditional resource spent where compact state becomes insufficient.

These analyses are not peripheral to the main table; they are where the selective-materialization claim can be tested. A controller that improves one small slice but cannot explain where the saved full-history budget went, or why those saved calls did not damage the final diagnosis, would still be hard to distinguish from prompt-order noise. The mechanism package therefore asks for accountability on both sides: where memory budget is spent, and where compression fails.

7 Related Work

Direct competition: consultation and differential diagnosis. Automatic medical consultation has been studied with dialogue systems, reinforcement learning, generation-based diagnosis, and, more recently, LLM-driven agents (Valizadeh and Parde, 2022; Chen et al., 2023b; Liu et al., 2022; Yan et al., 2023; Hou et al., 2023; Zhao et al., 2024; Tu et al., 2024; Li et al., 2024; Chen et al., 2024; Jia et al., 2025). These works mainly improve inquiry quality, diagnosis quality, or multi-agent coordination. We inherit the same task substrate but ask a different question: given an already capable consultation pipeline, how much dynamic memory should it expose on each round?

Mechanism-adjacent work: memory, uncertainty, and efficient serving. Recent work increasingly treats memory as an active design object that can be compressed, retrieved, updated, or learned as a reusable skill (Zhang et al., 2026; Liu et al., 2026; Liang et al., 2025). Our compact codec is also related to concise state descriptions for language-model actors (Nottingham et al., 2024). On the decision side, our trigger is closest in spirit to uncertainty-aware planning, abstention, calibration, and selective prediction in LLMs (Hu et al., 2024; Chen et al., 2023a; Detommaso et al., 2024; Kumar et al., 2024; Xiong et al., 2024; Liu et al., 2024; Feng et al., 2024): uncertainty matters here only if it changes behavior. On the systems side, CASM is compatible with efficient serving stacks such as vLLM and PagedAttention (Kwon et al., 2023), but does not introduce a new serving kernel or cache runtime.

Boundary line: self-evolving skills and long-horizon memory. Self-evolving agent work studies broader questions about learned memory skills, hierarchical reflection, and long-horizon skill accumulation (Yang et al., 2026; Ye et al., 2025; Yang et al., 2025; Yu et al., 2025). We cite this line to clarify scope rather than to compete head-on with it: CASM is an inference-time control policy over memory exposure, not a lifelong memory learner or a self-evolving agent architecture.

8 Discussion

This work is best read as identifying a controllable interface inside an existing consultation agent. The scientific object is the memory-exposure decision itself. If compact-only defines a real lower bound, selective fallback recovers quality

on disagreement-heavy rounds, and the controller remains nontrivially selective, then the paper contributes a concrete mechanism for allocating memory cost rather than merely another prompt variant with a favorable headline row. This scope is intentionally precise: CASM is narrower than a new consultation policy, a learned memory system, or a new serving kernel, and that narrowness keeps both gains and failures attributable to one decision variable.

That attributable scope is also why the paper keeps static disease knowledge shared across both memory views and varies only the serialization of the dynamic consultation trace. Many memory interventions simultaneously alter retrieval content, summarization policy, and decoding context, which makes failure analysis difficult: when a prediction changes, it is often unclear which component actually mattered. CASM is less ambitious, but the tradeoff is deliberate. By holding the backbone, the diagnosis objective, and the shared knowledge substrate fixed, the paper can ask a cleaner question about *when* compact state stops being enough.

9 Limitations

This paper is limited in two ways. First, the empirical evidence is strongest on a matched development slice rather than on a broader frozen test package. The manuscript therefore establishes the regime and the decision interface, but not yet a final cross-split ranking. Second, the trigger depends on the calibration quality of the compact-view uncertainty signal. If compact uncertainty is poorly aligned with the value of full-history exposure, the controller can fail in both directions, either wasting budget through unnecessary escalation or missing exactly the rounds where fallback would help.

10 Ethics Statement

This paper studies automated medical consultation as a controlled research benchmark, not as a deployable clinical product. Real-world medical deployment would require clinician oversight, safety evaluation, data governance, and regulatory review. We also caution against interpreting systems efficiency improvements as inherently desirable in clinical contexts: faster inference matters only if diagnostic quality and safety remain acceptable.

