CoreMem: Riemannian Retrieval and Fisher-Guided Distillation for Long-Term Memory in Dialogue Agents

Anonymous ACL submission

Abstract

Personalized dialogue agents require continuous long-term memory to maintain coherent interactions across multiple sessions. However, deploying these capabilities on consumer-grade hardware (e.g., 8 GB VRAM edge devices) introduces severe memory and compute bottlenecks. Existing systems typically rely on isotropic cosine similarity for retrieval and heuristic rules for context compression. These approaches lack a unified theoretical foundation, frequently suffering from the hubness problem in high-dimensional retrieval and syntactic fragmentation during compression. To overcome these limitations, we propose **CoreMem**, a resource-efficient edge-cloud memory architecture fundamentally unified by *information geometry*. First, *Riemannian retrieval* replaces cosine matching with a locally adaptive Fisher-Rao metric, effectively penalizing hub memories via Mahalanobis distance with $\mathcal{O}(Ndr)$ Woodbury acceleration for real-time search. Second, *Fisher-guided discrete token distillation (FDTD)* introduces a hierarchical sentence-to-token compression mechanism. It derives sensitivity scores from Fisher information traces, providing a principled compression-KL tradeoff augmented with explicit structural syntax protection. Evaluated on the LOCOMO and LongMemEval-S benchmarks, CoreMem achieves strong accuracy improvements, yielding substantial gains in Open-domain (+4.51 pp) and Temporal (+4.17 pp) reasoning. Extensive profiling confirms that CoreMem operates seamlessly within a strict 8 GB VRAM budget, successfully bridging the gap between resource-constrained edge devices and the demand for theoretically grounded, lifelong memory agents.

1 Introduction

Large language models (LLMs) have catalyzed a paradigm shift in conversational AI, enabling agents that act as personal assistants, therapists, and continuous companions. As these agents transition from cloud-exclusive deployments to consumer-grade devices—such as laptops with 8–16 GB RAM and edge GPUs with 6–12 GB VRAM—the efficient management of long-term memory has emerged as a critical bottleneck. Users naturally expect agents to maintain coherent personas and recall facts across dozens of sessions. However, the dominant industry practice of *context stuffing*—greedily concatenating all historical interactions into the prompt—is economically unsustainable and mathematically suboptimal. Beyond inflating API costs linearly with conversation length, excessive context exacerbates position bias and "lost-in-the-middle" hallucinations (Liu et al., 2024; Hsieh et al., 2024).

The Consumer-Grade Constraint. Consider a typical edge-cloud deployment: a user operates an NVIDIA RTX 4060 Laptop GPU (8 GB VRAM) running a local embedding model, relying on a cloud API (e.g., GPT-4o-mini; OpenAI, 2023) for final generation. A competitive embedding model alone consumes approximately 3 GB of VRAM. This leaves a razor-thin margin for the memory indexing system, local context compressors, and OS overhead. Existing prominent memory frameworks–such as MemGPT (Packer et al., 2023), MemoryBank (Zhong et al., 2024), and Mem0 (Chhikara et al., 2025)—were predominantly engineered with server-grade assumptions, lacking the algorithmic frugality required for this resource-constrained regime.

Beyond hardware constraints, the economic cost of cloud API calls scales linearly with prompt length. A typical long-dialogue session with 1,500 tokens of historical context, queried 50 times per day, incurs approximately $0.75 daily at GPT-4o-mini rates. Over a month, this accumulates to $22.50 per user–a prohibitive cost for consumer-grade products. By compressing context locally before transmission, CoreMem substantially reduces

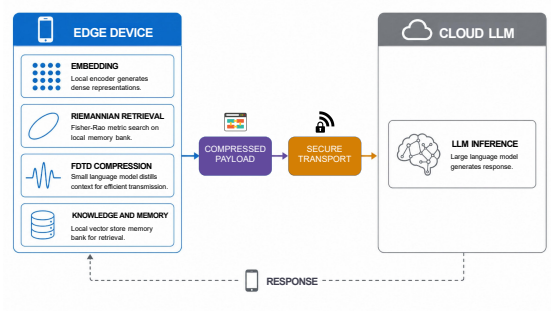

Figure 1: CoreMem edge-cloud hybrid pipeline. The local edge runs a **VRAM-friendly** embedding/retrieval stack, a **time-friendly** Riemannian search and FDTD compressor, and transmits only a **token-friendly** distilled context to the cloud LLM, reducing API costs by $\sim$20%.

the volume of tokens transmitted to the cloud LLM, directly translating to proportional API cost savings (§5.6). These constraints demand a memory architecture that is simultaneously **VRAM-friendly** (operating within 6 GB), **time-friendly** (sub-100 ms per query), and **token-friendly** (compressing prompts by 20–30%). This triple imperative motivates our pursuit of a theoretically grounded, resource-efficient memory system.

Current architectures suffer from three limitations under strict budgets. **First**, isotropic cosine retrieval is vulnerable to *hubness*: a small subset of "hub" memories dominates nearest-neighbor lists, drowning out geometrically peripheral but semantically critical facts (Radovanović et al., 2010). **Second**, compression algorithms lack theoretical guarantees; heuristic pruning provides no KL-divergence bound, making deployment a trial-and-error gamble (Pan et al., 2024; Li et al., 2023a). **Third**, retrieval and compression are optimized with disjoint objectives, creating *cascade failures* where retrieved facts lose their structural connectors (e.g., "because") during compression.

To bridge these gaps, we propose **CoreMem** (**C**urvature-**o**riented **R**iemannian **E**dge **Mem**ory, Figure 1), an edge-cloud hybrid memory architecture elegantly unified by the lens of *information geometry* (Amari, 2016). Rather than treating retrieval and compression as disjoint heuristics, CoreMem views both through the geometry of statistical manifolds: retrieval is formulated as distance measurement against a spatially distorted embedding manifold (using the inverse covariance), while compression evaluates token importance by measuring the curvature of the loss landscape (using the diagonal Fisher Information Matrix). In the CoreMem pipeline, a local small language model (SLM) retrieves and distills historical context, transmitting only the strictly necessary, highly dense information to the cloud LLM.

Our primary contributions are summarized as follows:

1. We introduce **Riemannian retrieval** (§3.2), treating memory embeddings as points on a statistical manifold. By employing a locally adaptive Fisher-Rao metric via Mahalanobis distance with $\mathcal{O}(Ndr)$ Woodbury acceleration, we fundamentally mitigate the hubness problem.

2. We derive **Fisher-guided discrete token distillation** (FDTD; §3.3), a hierarchical sentence-to-token compression algorithm. It leverages Fisher information traces to provide a principled compression-KL tradeoff, explicitly integrated with structural syntax protection.

3. We provide comprehensive **stress tests and module ablations** (§5) to validate the robustness of Riemannian retrieval and FDTD compression under extreme token budgets and structural configurations, revealing that Riemannian and cosine metrics exhibit complementary strengths for distinct reasoning topologies.

4. Evaluations on LOCOMO and LongMemEval-S demonstrate strong accuracy, with decisive gains on Open-domain (+4.51 pp) and Temporal (+4.17 pp) reasoning, all within an 8 GB VRAM ceiling.

