

Dual-System Memory Consolidation for Lifelong Learning in Language Models: Combining Direct Weight Editing with Sleep-Wake Training

Vladimir Baranov
vlad@chatsetter.ai

Abstract

Large language models lack persistent memory: each session begins from a blank state, and all conversational context is lost when the session ends. We present a dual-system architecture for lifelong learning that combines two complementary mechanisms: MEMIT (Mass-Editing Memory in a Transformer) for instant factual injection during the wake phase, and LoRA fine-tuning during offline sleep cycles for permanent consolidation. Drawing on Complementary Learning Systems theory from neuroscience, MEMIT serves as a hippocampal analog—fast, high-fidelity encoding that is fragile under continued use—while LoRA sleep consolidation serves as a neocortical analog—slow, stable integration that survives model restarts. A nap mechanism acts as a selective consolidation filter, transferring strongly-encoded MEMIT facts into durable LoRA weights while allowing weakly-encoded traces to decay—mirroring the triage function of biological sleep. We introduce covariance-regularized MEMIT with cross-edit null-space constraints, which preserves previously injected facts when new facts are added. We validate the system across three model scales (3B, 8B, 70B parameters) on consumer and datacenter hardware. Ablation experiments demonstrate that the dual system outperforms either component alone: MEMIT provides instant recall (5/5 facts, <5s) unavailable to LoRA-only systems (0/5 immediate recall), while sleep consolidation provides persistence (4/5 post-restart) unavailable to MEMIT-only systems. On the 8B model, MEMIT sustains 0.83 recall at 60 facts with near-zero perplexity impact ($\Delta PPL < 0.03$). Cross-edit null-space constraints achieve perfect retention (1.00) of previously injected facts through both filler conversations and new injections. The full lifecycle—wake, nap, and deep sleep—completes successfully on all three model sizes with validation scores of 5/5.

1 Introduction

Every modern large language model suffers from a fundamental limitation: it cannot learn from its own conversations. A user may spend hours sharing personal details, establishing preferences, and building context, but the moment the session ends, all of it vanishes. The context window provides an illusion of memory during a session, but this is working memory with a hard size limit that disappears between sessions and provides no mechanism for actual learning from experience.

Existing approaches keep the model’s weights frozen. Retrieval-augmented generation (RAG) stores conversation snippets in an external database and injects relevant ones into the prompt at inference time (Lewis et al., 2020). Memory-augmented architectures add external read-write modules (Wang et al., 2024, 2023a). These treat memory as an external resource accessed through the input—the model itself never changes.

In prior work (Baranov, 2026), we demonstrated that a local LLM can form persistent memories through biologically-inspired sleep-wake cycles using LoRA fine-tuning. That system successfully transferred conversational facts into model weights on a 3B model, but suffered from a critical limitation: facts were only available *after* a sleep cycle completed, creating a delay between learning and recall that ranged from minutes to hours.

This paper introduces a **dual-system architecture** that eliminates this delay by combining two complementary weight-modification mechanisms:

1. **MEMIT** (Mass-Editing Memory in a Transformer) (Meng et al., 2023) provides instant factual recall during the wake phase by directly editing MLP weights. Facts are available immediately after injection—no training required.
2. **LoRA sleep consolidation** provides permanent storage by training the model on MEMIT-held facts during offline nap and sleep cycles, then reverting the fragile MEMIT edits.

This design maps directly onto Complementary Learning Systems (CLS) theory (McClelland et al., 1995; Kumaran et al., 2016): MEMIT functions as the hippocampus (fast, high-fidelity episodic encoding that degrades under interference), while LoRA consolidation functions as the neocortex (slow, stable integration that generalizes across contexts). The nap mechanism—a lightweight consolidation cycle that transfers MEMIT facts to LoRA—corresponds to hippocampal replay during NREM sleep (Diekelmann and Born, 2010; Rasch and Born, 2013).

We make four contributions:

1. **A dual-system MEMIT+LoRA architecture with nap consolidation** that provides both instant and permanent memory through complementary weight-editing pathways.
2. **Covariance-regularized MEMIT with cross-edit null-space constraints** that preserves previously injected facts when new facts are added, using the Woodbury identity (Woodbury, 1950) for efficient regularization.
3. **Quantitative scaling analysis across 3B, 8B, and 70B parameters**, demonstrating that the dual system completes full lifecycle validation on all three scales.
4. **Ablation experiments** characterizing MEMIT capacity, regularization sensitivity, nap-extended capacity, cross-edit retention, and perplexity dynamics through the full lifecycle.