11 Conclusion

This paper treats memory exposure in multi-turn medical consultation as a resource-allocation policy rather than a fixed prompt requirement. CASM operationalizes this view with a compact state codec, uncertainty-triggered fallback, and a prefix-stable prompt layout, while leaving the base consultation pipeline intact. The contribution is a narrower intervention on *when* full history is exposed, while the consultation policy and memory learner remain unchanged. The result is straightforward: full-history materialization need not be paid uniformly across rounds, and even a lightweight deterministic controller can create a middle regime between always-full inference and overly compressed routing.

More broadly, this framing suggests a practical evaluation lens for consultation agents: memory interventions should be judged through paired task and systems outcomes, not through compression ratios alone. A useful controller is one that spends extra context on exactly the states where compact evidence becomes unreliable and stays quiet elsewhere. That criterion is intentionally modest, but it turns memory access from an uninspected prompt habit into an experimentally testable behavior.

References

Jiefeng Chen, Jinsung Yoon, Sayna Ebrahimi, Sercan Arik, Tomas Pfister, and Somesh Jha. 2023a. Adaptation with self-evaluation to improve selective prediction in llms. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5190–5213.

Junying Chen, Chi Gui, Anningzhe Gao, Ke Ji, Xidong Wang, Xiang Wan, and Benyou Wang. 2024. Cod, towards an interpretable medical agent using chain of diagnosis. *arXiv preprint arXiv:2407.13301*.

Wei Chen, Zhiwei Li, Hongyi Fang, Qianyuan Yao, Cheng Zhong, Jianye Hao, Qi Zhang, Xuanjing Huang, Jiajie Peng, and Zhongyu Wei. 2023b. A benchmark for automatic medical consultation system: frameworks, tasks and datasets. *Bioinformatics*, 39(1):btac817.

Gianluca Detommaso, Martin Bertran, Riccardo Fogliato, and Aaron Roth. 2024. Multicalibration for confidence scoring in llms. In *Proceedings of the 41st International Conference on Machine Learning*, pages 10624–10641.

Shangbin Feng, Weijia Shi, Yike Wang, Wenxuan Ding, Vidhisha Balachandran, and Yulia Tsvetkov. 2024. Don't hallucinate, abstain: Identifying llm

knowledge gaps via multi-llm collaboration. *arXiv preprint arXiv:2402.00367*.

Zhenyu Hou, Yukuo Cen, Ziding Liu, Dongxue Wu, Baoyan Wang, Xuanhe Li, Lei Hong, and Jie Tang. 2023. Mtdiag: an effective multi-task framework for automatic diagnosis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 14241–14248.

Zhiyuan Hu, Chumin Liu, Xidong Feng, Yilun Zhao, See-Kiong Ng, Anh Tuan Luu, Junxian He, Pang Wei W Koh, and Bryan Hooi. 2024. Uncertainty of thoughts: Uncertainty-aware planning enhances information seeking in llms. *Advances in Neural Information Processing Systems*, 37:24181–24215.

Zhihao Jia, Mingyi Jia, Junwen Duan, and Jianxin Wang. 2025. DDO: Dual-decision optimization for LLM-based medical consultation via multi-agent collaboration. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 26369–26386, Suzhou, China. Association for Computational Linguistics.

Abhishek Kumar, Robert Morabito, Sanzhar Umbet, Jad Kabbara, and Ali Emami. 2024. Confidence under the hood: An investigation into the confidence-probability alignment in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 315–334.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. *arXiv preprint arXiv:2309.06180*.

Shuyue Stella Li, Vidhisha Balachandran, Shangbin Feng, Jonathan S Ilgen, Emma Pierson, Pang Wei Koh, and Yulia Tsvetkov. 2024. Mediq: Question-asking llms and a benchmark for reliable interactive clinical reasoning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Jiafeng Liang, Hao Li, Chang Li, Jiaqi Zhou, Shixin Jiang, Zekun Wang, Changkai Ji, Zhihao Zhu, Runxuan Liu, Tao Ren, Jinlan Fu, and See-Kiong Ng. 2025. Ai meets brain: Memory systems from cognitive neuroscience to autonomous agents. *arXiv preprint arXiv:2512.23343*.

Hao Liu, Zi-Yi Dou, Yixin Wang, Nanyun Peng, and Yisong Yue. 2024. Uncertainty calibration for tool-using language agents. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16781–16805, Miami, Florida, USA. Association for Computational Linguistics.