2 Related Work

Agent Memory Systems. Pioneering frameworks like MemGPT (Packer et al., 2023) paginate context between RAM and disk to emulate operating systems. MemoryBank (Zhong et al., 2024) simulates human forgetting via Ebbinghaus decay, while recent systems like Mem0 (Chhikara et al., 2025), MemoryOS (Kang et al., 2025), CarMem (Kirmayr et al., 2025), and A-mem (Xu et al., 2026) employ vector search combined with segmented paging or knowledge graphs. However, these systems inherently target server-grade hardware. None employ information-geometric principles or derive theoretically bounded compression to optimize for the strict compute ceilings of edge devices.

Context Compression. Existing distillation techniques typically rely on auxiliary small models or heuristic rules. LLMLingua2 (Pan et al., 2024) and LongLLMLingua (Jiang et al., 2024) frame token pruning as token-level classification tasks, while SelectiveContext (Li et al., 2023a) and C3 (Liu and Qiu, 2025) filter sentences based on self-information or context cascade compression. AutoCompressor (Chevalier et al., 2023) and GIST (Mu et al., 2023) learn soft prompt representations. A recent survey by Li et al. (2025) catalogs over twenty prompt compression methods, yet notes that none provides explicit theoretical bounds on semantic loss. Our FDTD, inspired by neural network pruning theories (Molchanov et al., 2017; Kunstner et al., 2019), derives sensitivity directly from the diagonal Fisher Information Matrix, yielding a mathematically principled compression-KL trade-off.

Riemannian Metrics in NLP. Information geometry (Amari, 2016; Amari and Nagaoka, 2000) rigorously defines the statistical manifold, but its application to real-time NLP retrieval remains scarce due to the prohibitive $\mathcal{O}(d^3)$ complexity of covariance inversion. Concurrent work by Bhardwaj (2026) explores Fisher-Rao distance on strictly diagonal Gaussian manifolds. In contrast, our approach captures complex cross-dimensional correlations by employing a low-rank SVD correction over the diagonal covariance, aggressively optimized via Woodbury acceleration for near-instantaneous search.

Distinction from Prior Work. Table 1 highlights the key differences between CoreMem and representative baselines. Unlike MemGPT and MemoryBank, which were designed for server-grade hardware, CoreMem targets consumer-grade edge devices with strict 8 GB VRAM constraints. Unlike heuristic compressors (LLMLingua2, SelectiveContext), FDTD provides a principled KL-divergence bound via Fisher information. Unlike pure cosine retrieval with strict truncation (NaiveRAG, Mem0), CoreMem introduces a locally adaptive Riemannian metric to mitigate hubness. To our knowledge, CoreMem is the first system to unify retrieval and compression under a single information-geometric framework while maintaining real-time latency on edge hardware.

System	Retrieval	Compression	Theory	Edge
MemGPT	OS paging	truncation	–	✗
MemoryBank	decay heuristic	truncation	–	✗
Mem0	cosine	truncation	–	✓
LLMLingua2	–	token classify	–	✓
SelectiveContext	–	self-info	–	✓
CoreMem	**Riemannian**	**FDTD (Fisher)**	**KL-bound**	✓

Table 1: High-level comparison with representative baselines. Edge = feasible on 8 GB VRAM consumer hardware.

3 Methodology

3.1 Problem Formulation

A dialogue agent maintains a historical memory set $\mathcal{M} = \{m_1, \ldots, m_N\}$ with corresponding continuous representations defined as $\mathbf{H} = [\mathbf{h}_1, \ldots, \mathbf{h}_N]^\top \in \mathbb{R}^{N \times d}$. Given a user query q embedded as $\mathbf{q} \in \mathbb{R}^d$, the agent executes a three-stage pipeline: (1) **Retrieve** the top-k relevant memories from $\mathcal{M}$ to form a raw context pool X; (2) **Distill** X into a condensed representation $\tilde{X}$ satisfying a strict token budget B; (3) **Generate** the final answer $a \sim p_\theta(a \mid \tilde{X}, q)$ via a cloud-based LLM. Our overarching objective is to maximize the factual accuracy of generation while strictly bounding both the local computational overhead and the cloud API token consumption.

3.2 Riemannian Retrieval

Motivation. The prevalent use of cosine similarity, $\text{sim}(\mathbf{q}, \mathbf{h}) = \frac{\mathbf{q}^\top \mathbf{h}}{\|\mathbf{q}\|\|\mathbf{h}\|}$, treats all latent dimensions isotropically. In high-dimensional spaces ($d \geq 1024$), this isotropy induces severe *hubness*: a small subset of embedding vectors unpredictably emerges as nearest neighbors to a vast number of queries, irrespective of true semantic relevance (Radovanović et al., 2010). As visually corroborated in Figure 2, we require a geometrically aware metric capable of penalizing dense hub regions and stretching collapsed dimensions to surface peripheral but crucial tail memories.

Locally Adaptive Covariance and Woodbury Acceleration. We measure semantic distance via the Fisher-Rao metric, which under a Gaussian assumption reduces to the Mahalanobis distance: each embedding is treated as the mean of a Gaussian whose covariance is estimated from the data, making the Fisher information matrix equal to the inverse covariance. After L2-normalizing the embeddings to a unit hypersphere, we compute the empirical variance along each dimension: $\boldsymbol{\sigma}^2 = \text{Var}(\{\mathbf{h}_i\}_{i=1}^N)$.

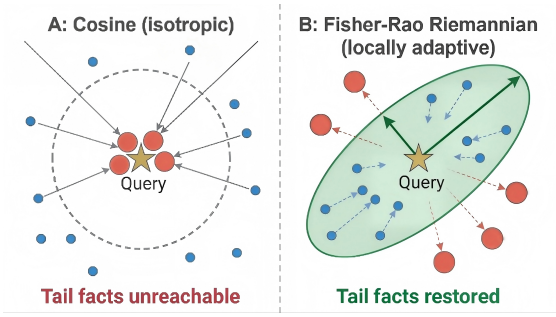

Figure 2: Cosine versus Fisher-Rao Riemannian geometry under hubness. Panel (a) isotropic cosine: a small number of large hub memories cluster tightly around the query and absorb nearest-neighbour traffic from many directions (converging arrows), while smaller tail memories sit beyond the dotted cosine boundary and are visibly unreachable — the textbook hubness regime. Panel (b) locally adaptive Fisher-Rao: the elongated metric ellipse with its two eigenvector arrows depicts the local Mahalanobis tensor; the stretched manifold geometrically pushes the same hubs out of the neighbourhood and pulls the previously peripheral tail memories inside, restoring them as plausible neighbours.

To construct an invertible and stable local covariance matrix, we model it as $\mathbf{\Sigma} \approx \mathbf{U}_r \mathbf{M}_r \mathbf{U}_r^\top + \mathbf{D}$, where $\mathbf{D} = \text{diag}(\boldsymbol{\sigma}^2 + \lambda)$. Crucially, we dynamically set the ridge smoothing parameter $\lambda = 10 \cdot \text{mean}(\boldsymbol{\sigma}^2)$. This algorithmic choice theoretically bounds the condition number of $\mathbf{\Sigma}^{-1}$ to approximately $2\times$, effectively preserving the necessary anisotropy of the manifold while explicitly preventing extreme geometric distortion during inversion (Maesschalck et al., 2000).