2 Related Work

2.1 Knowledge Editing

ROME (Meng et al., 2022) demonstrated that factual associations in GPT-style models are localized in MLP layers and can be edited by rank-one updates to the value projection. MEMIT (Meng et al., 2023) extended this to batch editing, distributing updates across multiple layers simultaneously. Subsequent work has explored the scalability, reliability, and limitations of direct weight editing (Mitchell et al., 2022; Yao et al., 2023). These methods edit the raw completion pathway (e.g., “The Eiffel Tower is in” → “Paris”) but do not affect chat-template-formatted queries, a distinction that motivates our dual-system design.

2.2 Parameter-Efficient Fine-Tuning

LoRA (Hu et al., 2022) injects trainable low-rank matrices into frozen transformer layers, reducing trainable parameters by orders of magnitude. QLoRA (Dettmers et al., 2023) extends this to 4-bit quantized models. For continual learning specifically, O-LoRA (Wang et al., 2023b) learns tasks in orthogonal subspaces, and InfLoRA (Liang and Li, 2024) designs interference-free adaptation. Our system uses standard LoRA with adapter fusion after each sleep cycle, as conversational memories are not discrete tasks.

2.3 Complementary Learning Systems and Sleep-Inspired ML

CLS theory (McClelland et al., 1995; Kumaran et al., 2016) posits that the brain requires two learning systems: a hippocampal system for rapid episodic encoding and a neocortical system for gradual statistical extraction, with sleep mediating transfer between them (Diekelmann and Born, 2010; Rasch and Born, 2013). Several works have operationalized this for neural networks: Tadros et al. (2022) use Hebbian replay during simulated sleep; Carta et al. (2024) introduce wake-NREM-REM phases for image classification; Harun et al. (2023) propose on-device continual learning with sleep. For language models specifically, concurrent work explores RL-based knowledge seeding (Anonymous, 2025) and sleep-cycle training (Anonymous, 2024).

Our work differs in three ways. First, we combine *two distinct weight-modification mechanisms* (MEMIT and LoRA) rather than a single training algorithm, creating a genuine dual-system architecture. Second, we target conversational memory rather than task-incremental classification. Third, we validate across three model scales on both consumer and datacenter hardware.

2.4 Continual Learning

Classical approaches include EWC (Kirkpatrick et al., 2017), LwF (Li and Hoiem, 2017), and progressive networks (Rusu et al., 2016). Recent LLM-specific work shows that forgetting intensifies with scale (Luo et al., 2023), loss landscape flatness influences forgetting severity (Li et al., 2024), and apparent drops may reflect disrupted alignment rather than true knowledge loss (Zheng et al., 2025). Surveys catalog the full landscape (Wu et al., 2024; Shi et al., 2024). Our system combines low learning rates, LoRA-constrained updates, experience replay, and validation gating—an integrated approach rather than a single mechanism.

3 Method

3.1 System Overview

The system operates as a state machine with three phases: **wake** (inference with MEMIT injection), **nap** (quick LoRA consolidation of MEMIT facts), and **deep sleep** (full consolidation with curation, replay, dreaming, and training). An orchestrator manages transitions based on health-based triggers that track MEMIT edit count, elapsed time, and model perplexity.

The dual-system architecture is summarized in Figure 1. During wake, facts extracted from conversation are immediately injected into MLP weights via MEMIT, providing instant recall through raw text completion. When sleep pressure accumulates (via edit count, time, or perplexity drift), a nap consolidates MEMIT facts into LoRA weights and reverts the fragile MEMIT edits. Periodically, a full sleep cycle performs deep consolidation with experience replay and synthetic data generation.

Wake Phase	→ Chat with user, extract facts, inject via MEMIT ↓ (<i>sleep pressure: edits, time, perplexity</i>)
Nap	→ MEMIT facts → Q&A pairs → 1-epoch LoRA → revert MEMIT ↓ (<i>continued wake, more facts</i>)
Deep Sleep	→ Curate → Replay → Dream → LoRA train → Validate → Fuse

Figure 1: Dual-system lifecycle. MEMIT provides instant recall during wake; nap consolidates to LoRA; deep sleep performs thorough integration with validation gating.