Jiaqi Liu, Yaofeng Su, Peng Xia, Siwei Han, Zeyu Zheng, Cihang Xie, Mingyu Ding, and Huaxiu Yao. 2026. Simplemem: Efficient lifelong memory for llm agents. *arXiv preprint arXiv:2601.02553*.

Wenge Liu, Yi Cheng, Hao Wang, Jianheng Thang, Yafei Liu, Ruihui Zhao, Wenjie Li, Yefeng Zheng, and Xiaodan Liang. 2022. "my nose is running." "are you also coughing?": Building a medical diagnosis agent with interpretable inquiry logics. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence*, pages 4266–4272.

Kolby Nottingham, Yasaman Razeghi, Kyungmin Kim, Jb Lanier, Pierre Baldi, Roy Fox, and Sameer Singh. 2024. Selective perception: Learning concise state descriptions for language model actors. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 327–341, Mexico City, Mexico. Association for Computational Linguistics.

Tao Tu, Anil Palepu, Mike Schaekermann, Khaled Saab, Jan Freyberg, Ryutaro Tanno, Amy Wang, Brenna Li, Mohamed Amin, Nenad Tomasev, and 1 others. 2024. Towards conversational diagnostic ai. *arXiv preprint arXiv:2401.05654*.

Mina Valizadeh and Natalie Parde. 2022. The ai doctor is in: A survey of task-oriented dialogue systems for healthcare applications. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6638–6660.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2024. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. In *The Twelfth International Conference on Learning Representations*.

Lian Yan, Yi Guan, Haotian Wang, Yi Lin, and Jingchi Jiang. 2023. Efficient evidence-based dialogue system for medical diagnosis. In *2023 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 3406–3413. IEEE.

Cheng Yang, Xuemeng Yang, Licheng Wen, Daocheng Fu, Jianbiao Mei, Rong Wu, Pinlong Cai, Yufan Shen, Nianchen Deng, Botian Shi, Yu Qiao, and Haifeng Li. 2025. Learning on the job: An experience-driven self-evolving agent for long-horizon tasks. *arXiv preprint arXiv:2510.08002*.

Yutao Yang, Junsong Li, Qianjun Pan, Bihao Zhan, Yuxuan Cai, Lin Du, Jie Zhou, Kai Chen, Qin Chen, Xin Li, Bo Zhang, and Liang He. 2026. Autoskill: Experience-driven lifelong learning via skill self-evolution. *arXiv preprint arXiv:2603.01145*.

Shicheng Ye, Chao Yu, Kaiqiang Ke, Chengdong Xu, and Yinqi Wei. 2025. H^2r: Hierarchical hindsight reflection for multi-task llm agents. *arXiv preprint arXiv:2509.12810*.

Simon Yu, Gang Li, Weiyan Shi, and Peng Qi. 2025. Polyskill: Learning generalizable skills through polymorphic abstraction. *arXiv preprint arXiv:2510.15863*.

Haozhen Zhang, Quanyu Long, Jianzhu Bao, Tao Feng, Weizhi Zhang, Haodong Yue, and Wenya Wang. 2026. Memskill: Learning and evolving memory skills for self-evolving agents. *arXiv preprint arXiv:2602.02474*.

Xuehan Zhao, Jiaqi Liu, Yao Zhang, Zhiwen Yu, and Bin Guo. 2024. Haiformer: Human-ai collaboration framework for disease diagnosis via doctor-enhanced transformer. In *ECAI 2024*, pages 1495–1502. IOS Press.

A Additional Experimental Details

A.1 Reproducibility Recipe

The results reported in Table 1 use a Qwen2.5-7B-Instruct backbone, the released consultation pipeline, and the CASM inference-time controller only. The current matched slice uses the `slots_topk_d1` codec with a margin trigger, together with the released consultation limits of `max_turns=10` and `floor_turns=3`. All main-paper latency measurements are taken on one designated NVIDIA H200 GPU so that the reported task–system tradeoff is not mixed across machines. A separate shared RTX 4090 environment was used for broader ML systems reproduction and auxiliary analysis, but those runs do not enter the consultation tables in this paper.

A.2 Latency Measurement Protocol

Latency claims in this paper follow a fixed protocol:

- warm up each method before recording timings,
- run three timed repeats under the same hardware assignment,
- avoid mixing primary latency numbers across different servers,
- report prompt tokens and prefill tokens alongside latency whenever possible.

The main table reads latency through the median repeated run, with repeat statistics left to the appendix.