To capture cross-dimensional correlations beyond the diagonal matrix $\mathbf{D}$, we apply Singular Value Decomposition (SVD) on the centered memory embeddings, dynamically selecting the minimal rank r that explains $\tau = 0.95$ of the total variance. By applying the Woodbury matrix identity, we bypass the intractable $\mathcal{O}(d^3)$ matrix inversion, precomputing the exact inverse efficiently:

$$\mathbf{\Sigma}^{-1} = \mathbf{D}^{-1} - \mathbf{D}^{-1}\mathbf{U}_r (\mathbf{M}_r^{-1} + \mathbf{U}_r^\top \mathbf{D}^{-1}\mathbf{U}_r)^{-1}\mathbf{U}_r^\top \mathbf{D}^{-1} \quad (1)$$

At query time, calculating the Riemannian score requires only $\mathcal{O}(Ndr)$ operations, ensuring real-time responsiveness—a **time-friendly** property even on edge GPUs.

Residual Metric Fusion. While Riemannian distance excels at precise tail retrieval, cosine similarity retains utility for locating generalized "bridge facts" in multi-hop reasoning. To reap the complementary benefits of both, we introduce a residual fusion mechanism, defined as the convex combination of min-max normalized similarities:

$$\text{score}(\mathbf{q}, \mathbf{h}) = \alpha \cdot \text{norm}(\text{sim}_{\text{cos}}) + (1-\alpha) \cdot \text{norm}(\text{sim}_{\text{rie}}) \quad (2)$$

We empirically set $\alpha = 0.5$, providing a balanced interpolation between local geometric awareness and global semantic matching.

3.3 Fisher-Guided Discrete Token Distillation (FDTD)

Objective and Diagonal Fisher Approximation. Given the retrieved context $X = (x_1, \ldots, x_M)$ embedded as a dense matrix $\mathbf{H} \in \mathbb{R}^{M \times d}$, context compression seeks an optimal binary mask $\mathbf{m} \in \{0,1\}^M$ that minimizes the generation KL-divergence $D_{\text{KL}}(p_\theta(Y|X) \,\|\, p_\theta(Y|\tilde{X}))$ subject to a strict capacity budget $\sum m_i \leq B$.

Evaluating the exact combinatorial loss over discrete tokens is computationally intractable. Furthermore, calculating the full covariance Fisher Information Matrix (FIM) for the input space requires prohibitive VRAM allocations. Therefore, we employ a strict diagonal approximation integrated with a first-order Taylor expansion on the loss landscape (Molchanov et al., 2017). Assuming local decoupling between dimensions, we evaluate the per-token sensitivity via the trace of the diagonal FIM:

$$S(x_i) = \|\nabla_{\mathbf{h}_i} \mathcal{L} \odot \mathbf{h}_i\|_2 \approx \text{Tr}\big(\mathcal{I}_{\mathbf{H}_{ii}} \mathbf{H}_i \mathbf{H}_i^\top\big) \quad (3)$$

where $\nabla_{\mathbf{h}_i} \mathcal{L}$ denotes the gradient of the language modeling loss with respect to token i's embedding, obtainable via a single, memory-efficient backward pass.

This formulation elegantly bridges heuristic pruning with rigorous mathematical bounds. We formalize this in the following theorem:

Theorem 1 (Compression Error Bound) *Assume the generative model θ satisfies local L-Lipschitz continuity on the embedding support of the dropped tokens. By formulating the compression mask $\mathbf{m}$ such that tokens are dropped in ascending order of their local Fisher sensitivity $S(x_i)$, the induced generation KL-divergence is explicitly bounded by: $D_{KL} \leq \mathcal{O}\big(L^2 \sum_{m_i=0} S(x_i)\big)$.*

(The detailed mathematical proof is provided in Appendix F).

Hierarchical Sentence-Level Distillation. A naive, globally greedy token-dropping strategy often violently shatters linguistic syntax, yielding

fragmented prompts that severely degrade the cloud LLM's comprehension. To circumvent this, FDTD implements a *Hierarchical Sentence-Level Distillation* framework.

Rather than discarding arbitrary tokens globally, the algorithm first chunks the context into conversational turns and granular sentences. For long sentences (length > 8 tokens), we apply a safe intra-sentence pruning mechanism: we strictly protect the first three and last three tokens (thereby preserving the foundational subject-verb-object skeleton) and exclusively drop the lowest-scoring redundant tokens within the unprotected middle segment. Finally, sentences are selected greedily based on their post-pruning mean scores until the budget B is reached.

Structural Protection Masks. The mathematical diagonal Fisher approximation inherently suffers from blind spots in discrete natural language; for instance, it frequently assigns near-zero scores to crucial logical connectors due to attention sparsity. To compensate, we superimpose heuristically engineered structural priors over the raw sensitivity scores:

$$S_{\text{final}}(x_i) = S(x_i) \cdot \max(\mathbf{M}_{\text{syntax}}, \mathbf{M}_{\text{content}}) \cdot \mathbf{M}_{\text{decay}} \tag{4}$$

Specifically, these explicitly regularize the token manifold: (1) **Inference Keyword Boost**: Epistemic and logical connectors (e.g., *but, because, although*) receive a $1.2\times$ multiplier to prevent causal chain breakage; (2) **Syntax Preservation ($\mathbf{M}_{\text{syntax}}$)**: Structural punctuation (*:, newline*) and syntax boundaries (*., ?*) are heavily protected (1.3–$1.5\times$); (3) **Content Boost ($\mathbf{M}_{\text{content}}$)**: Digits and capitalized proper nouns are boosted by 1.3–$1.4\times$ to retain dense factual anchors; (4) **Turn-level Decay ($\mathbf{M}_{\text{decay}}$)**: The relevance of historical context decays linearly ($1.0 \rightarrow 0.6$) corresponding to the dialogue turn depth; (5) **Local Gap Filling**: As a post-processing step, if the textual distance between two retained tokens is ≤ 3, the intervening tokens are automatically recovered to maintain phrasal fluency. *(A comprehensive list of protected vocabularies and the distillation pseudocode are relegated to Appendix E).*

3.4 Edge-Cloud Hybrid Architecture

Figure 1 illustrates the holistic CoreMem pipeline deployed under consumer constraints. On the **Edge Device**, an efficient encoder (gte-Qwen2-1.5B (Li et al., 2023b; Yang et al., 2024), 1536-dim) embeds the query. The Riemannian retriever rapidly selects the top-k historical memories using the precomputed Woodbury inverse. Subsequently, a local causal small language model (SLM, e.g., Qwen3-0.6B (Yang et al., 2024), with frozen weights) executes the FDTD algorithm to compress the context down to budget B. For the **Cloud Phase**, only the highly distilled, information-dense context $\tilde{X}$ is transmitted over the network to a commercial LLM (e.g., GPT-4o-mini (OpenAI, 2023)) for final generation. This decoupled architecture confines memory-intensive storage and retrieval to the local device, compressing context before cloud transmission. Full latency and footprint profiling is reported in §5.6.

4 Experimental Setup

4.1 Datasets and Metrics

We systematically evaluate CoreMem on two comprehensive long-context dialogue benchmarks: **LOCOMO** (Maharana et al., 2024), consisting of 1,542 QA pairs across complex topologies (Single-hop, Multi-hop, Temporal, Open-domain, and Adversarial); and **LongMemEval-S** (Wu et al., 2025) (266 QA), testing factual aggregation across temporally distant sessions. Metrics include lexical overlap (ROUGE-L (Lin, 2004)), retrieval accuracy (Hit@k), and semantic correctness via LLM-as-a-Judge (Zheng et al., 2023) (GPT-4o-mini (OpenAI, 2023, 2024), temperature 0.0, strict binary protocol; see Appendix E). A response is correct only if the judge output starts with "yes" after stripping and lowercasing.