3.2 Wake Phase: MEMIT Injection

During wake, facts are extracted from user messages as (s, r, o) triples (e.g., “Viktor”, “lives in”, “Portland”) and injected into the model’s MLP down-projection weights via MEMIT. Our implementation introduces three modifications to the original algorithm.

Fact representation. Each fact is a **FactTriple** (s, r, o) that generates a completion prompt $p = s \oplus r$ (e.g., “Viktor lives in”) and target $t = o$ (e.g., “Portland”). MEMIT edits the model so that $P(t | p)$ is maximized.

Target value optimization. For each fact, we compute a target value vector v^* by gradient descent on the log-probability of the target token:

$$v^* = \arg \min_v -\log P(t | p; \mathbf{W} + \Delta \mathbf{W}(v)) + \lambda_{\text{KL}} D_{\text{KL}}(P_{\text{new}} \| P_{\text{orig}}) \quad (1)$$

where λ_{KL} controls the trade-off between fact injection strength and preservation of the original output distribution. We optimize for 30 steps with learning rate 0.5.

Covariance regularization via Woodbury identity. The original MEMIT uses identity regularization ($\lambda \mathbf{I}$) when solving the constrained least-squares problem for weight updates. We replace this with the empirical covariance matrix $\hat{\mathbf{C}}$ of MLP intermediate activations, estimated from 200 reference texts spanning diverse topics:

$$\hat{\mathbf{C}} = \frac{1}{N} \sum_{i=1}^N \mathbf{k}_i \mathbf{k}_i^\top \quad (2)$$

where \mathbf{k}_i are MLP intermediate vectors (after SiLU-gated up-projection) from reference inputs. The weight update at layer ℓ is then:

$$\Delta \mathbf{W}_\ell = \mathbf{R}_\ell \mathbf{K}^\top (\mathbf{K} \mathbf{K}^\top + \lambda \hat{\mathbf{C}})^{-1} \quad (3)$$

where \mathbf{R}_ℓ is the residual (desired output minus current output) and \mathbf{K} is the matrix of key vectors. Computing $(\mathbf{K} \mathbf{K}^\top + \lambda \hat{\mathbf{C}})^{-1}$ directly would require inverting a $d \times d$ matrix (where d is the MLP intermediate dimension, e.g., 14336 for 8B). We use the Woodbury identity to keep the inversion in $N \times N$ space (where N is the number of facts, typically ≤ 10):

$$(\lambda \hat{\mathbf{C}} + \mathbf{K} \mathbf{K}^\top)^{-1} = \frac{1}{\lambda} \hat{\mathbf{C}}^{-1} - \frac{1}{\lambda} \hat{\mathbf{C}}^{-1} \mathbf{K} \left(\mathbf{I} + \frac{1}{\lambda} \mathbf{K}^\top \hat{\mathbf{C}}^{-1} \mathbf{K} \right)^{-1} \frac{1}{\lambda} \mathbf{K}^\top \hat{\mathbf{C}}^{-1} \quad (4)$$

This reduces the computational bottleneck from $O(d^3)$ to $O(N^3 + N^2 d)$, making covariance regularization practical even for large models.

Cross-edit null-space constraints. When injecting a new batch of facts, we include key vectors from all previously active MEMIT edits in the constraint set. This ensures that new edits operate in the null space of previous key vectors, preventing overwriting:

$$\mathbf{K}_{\text{combined}} = [\mathbf{K}_{\text{new}} \mid \mathbf{K}_{\text{prev}}], \quad \mathbf{R}_{\text{combined}} = [\mathbf{R}_{\text{new}} \mid \mathbf{0}] \quad (5)$$

The zero residual for previous keys means the update must not alter outputs for previously edited inputs.

Layer-wise residual distribution. The total residual is distributed across target layers (e.g., layers 8–15 for 3B, 12–19 for 8B), with each layer absorbing $1/L_{\text{remaining}}$ of the residual at its position. This prevents concentration of the entire update in a single layer, reducing the risk of catastrophic perturbation.

Dequantize-edit workflow. For quantized models, we dequantize only the target MLP layer before editing (converting packed 4-bit weights to float), apply the MEMIT delta, and keep the edited layer in float format. This adds ~ 48 MB per edited layer on the 3B model.