A.3 Sanity Alignment with the Released Base Paper

Source	Scope	Acc	Avg.n	Note
Released base-paper utility	DXY seeded reference	82.7–87.5	–	released utility lists five DXY accuracy values; split and reporting protocol are not identical to this paper
Reproduced FULL-HISTORY BASELINE	DXY dev32	0.875	4.3	matched 32-case development slice used in Table 1
CASM	DXY dev32	0.906	4.3	same slice, same latency protocol, selective fallback enabled

Table 2: Sanity alignment between the released base-paper reference and the reproduced development-slice comparison used here. This table is intended to prevent apples-to-oranges reading across different splits, seeds, and reporting conventions.

A.4 Primary Configuration Knobs

Field	Values
`memory_mode`	`full`, `compact`, `casm`
`compact_codec`	`slots`, `slots_topk`, `slots_topk_d1`
`trigger_type`	`margin`, `entropy`, `margin+entropy`
`margin_threshold`	development sweep, then frozen
`entropy_threshold`	development sweep, then frozen
`prefix_caching`	optional systems enhancement

Table 3: Primary inference-time configuration knobs exposed by CASM.

A.5 Round-Level Analysis Definitions

The appendix makes the mechanism-analysis quantities explicit. Let $R(x)$ denote the number of rounds in case x, and let $\mathbb{I}[\cdot]$ be the indicator function. We define the case-level escalation rate as

$$\mathrm{EscRate}(x) = \frac{1}{R(x)} \sum_{t=1}^{R(x)} \mathbb{I}[\texttt{escalated}_{x,t} = 1]. \tag{5}$$

At the dataset level, we report the average of $\mathrm{EscRate}(x)$ across cases.

For disagreement replay, we compare compact-view and full-view predictions on the same saved round state. Let $m_{x,t}^{\texttt{compact}}$ and $m_{x,t}^{\texttt{full}}$ denote the top-1/top-2 confidence margins under the two views. We summarize the *margin gain* of escalation as

$$\Delta m_{x,t} = m_{x,t}^{\texttt{full}} - m_{x,t}^{\texttt{compact}}. \tag{6}$$

Positive $\Delta m_{x,t}$ values indicate that full memory sharpens the diagnosis distribution, while near-zero gains suggest that escalation may have been unnecessary.

Difficulty buckets are defined using the full-history baseline as a reference model. Specifically, each case is assigned a difficulty score using the final-round margin under FULL-HISTORY BASELINE. Cases are then partitioned into easy, medium, and hard buckets by quantile. This produces a stable analysis view in which selective fallback can be interpreted relative to a consistent baseline rather than to each method's self-reported confidence.

For round-level latency analysis, let $\ell_{x,t}$ denote the latency of round t in case x. The average round latency for a given turn index r is computed over

all cases that reach that turn:

$$\mathrm{RoundLatency}(r) = \frac{\sum_x \mathbb{I}[R(x) \geq r] \cdot \ell_{x,r}}{\sum_x \mathbb{I}[R(x) \geq r]}. \tag{7}$$

This definition avoids biasing late-round measurements toward shorter cases that terminate early and makes the eventual escalation-by-round plots easier to interpret.

B Prompt Layout and Memory Serialization

B.1 Compact Memory Fields

The compact memory serializer currently exposes:

- confirmed positive symptoms,
- confirmed negative symptoms,
- queried/unqueried symptom status,
- top-k disease confidence values,
- current turn index,
- remaining consultation budget,
- optional semantic sketch features.

B.2 Full Memory Fields

The full-history view exposes the original consultation trace. Stable disease-specific knowledge remains shared prompt context rather than a variable that changes across the compact and full-history views.

B.3 Prompt Ordering Strategy

The prompt layout follows a prefix-stable order:

1. stable system and policy instructions,
2. stable disease- or task-level scaffolding,
3. dynamic compact or full memory fields,
4. round-specific query target and output slot.

This ordering keeps more tokens reusable across rounds and makes compact/full differences easier to interpret because the two views share a common prompt shell.

B.4 Negative Controls

Not every selective-fallback configuration should be presented as a success case. Any configuration with escalation rate $= 1.0$ and full-memory rate $= 1.0$ is treated as an *all-full fallback control*. At the opposite extreme, a configuration that remains almost entirely compact while suffering a clear quality collapse is treated as a *compact-insufficiency control*. Together these controls define the two failure boundaries around the target operating region.