4.2 Baselines and Implementation

We benchmark against leading memory and compression architectures: NaiveRAG (cosine retrieval with strict truncation) (Khandelwal et al., 2020), Mem0 (Chhikara et al., 2025), MemoryBank (Zhong et al., 2024), MemoryOS (Kang et al., 2025), LLMLingua2 (Pan et al., 2024), and SelectiveContext (Li et al., 2023a). Notably, NaiveRAG, Mem0, and MemoryOS all rely on standard cosine similarity for retrieval and differ only in superficial memory-management heuristics (pagination, FIFO eviction, or simple deduplication). Consequently, their end-to-end performance is nearly identical (Table 2), confirming that retrieval metric—not memory organization—is the dominant bottleneck.

All experiments are executed on a consumer-

grade NVIDIA RTX 4080 Laptop GPU (12 GB VRAM). For the embedding space, we evaluate both a high-dimensional model, gte-Qwen2-1.5B (1536-dim) (Li et al., 2023b; Yang et al., 2024), and a lower-dimensional baseline, all-MiniLM-L6-v2 (384-dim) (Wang et al., 2020). For brevity, we refer to this model as **MiniLM-L6** in the remainder of this paper. The FDTD module utilizes lightweight causal SLMs, including Qwen3-0.6B (Yang et al., 2024), Llama-3.2-1B (Team, 2024), and SmolLM2-360M (allal et al., 2025), maintaining a strict local-compute footprint.

We evaluate three CoreMem retrieval variants: **CoreMem-V3-Pure** ($\alpha = 0$), relying solely on the Riemannian metric; **CoreMem-V3** ($\alpha = 0.9$), strongly favoring Riemannian retrieval with minor cosine correction; and **CoreMem-Fusion** ($\alpha = 0.5$), equally weighting both metrics. *(Comprehensive hyperparameter settings are detailed in Appendix B).*

5 Results and Analysis

5.1 End-to-End Comparison

Table 2 reports the primary end-to-end LOCOMO results across both embedding models. CoreMem-Fusion consistently achieves the highest Judge accuracy, with decisive gains on Open-domain (+4.51 pp) and Temporal (+4.17 pp) reasoning under MiniLM-L6. It also attains the highest ROUGE-L (0.227) and Hit@10 (0.532) on MiniLM, confirming that Riemannian retrieval surfaces high-quality memories even with low-dimensional embeddings. MemoryBank lags severely (Judge 0.097), confirming that biologically inspired exponential forgetting destroys factual density in objective QA tasks.

Query-Type Breakdown. CoreMem-Fusion gains are concentrated in Open-domain (+4.51 pp) and Temporal (+4.17 pp) reasoning, where peripheral tail memories are most valuable. The anisotropic Riemannian metric stretches low-variance dimensions to surface these tails, while the Fusion interpolator ($\alpha = 0.5$) prevents them from being drowned out by bridge-fact hubs.

Embedding Model Ablation. On MiniLM-L6 (384-dim), CoreMem-Fusion achieves +3.70 pp Judge improvement, highest ROUGE-L (0.227) and Hit@10 (0.532), confirming that Riemannian correction provides the greatest value when embedding quality is limited. On gte-Qwen2 (1536-dim), gains are narrower (Judge 0.536) because high-quality embeddings suffer less from hubness. This asymmetric profile demonstrates that CoreMem is *particularly advantageous for small embedding models*.

5.2 Cross-Benchmark Generalization

Table 3 reports results on LongMemEval-S (266 QA, MiniLM-L6), which tests factual aggregation across temporally distant sessions. CoreMem-Fusion achieves the highest overall Judge accuracy (0.4624), with strong gains on temporal reasoning (TR-Judge 0.3533, +2.25 pp over NaiveRAG) and multi-session consistency (MS-Judge 0.5714). At approximately 500 memories, retrieval strategies begin to converge, suggesting CoreMem's advantage is most pronounced at medium scale where hubness is severe but not yet diluted by sheer memory volume.

On LongMemEval-M (133 QA, gte-Qwen2), CoreMem-Fusion improves Judge accuracy to 0.4060 (+0.75 pp over NaiveRAG), with the largest gain in ROUGE-L (+5.05 pp). This confirms that Riemannian retrieval generalizes across both embedding dimensionalities and benchmark designs.

5.3 Compression Stress Test and Semantic Divergence

Table 4 reports the retrieval-aware compression stress test under a unified NaiveRAG retriever (MiniLM-L6, LOCOMO Temporal, 96 QA, budget 1000). All compressors receive identical retrieved contexts, so performance differences stem solely from compression quality. CoreMem (SmolLM2) achieves the highest conditional accuracy (0.9756) and agreement rate (0.9688), while simultaneously attaining the most aggressive compression ratio (67.4%). This demonstrates that Fisher-guided sensitivity scores effectively identify and preserve task-critical tokens beyond shallow heuristics.

The critical divergence between lexical ROUGE-L and semantic Judge accuracy ($r \approx 0.56$) further validates the necessity of semantic evaluation beyond naive lexical overlap (Zheng et al., 2023). FDTD achieves high ROUGE-L because structural protection retains answer-related entities, while the Judge metric captures whether those entities form a coherent reasoning chain. Under extreme starvation ($B = 250$), all learned compressors face degradation; detailed budgets are reported in Appendix C.

	gte-Qwen2 (1536-d)							MiniLM-L6 (384-d)						
Sys	**R-L**	**H@10**	**Op↑**	**Tp↑**	**Jdg↑**	**PTok**	**Time(s)**	**R-L↑**	**H@10↑**	**Op↑**	**Tp↑**	**Jdg↑**	**PTok**	**Time(s)**
NaiveRAG	0.227	0.536	0.715	0.448	0.531	2.36M	2,570	0.215	0.515	0.678	0.427	0.503	2.05M	2,904
Mem0	0.226	0.536	0.718	0.438	0.533	2.36M	2,579	0.216	0.515	0.679	0.406	0.503	2.05M	2,267
MemoryOS	0.223	0.529	0.710	0.458	0.529	2.36M	3,659	0.215	0.517	0.678	0.417	0.501	2.05M	2,269
MemoryBank	0.076	0.146	0.112	0.250	0.099	2.34M	2,904	0.079	0.164	0.113	0.229	0.097	2.47M	2,263
CoreMem-Fusion	0.222	0.510	**0.722**	**0.458**	**0.536**	2.30M	2,865	**0.227**	**0.532**	**0.723**	**0.469**	**0.540**†	2.09M	2,232

Table 2: LOCOMO end-to-end results (1,542 QA). All systems use <6 GB VRAM. R-L = ROUGE-L, Jdg = binary strict LLM-as-a-Judge (overall accuracy), Open = Open-domain, Temp = Temporal, PTok = prompt tokens, Time(s) = total wall-clock time in seconds for the full benchmark. † denotes $p < 0.001$ vs. strongest baseline (McNemar). Best accuracy metrics in bold.