3.3 Nap: Quick Consolidation

When sleep pressure exceeds the nap threshold (default: 0.4), the system performs a quick consolidation:

1. Retrieve all active MEMIT facts from the edit ledger.
2. Convert each fact to Q&A training pairs (e.g., “Where does Viktor live?” → “Viktor lives in Portland.”).
3. Train a 1-epoch LoRA adapter on these pairs.
4. Validate: test recall of consolidated facts.
5. On success: revert all MEMIT edits (facts are now in LoRA weights).
6. On partial failure: re-inject unconsolidated facts via MEMIT.

The nap takes 30–120 seconds depending on model size and number of facts. After a successful nap, MEMIT capacity is freed for new facts, effectively extending the system’s memory ceiling.

3.4 Deep Sleep: Full Consolidation

Deep sleep is a four-stage pipeline triggered when sleep pressure exceeds the sleep threshold (default: 0.8) or manually:

Stage 1: Curation. Conversation exchanges are scored on novelty, importance, and utility. Exchanges below configurable thresholds are discarded. MEMIT-held facts are converted to training pairs and merged with curated conversation data.

Stage 2: Replay and dreaming. High-scoring examples from previous cycles are mixed into training data at a configurable ratio (20% for light sleep, 60% for deep sleep), with priority decaying by factor $d = 0.85$ per replay (spaced repetition). During deep sleep, the model generates synthetic Q&A pairs from its knowledge (“dreaming”), creating associative connections.

Stage 3: LoRA training. The combined dataset trains a LoRA adapter. Iterations scale with data: $N_{\text{iters}} = |\mathcal{D}| \times \text{epochs}$. Light sleep uses learning rate 1×10^{-4} with 3 epochs; deep sleep uses 5×10^{-5} with 2 epochs.

Stage 4: Validation and fusion. Pre-sleep and post-sleep benchmark scores are compared. The cycle is accepted if $s_{\text{post}} \geq \tau \cdot s_{\text{pre}}$ (default $\tau = 0.5$). On success, the adapter is fused into base weights and MEMIT edits are reverted. On failure, the model rolls back to the pre-sleep checkpoint and MEMIT edits are re-applied.

3.5 Health-Based Sleep Triggers

Sleep pressure is a weighted combination of three signals:

$$\text{pressure} = w_e \cdot \frac{n_{\text{edits}}}{n_{\max}} + w_t \cdot \frac{t_{\text{elapsed}}}{t_{\max}} + w_p \cdot \max\left(0, \frac{\text{PPL}_{\text{current}}}{\text{PPL}_{\text{baseline}}} - 1\right) \quad (6)$$

where $w_e = 0.6$, $w_t = 0.3$, $w_p = 0.1$ are the edit, time, and perplexity weights; $n_{\max} = 50$ is the maximum active edits; and $t_{\max} = 7200$ s is the maximum wake duration. The system triggers a nap at pressure ≥ 0.4 and full sleep at ≥ 0.8 .

4 Experimental Setup

4.1 Hardware

Experiments span two hardware configurations:

- **Consumer:** MacBook Air M3 with 8 GB unified memory, using the MLX framework ([Hannun et al., 2023](#)). Runs the 3B model.
- **Datacenter:** Dual NVIDIA H100 GPUs (80 GB each, 160 GB total) via Vast.ai, using PyTorch with bitsandbytes 4-bit quantization. Runs 8B and 70B models.

4.2 Models

- **3B:** `mlx-community/Llama-3.2-3B-Instruct-4bit` (MLX 4-bit quantization)
- **8B:** `meta-llama/Llama-3.1-8B-Instruct` (bfloating16, unquantized)
- **70B:** `meta-llama/Llama-3.1-70B-Instruct` (bitsandbytes NF4 quantization)

4.3 MEMIT Configuration

Table 1 summarizes the MEMIT hyperparameters across model sizes.

Table 1: MEMIT configuration per model size.

Parameter	3B	8B	70B
Target layers	8–15	12–19	40–55
λ_{reg}	0.1	0.1	0.1
Target module	down_proj	down_proj	down_proj
Covariance samples	200	200	200
v^* learning rate	0.5	0.5	0.5
v^* optimization steps	30	30	30
KL factor	0.0625	0.0625	0.0625

4.4 LoRA Configuration

All models use rank 16, alpha 32, targeting 8 transformer layers. Light sleep: learning rate 1×10^{-4} , 3 epochs. Deep sleep: 5×10^{-5} , 2 epochs. Nap: 1×10^{-4} , 1 epoch.