C Case Studies and Failure Analysis

C.1 Case-Study Protocol

Case studies are not chosen for anecdotal appeal alone. Each included trajectory must satisfy one of four archetypes and carry the same evidence bundle: compact-view uncertainty, full-view uncertainty, whether escalation fired, the resulting inquiry or diagnosis difference, and the final case outcome. The four archetypes are *easy/no escalation*, *helpful escalation*, *unnecessary escalation*, and *memory-irrelevant failure*. This protocol keeps the appendix from devolving into cherry-picked narratives.

Case archetype	Inclusion rule	Required evidence	Interpretation target
Easy / no escalation	low uncertainty, no fallback, no quality loss	round trace, final outcome, compact sufficiency	show that always-full is unnecessary
Helpful escalation	compact/full disagree and fallback improves outcome	replay delta, trigger event, changed final result	show that fallback is useful rather than decorative
Unnecessary escalation	fallback fires but compact result was already adequate	trigger event, near-zero replay gain, same outcome	bound controller conservatism
Memory-irrelevant failure	both views fail on the same case	compact/full agreement, failure persists after fallback	show the limit of memory control alone

Table 4: Case-study selection protocol used in the appendix.

C.2 Failure Taxonomy

We organize failure analysis into four categories:

- **compact insufficiency**: compact memory misses information that full memory recovers;
- **trigger conservatism**: escalation happens, but the extra cost is unnecessary;
- **trigger miss**: compact memory is insufficient, but the trigger fails to escalate;
- **backbone limitation**: both compact and full memory fail, suggesting the bottleneck is not materialization alone.

This taxonomy helps prevent overclaiming. If too many errors fall into the last category, then diagnosis quality itself, rather than memory control, is the bottleneck.

C.3 Replay Reporting Schema

The disagreement-replay section uses a fixed schema rather than free-form prose. Each replay example records whether compact and full views disagree, the two confidence margins, the margin gain after escalation, whether the trigger fired,

whether fallback changed the final outcome, and whether the case is *helpful*, *neutral*, or *missed*. Without this schema, disagreement analysis risks becoming anecdotal even when the aggregate table looks promising.

C.4 Replay Outcome Labels

Label	Operational condition	Interpretation
Helpful	fallback fires and improves the final inquiry or diagnosis outcome	uncertainty exposed a genuinely useful need for more memory
Neutral	fallback fires but final outcome remains unchanged	the controller may be conservative, but not necessarily wrong
Missed	fallback does not fire on a round where replay suggests full memory would have helped	the current trigger underreacts to clinically relevant uncertainty
Irrelevant	compact and full agree, or both fail identically	the bottleneck is not the materialization choice on that round

Table 5: Replay labels used to keep disagreement analysis operational rather than impressionistic.

These labels matter because they determine how the appendix should be read. A paper with many *neutral* cases may still support the lower-bound story, but it weakens the efficiency story. A paper with many *missed* cases weakens the current trigger more directly than it weakens the overall memory-exposure framing. Separating these outcomes is therefore necessary for honest interpretation.

C.5 Minimal Case-Study Writeup Schema

Every case-study paragraph should follow the same compact schema: **context** (case difficulty and round), **compact-view state** (uncertainty and predicted action), **full-view state** (what changes after escalation), **controller decision**, and **outcome**. This keeps the appendix useful even when only a few representative cases are shown. The purpose of a case study is not to retell the whole dialogue, but to expose the smallest amount of trajectory evidence needed to justify one interpretation label.

D Logging Schema and Reproducibility Artifacts

All tables and figures in the manuscript are derived from structured artifacts rather than copied from raw logs. This keeps the paper auditable and makes it straightforward to refresh the manuscript once frozen matched-split summaries or round-level exports become available.

Field	Meaning
`case_id`, `round_id`	case and round identifiers
`dataset`, `model`	dataset and model metadata
`memory_mode`	`full`, `compact`, or `casm`
`used_full_memory`	whether full memory was materialized in the round
`escalated`	whether the round escalated from compact to full
`top1_conf`, `top2_conf`, `margin`, `entropy`	uncertainty statistics
`prompt_tokens`, `prefill_tokens`	token-level systems measurements
`ttft_ms`, `round_latency_ms`	per-round timing measurements
`acc`, `avg_n`, `case_latency_ms`	case-level summary signals

Table 6: Core logging fields used to build tables and mechanism analyses.