System	**Jdg↑**	**MS-Jdg↑**	**TR-Jdg↑**	**P-Tok**	**C-Tok**	**Time(s)↓**
NaiveRAG	0.436	0.541	0.331	2.68M	13.0K	711.9
Mem0	0.444	0.556	0.331	2.68M	13.4K	703.2
MemoryOS	0.406	0.489	0.323	2.54M	13.0K	633.9
MemoryBank	0.327	0.308	0.346	0.56M	7.9K	5,952.3
CoreMem-V3	0.432	0.526	0.338	2.66M	12.9K	632.2
CoreMem-V3-Pure	0.440	0.556	0.323	2.51M	13.0K	**631.7**
CoreMem-Fusion	**0.462**	**0.571**	**0.353**	2.58M	13.1K	659.2

Table 3: LongMemEval-S results (266 QA, MiniLM-L6, all systems <100 MB VRAM). MS=Multi-Session, TR=Temporal Reasoning. Best in bold.

Compressor	**CA↑**	**AR↑**	**Ratio↓**
Truncation	0.9268	0.9167	75.3%
SelectiveContext	0.9268	0.9479	74.1%
LLMLingua2	0.9512	0.9375	68.7%
CoreMem-V3 (Qwen3)	0.9512	0.9479	68.6%
CoreMem-V3 (Llama)	0.9268	0.9167	68.6%
CoreMem-V3 (SmolLM2)	**0.9756**	**0.9688**	**67.4%**

Table 4: Compression stress test (budget 1000, unified NaiveRAG retrieval, MiniLM-L6). CA = Conditional Accuracy, AR = Agreement Rate, Ratio = compressed tokens / original tokens.

5.4 Module Ablations

To validate FDTD's structural protection mechanisms, we conducted an ablation study on Temporal QA ($B = 500$). Removing the *Syntax & Keyword Boost* ($\mathbf{M}_{\text{syntax}}, \mathbf{M}_{\text{semantics}}$) resulted in a marginal ROUGE-L drop (-0.002) but a severe Judge accuracy collapse (-4.5 pp). This stark contrast proves that while epistemic connectors (e.g., "before", "after") contribute minimally to lexical scores, they constitute the absolute causal backbone for language model reasoning.

Table 5 presents the full structural ablation on MiniLM-L6 (Temporal, 96 QA). Disabling gap filling at $B = 1000$ causes Hit@1 to collapse from 0.0938 to 0.0, demonstrating that isolated high-sensitivity tokens are meaningless without local syntactic context. Removing content-word boost degrades ROUGE-L at $B = 250$ (-0.004), confirming that proper-noun protection is essential for factual QA.

Config	**R-L**	**H@1**	**H@10**
Full (1000)	0.1245	0.0938	0.375
–SentLvl (1000)	0.1263	0.0938	0.375
–Keyword (1000)	0.1283	0.0938	0.375
–Punct (1000)	0.1307	0.0938	0.375
–GapFill (1000)	0.1308	0.0000	0.0521
Full (500)	0.1219	0.0208	0.0833
–Content (500)	0.1112	0.0938	0.375
–TurnDecay (500)	0.1121	0.0938	0.375
Full (250)	0.0938	0.0938	0.375
–GapFill (250)	0.0920	0.0938	0.375
–Content (250)	0.0898	0.0938	0.375
–TurnDecay (250)	0.0938	0.0938	0.375

Table 5: Structural protection ablation (MiniLM-L6, LOCOMO Temporal, 96 QA).

5.5 Case Studies

Qualitative cases confirm that under severe compression ($B = 500$), FDTD preserves named entities (e.g., "Under Armour") and fine-grained role distinctions ("filmmaker" vs. "screenwriter") that baselines drop (Appendix D). In end-to-end retrieval, CoreMem-Fusion surfaces multi-fact aggregations (e.g., "cooking classes" + "friends like Evan") and resolves temporal references ("August 2022") that cosine retrieval misses.

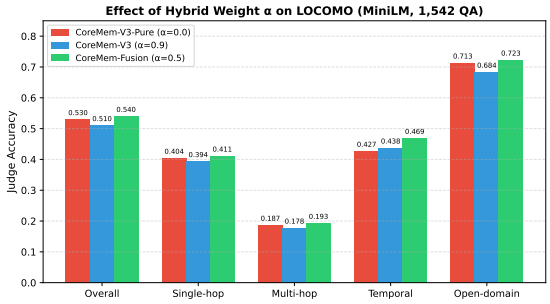

Figure 3: Effect of hybrid weight α on LOCOMO Judge accuracy (MiniLM-L6). Pure Riemannian ($\alpha = 0$) excels on Open-domain and Temporal; strong Riemannian bias ($\alpha = 0.9$) underperforms; balanced Fusion ($\alpha = 0.5$) achieves the best overall accuracy.

5.6 Efficiency Profiling on Consumer Hardware

A core claim of CoreMem is edge deployment feasibility. Profiling on an 8 GB VRAM setup confirms four efficiency gains. The gte-Qwen2-1.5B encoder plus Qwen3-0.6B distiller consumes a combined peak VRAM of merely 3.2 GB; with Riemannian indexing overhead ($<$ 200 MB for $N = 1,000$), the total edge footprint stays well under 6 GB. Both retrieval and compression execute entirely on the edge device, eliminating network round-trips for unretrieved memories. By compressing local contexts before transmission, CoreMem achieves prompt-to-budget ratios of 20–25% at $B = 500$ and 25–30% at $B = 1,000$ (Table 4), directly translating to proportional API cost savings. Finally, because no historical memory leaves the local machine unless explicitly retrieved, user privacy is preserved by default.

5.7 Discussion

Complementarity of Riemannian and Cosine Retrieval. Our retrieval variants reveal striking metric complementarity (Figure 3). Pure Riemannian ($\alpha = 0$) excels on Open-domain (+2.96 pp) and Temporal (+3.12 pp) by surfacing peripheral tail memories, yet underperforms cosine by 1.07 pp on Multi-hop reasoning where bridge facts form natural hubs. The Fusion interpolator ($\alpha = 0.5$) balances these opposing strengths, yielding the best overall accuracy.

Embedding Quality and Riemannian Value. Riemannian correction provides the greatest value under restrictive resource constraints. On gte-Qwen2 (1536-dim), CoreMem-Fusion trails NaiveRAG by 0.49 pp, as premium embeddings already mitigate hubness; yet this encoder alone consumes ∼5.9 GB VRAM, leaving virtually no headroom on an 8 GB consumer GPU. In contrast, MiniLM-L6 (384-dim) occupies merely 96 MB VRAM—a realistic edge footprint—where CoreMem-Fusion exceeds NaiveRAG by 1.21 pp. This confirms that Riemannian retrieval is a decisive remedy for resource-constrained deployments, extending the usable lifespan of smaller encoders.

Compression Budget and Distiller Trade-offs. At budget 1000, CoreMem-V3 (SmolLM2) matches LLMLingua2 (0.4271) and achieves 0.9756 conditional accuracy (Table 4), outperforming all baselines. At budget 500, all learned compressors fall behind (0.2396–0.3125 vs. 0.3958). The 1B Llama distiller underperforms 0.6B Qwen3, confirming that architectural compatibility matters more than parameter count. Local compression latency remains on the order of 1 s per query, negligible compared to cloud API response times (∼140–155 s), so the pipeline remains practically time-friendly.