4.5 Evaluation Protocol

We evaluate along three axes:

- **Raw completion recall:** Given a prompt like “Viktor lives in”, does the model complete with the correct target? This tests the MEMIT edit pathway.
- **Chat-template recall:** Given a question like “Where does Viktor live?” in chat format, does the model answer correctly? This tests LoRA-consolidated knowledge.
- **Perplexity:** Cross-entropy loss on reference texts, measuring model coherence.

5 Results

5.1 MEMIT Capacity Scaling

Table 2 shows MEMIT’s raw capacity (without nap consolidation) across model sizes, measured by injecting increasing batches of synthetic facts and testing cumulative recall.

Table 2: MEMIT capacity across model sizes. Recall is measured cumulatively after each batch of 5 facts. The 8B model achieves the best capacity density.

Model	5 facts	10 facts	15 facts	20 facts	40 facts
3B (4-bit)	0.80	0.70	0.60	—	—
8B	0.90	0.82	0.80	0.78	0.82
70B (4-bit)	0.80	0.80	0.73	0.70	0.80

The 3B model’s capacity ceiling is approximately 10 facts at 0.70 recall, consistent with its smaller MLP dimension (11008 intermediate) providing less room for orthogonal edits. The 8B model sustains 0.82 recall at 40 facts—the highest capacity density of the three—likely due to its larger MLP (14336 intermediate) without the quantization artifacts of the 70B model. The 70B model shows competitive recall at 40 facts but with more variance, potentially due to 4-bit quantization interacting with float-precision MEMIT edits across two GPUs.

5.2 Dual System vs. Components (Ablation 1)

We compare three conditions on the 8B model, each learning the same 5 facts:

1. **MEMIT-only:** Inject 5 facts, no consolidation. Measure recall immediately and after 20 filler turns.
2. **LoRA-only:** Teach 5 facts via conversation, trigger full sleep, clear context. Measure recall.
3. **MEMIT+LoRA:** Inject via MEMIT, trigger nap, then full sleep. Clear context. Measure recall.

The key finding is complementarity. MEMIT provides *instant, lossless* recall: all 5 facts are available within 4 seconds of injection and survive 20 unrelated conversation turns with zero degradation ($1.00 \rightarrow 1.00$). However, MEMIT edits do not survive model restarts. LoRA-only achieves only 1/5 raw completion recall after a full sleep cycle—the LoRA training targets the chat-template pathway, not raw completions. The dual system inherits the best of both: instant recall from MEMIT during wake (5/5), and durable recall after consolidation and restart (4/5). The one missed fact (“Elena Voronov works as marine biologist”) shows the nap acting as a selective filter—facts with weaker LoRA training signal are not consolidated, a property we discuss further in Section 6.4.

Table 3: Dual system comparison (Ablation 1, 8B model, 5 facts). MEMIT provides instant recall but no restart persistence. LoRA provides some persistence but no instant recall. The dual system provides both.

Metric	MEMIT-only	LoRA-only	MEMIT+LoRA
Time to first recall	4.4s (instant)	24.6s (after sleep)	4.0s (instant)
Immediate recall	1.00	0.00	1.00
Post-filler/restart recall	1.00	0.20	0.80
Baseline perplexity	4.12	4.12	4.12
Final perplexity	4.10	4.46	5.16

5.3 Cross-Edit Retention (Ablation 2)

This experiment tests whether MEMIT-injected facts survive subsequent operations: filler conversations and new fact injections.

Table 4: MEMIT retention through interference (Ablation 2, 8B model). Null-space constraints achieve perfect preservation of Batch A facts through both filler chat and new fact injection. Nap consolidation selectively transfers the strongest-encoded facts.

Stage	Batch A (5)	Batch B (5)	Combined (10)
After A injection	1.00	—	—
After 20 filler turns	1.00	—	—
After B injection	1.00	1.00	1.00
After nap consolidation	0.20	0.80	0.50

The null-space constraint (Equation 5) achieves **perfect retention**: Batch A recall remains 1.00 through both 20 filler turns and the injection of 5 new Batch B facts. Without constraints, injecting Batch B would overwrite Batch A edits, as both compete for the same MLP weight dimensions. With constraints, Batch B edits are projected into the orthogonal complement of Batch A’s key vectors, achieving null-space retention of 1.00.