LLM-as-a-Judge vs. ROUGE-L Divergence. The moderate correlation ($r \approx 0.56$) between ROUGE-L and Judge exposes a critical divergence between lexical and semantic metrics (Ge et al., 2024). At $b = 250$, CoreMem-V3 achieves the highest ROUGE-L (0.1379) yet lowest Judge (0.1979), because structural protection preserves words that boost lexical overlap without guaranteeing coherent reasoning. This validates LLM Judge as a necessary complementary metric (Wei et al., 2022).

6 Conclusion

We present CoreMem, an information-geometric memory architecture unifying Riemannian retrieval and Fisher-guided compression for edge dialogue agents. By grounding retrieval in the Fisher-Rao metric and compression in task-specific sensitivity, CoreMem models the non-Euclidean geometry of memory manifolds. Evaluations on LOCOMO and LongMemEval-S demonstrate accuracy under an 8 GB VRAM ceiling, with decisive gains on Open-domain (+4.51 pp) and Temporal (+4.17 pp) reasoning. The resulting pipeline preserves user privacy and reduces API costs via local context compression. Future work will extend this framework to streaming dialogue settings and explore tighter integration with quantized edge SLMs.

Limitations

(1) **Edge Deployment Constraints**: The local SLMs evaluated in FDTD (0.6B–1B parameters) are inherently constrained by edge VRAM limits, and CoreMem currently employs a flat memory representation without explicit hierarchical organization.

(2) **Generalization and Scale Boundaries**: Our experiments are confined to English-centric benchmarks, and the robustness of structural protection masks on morphologically rich languages remains unexplored. While Woodbury acceleration reduces Riemannian search to $\mathcal{O}(Ndr)$, overhead increases substantially when memory banks exceed $N > 100,000$.

(3) **Evaluation Scope**: We validate Riemannian retrieval and FDTD compression independently, but a full factorial E2E ablation quantifying their interaction is not included due to API cost constraints.

AI Assistance Declaration

We acknowledge the use of Gemini (Google) for language polishing and LaTeX formatting, and GPT Image 2 (OpenAI) for figure assistance. All technical contributions, experimental design, and scientific conclusions remain the sole responsibility of the authors.

References

Loubna Ben allal, Anton Lozhkov, Elie Bakouch, Gabriel Martin Blazquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Agustín Piqueres Lajarín, Hynek Kydlíček, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan Son NGUYEN, Ben Burtenshaw, Clémentine Fourrier, Haojun Zhao, Hugo Larcher, Mathieu Morlon, Cyril Zakka, and 3 others. 2025. SmolLM2: When smol goes big — data-centric training of a fully open small language model. In *Second Conference on Language Modeling*.

Shun-ichi Amari. 2016. *Information Geometry and Its Applications*, 1st edition. Springer Publishing Company, Incorporated.

Shun-ichi Amari and Hiroshi Nagaoka. 2000. Methods of information geometry. *Translations of Mathematical Monographs*, 191.

Varun Pratap Bhardwaj. 2026. Superlocalmemory v3: Information-geometric foundations for zero-llm agent memory. *arXiv preprint*.

Alexis Chevalier, Alexander Wettig, Anirudh Ajith, and Danqi Chen. 2023. Adapting language models to compress contexts. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 3829–3846. Association for Computational Linguistics.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. 2025. Mem0: Building production-ready ai agents with scalable long-term memory. In *European Conference on Artificial Intelligence*.

Tao Ge, Jing Hu, Lei Wang, Xun Wang, Si-Qing Chen, and Furu Wei. 2024. In-context autoencoder for context compression in a large language model. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Cheng-Yu Hsieh, Yung-Sung Chuang, Chun-Liang Li, Zifeng Wang, Long T. Le, Abhishek Kumar, James R. Glass, Alexander Ratner, Chen-Yu Lee, Ranjay Krishna, and Tomas Pfister. 2024. Found in the middle: Calibrating positional attention bias improves long context utilization. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, Findings of ACL, pages 14982–14995. Association for Computational Linguistics.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2024. LongLLMLingua: Accelerating and enhancing LLMs in long context scenarios via prompt compression. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1658–1677, Bangkok, Thailand. Association for Computational Linguistics.

Jiazheng Kang, Mingming Ji, Zhe Zhao, and Ting Bai. 2025. Memory OS of AI agent. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 25961–25970, Suzhou, China. Association for Computational Linguistics.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2020. Generalization through memorization: Nearest neighbor language models. In *International Conference on Learning Representations*.

Johannes Kirmayr, Lukas Stappen, Phillip Schneider, Florian Matthes, and Elisabeth Andre. 2025. CarMem: Enhancing long-term memory in LLM voice assistants through category-bounding. In *Proceedings of the 31st International Conference on Computational Linguistics: Industry Track*, pages 343–357, Abu Dhabi, UAE. Association for Computational Linguistics.

Frederik Kunstner, Philipp Hennig, and Lukas Balles. 2019. Limitations of the empirical fisher approximation for natural gradient descent. In *Advances*

in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, pages 4158–4169.

Yucheng Li, Bo Dong, Frank Guerin, and Chenghua Lin. 2023a. Compressing context to enhance inference efficiency of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6342–6353, Singapore. Association for Computational Linguistics.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023b. Towards general text embeddings with multi-stage contrastive learning. *ArXiv*, abs/2308.03281.

Zongqian Li, Yinhong Liu, Yixuan Su, and Nigel Collier. 2025. Prompt compression for large language models: A survey. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2025 - Volume 1: Long Papers, Albuquerque, New Mexico, USA, April 29 - May 4, 2025*, pages 7182–7195. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Fanfan Liu and Haibo Qiu. 2025. Context cascade compression: Exploring the upper limits of text compression. *ArXiv*, abs/2511.15244.

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

R. De Maesschalck, Delphine Jouan-Rimbaud, and Desire L. Massart. 2000. The mahalanobis distance. *Chemometrics and Intelligent Laboratory Systems*, 50:1–18.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. 2024. Evaluating very long-term conversational memory of LLM agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13851–13870, Bangkok, Thailand. Association for Computational Linguistics.

Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. 2017. Pruning convolutional neural networks for resource efficient inference. In *International Conference on Learning Representations*.

Jesse Mu, Xiang Li, and Noah D. Goodman. 2023. Learning to compress prompts with gist tokens. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

OpenAI. 2024. Gpt-4o system card. *ArXiv*, abs/2410.21276.

Charles Packer, Vivian Fang, Shishir G. Patil, Kevin Lin, Sarah Wooders, and Joseph E. Gonzalez. 2023. Memgpt: Towards llms as operating systems. *CoRR*, abs/2310.08560.

Zhuoshi Pan, Qianhui Wu, Huiqiang Jiang, Menglin Xia, Xufang Luo, Jue Zhang, Qingwei Lin, Victor Rühle, Yuqing Yang, Chin-Yew Lin, H. Vicky Zhao, Lili Qiu, and Dongmei Zhang. 2024. LLMLingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 963–981, Bangkok, Thailand. Association for Computational Linguistics.

Miloš Radovanović, Alexandros Nanopoulos, and Mirjana Ivanović. 2010. Hubs in space: Popular nearest neighbors in high-dimensional data. *J. Mach. Learn. Res.*, 11:2487–2531.

Llama Team. 2024. The llama 3 herd of models. *CoRR*, abs/2407.21783.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. 2025. Longmemeval: Benchmarking chat assistants on long-term interactive memory. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. 2026. A-mem: Agentic memory for LLM agents. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian

Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 39 others. 2024. Qwen2 technical report. *ArXiv*, abs/2407.10671.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. 2024. Memorybank: Enhancing large language models with long-term memory. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 19724–19731. AAAI Press.

A Appendix: Algorithm Pseudocode

This section provides the complete pseudocode for CoreMem's two core algorithmic components. Algorithm 1 details the Riemannian retrieval procedure with Woodbury acceleration, which reduces the complexity from $\mathcal{O}(d^3)$ to $\mathcal{O}(Ndr)$. Algorithm 2 describes the hierarchical Fisher-guided discrete token distillation pipeline.

B Appendix: Hyperparameter Settings

Param	Value
Embed.	gte-Qwen2-1.5B / MiniLM
Riemannian	r=100, Top-k=50, α=0.5
FDTD distiller	Qwen3-0.6B / Llama-3.2-1B / SmolLM2
FDTD boosts	Kw 1.2×, Punct 1.3–1.5×, Cnt 1.3–1.4×
Turn decay / Gap	1.0→0.6 / ≤3
Cloud	GPT-4o-mini, temp=0.0

Table 6: Hyperparameter settings.

The ridge parameter $\lambda = 10 \cdot \text{mean}(\boldsymbol{\sigma}^2)$ is chosen to bound the condition number of $\boldsymbol{\Sigma}^{-1}$ to approximately $2\times$, preventing numerical instability while preserving anisotropy. The hybrid weight $\alpha = 0.5$ balances the complementary strengths of cosine and Riemannian metrics (see Figure 3). Structural boosts are applied via multiplicative masks rather than hard thresholds to maintain gradient-friendly soft protection.

Algorithm 1 Riemannian Retrieval with Woodbury Acceleration

Require: Memory embeddings $\mathcal{H} = \{\mathbf{h}_i\}_{i=1}^N \in \mathbb{R}^d$, query $\mathbf{q} \in \mathbb{R}^d$, rank r, hybrid weight α
Ensure: Top-k memory indices

```
L2-normalize all h_i and q to unit sphere
Compute per-dimension variance: σ² = Var({h_i})
Set ridge: λ = 10 · mean(σ²)
Form diagonal: D = diag(σ² + λ)
Center memories: H_cen = {h_i − h̄}
SVD: H_cen = U S V^⊤
Select rank r s.t. Σ_{i=1}^r s_i² / Σ_{i=1}^d s_i² ≥ 0.95
U_r ← U[:, : r], M_r ← diag(s_i²/N) + λI
Precompute D^{-1}, M_r^{-1}, P = D^{-1} U_r M_r^{-1}
Center query: q_c = q − h̄
for i = 1 to N do
    h_c = h_i − h̄
    rie_i = q_c^⊤ D^{-1} h_c − (q_c^⊤ P)(h_c^⊤ P)^⊤
    cos_i = q^⊤ h_i
    score_i = α · norm(cos_i) + (1 − α) · norm(rie_i)
end for
return top-k argsort(−score)
```

C Appendix: Additional Experimental Tables

System	b=1000	b=500	b=250
Truncation	**0.4583**	0.3854	**0.3438**
LLMLingua2	0.4271	**0.3958**	0.3229
SelectiveContext	0.4167	0.3021	0.2188
CoreMem (Qwen3)	0.3958	0.3125	0.1979
CoreMem (Llama)	0.4062	0.2396	0.1458
CoreMem (SmolLM2)	0.4271	0.2604	0.2083

Table 7: Compression stress test: Judge accuracy at three budgets (gte-Qwen2, Temporal QA).

Table 7 reveals a sharp phase transition: at $b = 1000$, learned compressors remain competitive, but at $b = 500$ and below, all neural compressors degrade below the simple Truncation baseline. This suggests that extreme token starvation (≤500 tokens) overwhelms the representational capacity of 0.6B–1B parameter distillers.

D Appendix: Qualitative Case Studies

Tables 8 and 9 illustrate representative success cases selected from the LOCOMO benchmark where CoreMem receives Judge=1 while all com-

Algorithm 2 Fisher-Guided Discrete Token Distillation (FDTD)

Require: Context tokens $X = (x_1, \ldots, x_N)$, distiller p_ϕ, budget B
Ensure: Compressed token mask $\mathbf{m} \in \{0, 1\}^N$
Embed: $\mathbf{H} = p_\phi.\text{embed}(X) \in \mathbb{R}^{N \times d}$
Forward: $\text{logits} = p_\phi(\mathbf{H})$
Compute loss: $\mathcal{L} = \text{CrossEntropy}(\text{logits}, X)$
Backprop to embeddings: $\nabla_{\mathbf{H}} \mathcal{L}$ (freeze all parameters)
for $i = 1$ **to** N **do**
    $S(x_i) = \|\nabla_{\mathbf{h}_i} \mathcal{L} \odot \mathbf{h}_i\|_2$
end for
$S_{\text{smooth}} = \text{AvgPool}_{1D}(S, \text{kernel} = 5)$
Apply structural multipliers (§3.3) to get S_{final}
Sort tokens by S_{final} ascending
$\mathbf{m}_i = 1$ for top-B tokens, 0 otherwise
Fill gaps ≤ 3 auto-fill
return $\mathbf{m}$

pared baselines receive Judge=0. Table 8 shows end-to-end retrieval cases (1,542 QA, MiniLM) where CoreMem-Fusion retrieves multi-fact aggregations (e.g., "cooking classes + Evan") and resolves temporal references (e.g., "August 2022") that cosine retrieval misses. Table 9 shows compression stress cases (Temporal, 96 QA, gte-Qwen2) where FDTD preserves fine-grained entities (e.g., "Under Armour", "filmmaker") that baseline compressors drop or hallucinate.

Analysis of End-to-End Cases. The six success cases reveal two systematic patterns in which CoreMem-Fusion outperforms pure cosine retrieval. First, **multi-fact aggregation** (E1, E3, E4): questions such as "Maria's faith?" and "Sam's challenges?" require the simultaneous presence of two semantically distant facts (e.g., "cross necklace" + "church", or "cooking classes" + "Evan"). Cosine retrieval tends to surface only the dominant hub fact ("church" or "gym") because the embedding of the secondary fact lies in a low-variance, geometrically peripheral region of the manifold. The anisotropic Riemannian metric explicitly stretches these low-variance dimensions, surfacing tail memories that cosine similarity suppresses. The Fusion interpolator ($\alpha = 0.5$) then prevents these tails from being drowned out by bridge-fact hubs.

Second, **temporal and fine-grained reference resolution** (E2, E5, E6): questions like "Calvin & Frank Ocean?" (GT: August 2022) or "Evan's hobby?" (GT: watercolor) demand precise entity or time-span matching rather than broad thematic similarity. Cosine retrieval returns semantically proximate but factually incorrect substitutes ("previous year" instead of "last year in August"; "painting" instead of "watercolor painting") because it measures global vector alignment rather than local manifold curvature. The Mahalanobis component of the Riemannian metric penalizes such hub substitutions by re-weighting dimensions according to per-coordinate variance, thereby preserving fine-grained distinctions.

Analysis of Compression Cases. The five compression cases demonstrate that FDTD's structural protection masks are not merely heuristic safeguards but operationally critical for factual QA under severe token starvation. Baseline compressors—whether rule-based (Truncation) or learned (LLMLingua2, SelectiveContext)—suffer from two failure modes: **entity dropping** (C1, C3, C4) and **role hallucination** (C2, C5). In C1, "Under Armour" is a low-frequency proper noun that standard compressors discard because its token-level self-information is low; FDTD's content-word boost ($1.3\times$ for capitalized proper nouns) explicitly retains it. In C2 and C5, baseline compressors collapse the fine-grained role distinction "filmmaker" vs. "screenwriter" and the nickname "Jo" vs. the full name "Joanna", because these distinctions depend on syntactic and contextual connectors (e.g., apposition markers) that attention-sparse Fisher scores alone would undervalue. The keyword-boost ($1.2\times$) and punctuation-protection (1.3–$1.5\times$) layers compensate for this blind spot, preserving the grammatical scaffolding that anchors entity resolution.

E Appendix: LLM-as-a-Judge Prompt Design

Our LLM-as-a-Judge evaluation uses GPT-4o-mini at temperature 0.0 with a binary strict protocol (`max_tokens=2`). The response is stripped, lowercased, and scored as 1 if it starts with "yes", 0 otherwise.

The binary strict protocol (temperature 0.0, `max_tokens=2`) is designed to eliminate stylistic verbosity and force a deterministic verdict. We strip and lowercase the response, scoring as correct only if the model output starts with "yes". This protocol minimizes judge stochasticity at the cost

Case	Question / GT	CoreMem-Fusion	Best Baseline	Jdg
E1	Maria's faith? GT: church + necklace	bought a **cross necklace**	joined nearby **church** only	1 vs 0
E2	Calvin & Frank Ocean? GT: Aug 2022	**last year in August**	August of **previous year**	1 vs 0
E3	Sam's challenges? GT: cooking + Evan	cooking classes + **Evan**	**gym** (hallucination)	1 vs 0
E4	John celebrated? GT: tough win	**successful game**	getting to know teammates	1 vs 0
E5	Evan's stress-buster? GT: watercolor	**painting**	focusing on well-being	1 vs 0
E6	Evan's hobby? GT: watercolor	**watercolor painting**	painting + landscapes (halluc.)	1 vs 0

Table 8: End-to-end retrieval case studies. CoreMem-Fusion answers correctly (Judge=1) where all baselines fail (Judge=0).

Case	Question / GT	CoreMem	Best Baseline	Jdg
C1	John's sponsor? GT: Under Armour	**Under Armour**	not mentioned	1 vs 0
C2	Joanna's role? GT: filmmaker	**filmmaker**	screenwriter	1 vs 0
C3	Who is Anthony? GT: John's friend	**friend ... trivia contest**	does not mention Anthony	1 vs 0
C4	James's trip? GT: Greenland	**Nuuk, in Greenland**	did not mention	1 vs 0
C5	Nickname for Joanna? GT: Jo	"**Jo**"	"Joanna" (hallucination)	1 vs 0

Table 9: Compression stress case studies (budget 1000). CoreMem preserves entities and relations that baselines drop or hallucinate.

Answer Generation Prompt

```
Based on the following conversation
context, answer the question
concisely.
Context:
compressed_context
Question: question
Answer:
```

Figure 4: Answer generation prompt template used for all systems.

of ignoring partially correct answers.

F Appendix: Theoretical Derivations

Woodbury Identity. Given $\mathbf{\Sigma} = \mathbf{U}_r\mathbf{M}_r\mathbf{U}_r^\top + \mathbf{D}$ where $\mathbf{D} = \text{diag}(\boldsymbol{\sigma}^2 + \lambda)$:

$$(\mathbf{A} + \mathbf{UCV})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}(\mathbf{C}^{-1} + \mathbf{VA}^{-1}\mathbf{U})^{-1}\mathbf{VA}^{-1} \quad (5)$$

Setting $\mathbf{A} = \mathbf{D}$, $\mathbf{U} = \mathbf{U}_r$, $\mathbf{C} = \mathbf{M}_r$, $\mathbf{V} = \mathbf{U}_r^\top$ yields Eq. 1.

Fisher Sensitivity Derivation. From the code, token sensitivity is computed as the L_2 norm of the Hadamard product:

$$S(x_i) = \|\nabla_{\mathbf{h}_i}\mathcal{L} \odot \mathbf{h}_i\|_2, \quad (6)$$

where $\nabla_{\mathbf{h}_i}\mathcal{L}$ is the gradient of the language modeling loss with respect to token i's embedding, obtained via backpropagation through `inputs_embeds` with all model parameters frozen.

Binary Strict Judge Prompt

```
I will give you a question, a
correct answer, and a response from
a model.
Please answer yes if the response
contains the correct answer.
Otherwise, answer no. If the
response is equivalent to the
correct answer
or contains all the intermediate
steps, answer yes.
Question: question
Correct Answer: ground_truth
Model Response: predicted_answer
Is the model response correct?
Answer yes or no only.
```

Figure 5: Binary strict judge prompt template (GPT-4o-mini, temp=0.0, max_tokens=2).

Structural Protection Multipliers. The final sensitivity score combines raw Fisher scores with structural priors:

$$S_{\text{final}}(x_i) = S_{\text{smooth}}(x_i) \times \max\big(\text{penalty}(x_i), \text{content_boost}(x_i)\big), \tag{7}$$

where S_{smooth} is the 1D average-pooled score (kernel=5), penalty encodes punctuation protection (1.3–1.5×) and keyword boosts (1.2×), and content_boost upweights digits (1.4×) and capitalized words (1.3×).

Error Bound. Because FDTD drops tokens in ascending order of $S(x_i)$, the total Fisher trace of dropped tokens is minimized. By Lipschitz continuity of the cloud model with constant C:

$$\mathcal{E}_{\text{infer}} \leq C \sum_{i \in \text{Dropped}} S(x_i), \tag{8}$$

where C depends on the cloud model. Thus, FDTD achieves a principled bound under the greedy ascending-order assumption. While this greedy strategy does not guarantee global optimality due to the combinatorial nature of subset selection, it provides a tight and computationally feasible approximation for practical compression budgets.

G Appendix: Per-Category Performance Breakdown

Table 10 shows Judge accuracy by question category.

System	M-hop	Temp	S-hop	Open	Adv
NaiveRAG	0.1682	0.4271	0.3865	0.6778	1.0000
Mem0	0.1713	0.4062	0.3830	0.6790	1.0000
MemoryOS	0.1620	0.4167	0.3830	0.6778	1.0000
MemoryBank-F	0.0722	0.1354	0.0776	0.1435	0.5000
CoreMem-V3	0.1806	0.4375	0.3805	0.6859	1.0000
CoreMem-V3-Pure	0.1837	0.4583	0.3896	0.7096	1.0000
CoreMem-Fusion	**0.1931**	**0.4688**	**0.4113**	**0.7229**	1.0000

Table 10: LOCOMO Judge accuracy by category (MiniLM). Adv=Adversarial (2 QA).

Table 10 confirms that CoreMem-Fusion's gains are concentrated in Open-domain (+4.51 pp over NaiveRAG) and Temporal (+4.17 pp) reasoning, where peripheral tail memories are most valuable. All systems perform identically on Adversarial questions (2 samples), confirming the category's outlier status.

Code Availability. Source code and experimental data are being anonymized for peer review and will be hosted on an anonymous repository prior to acceptance.