The post-nap results reveal the nap’s role as a selective consolidation filter. Batch B (0.80 recall) consolidates better than Batch A (0.20) because Batch B’s MEMIT edits were more recent and had stronger encoding signal at nap time. This is not a deficiency—it mirrors biological sleep consolidation, where not all hippocampal traces transfer to neocortex. The facts that survive nap are those with the strongest training signal, while weakly-encoded traces decay (Section 6.4).

5.4 Regularization Analysis (Ablation 3)

We sweep λ_{reg} across five values (0.01, 0.05, 0.1, 0.5, 1.0), injecting the same 10 facts at each setting and measuring recall and perplexity.

The results show that the 8B model is remarkably robust to λ at 10 facts: perfect recall (10/10) at all five values with perplexity changes within ± 0.02 —effectively noise. The mean delta norm is nearly constant across all settings (~ 2.58), indicating that the v^* optimization converges to similar solutions regardless of regularization strength. This contrasts sharply with the 3B model, where λ significantly affects recall (Section 5.1), and suggests that the 8B model’s MLP dimension (14336) provides sufficient capacity for 10 orthogonal edits without requiring strong regularization to prevent interference. The lambda–recall trade-off would likely emerge at higher fact counts or

Table 5: Lambda regularization sweep (Ablation 3, 8B model, 10 facts). The 8B model achieves perfect recall across all lambda values with negligible perplexity impact, indicating that covariance regularization is effective but the 8B model has sufficient capacity to absorb 10 edits regardless of regularization strength.

λ	Recall	PPL (base)	PPL (post)	Δ PPL
0.01	1.00	3.62	3.60	-0.020
0.05	1.00	3.62	3.61	-0.015
0.10	1.00	3.62	3.63	+0.004
0.50	1.00	3.62	3.62	-0.006
1.00	1.00	3.62	3.62	+0.001

on smaller models.

5.5 Nap-Extended Capacity (Ablation 4)

We compare the effective capacity ceiling under two conditions:

1. **MEMIT-only:** Inject facts in batches of 5, never consolidate. Continue until recall drops below 0.5.
2. **MEMIT+Nap:** Same batches, but trigger a nap after every 10 new facts. Continue until failure.

Table 6: Capacity ceiling comparison (Ablation 4, 8B model). MEMIT-only sustains high recall through 60 facts. MEMIT+Nap shows lower recall after nap consolidation, reflecting the nap’s role as a selective triage filter rather than a lossless transfer mechanism.

Total Facts	MEMIT-only Recall	MEMIT+Nap Recall	Naps Triggered
5	0.80	0.80	0
10	0.80	0.60	1
15	0.73	0.40	1
20	0.70	—	—
30	0.73	—	—
40	0.83	—	—
50	0.80	—	—
60	0.83	—	—

The results invert the initial hypothesis. MEMIT-only sustains remarkably stable recall—0.80 to 0.83—through 60 facts (12 batched edits), with no sign of degradation. The 8B model’s MLP dimension (14336) provides sufficient orthogonal capacity for at least 60 simultaneous edits, far exceeding the expected ceiling.

MEMIT+Nap, by contrast, shows *lower* recall after the first nap at 10 facts (0.60) and drops below the 0.5 threshold at 15 facts. This is not a failure of capacity extension—it reveals the nap’s function as a **selective consolidation filter**. The nap transfers MEMIT facts to LoRA weights and then reverts the MEMIT edits, but LoRA consolidation from a single training epoch is lossy by design: only facts with sufficiently strong training signal survive the transfer. Facts that were weakly encoded or ambiguous are filtered out during consolidation.

This maps directly to biological sleep triage (Section 6.4): not all hippocampal traces transfer to neocortex during NREM replay. The nap acts as a quality gate, preserving the strongest memories

and allowing marginal ones to decay. In a production system, facts that fail nap consolidation can be re-injected via MEMIT for another consolidation attempt, or flagged for deep sleep processing.

5.6 Perplexity Dynamics (Ablation 5)

We track perplexity through the complete lifecycle: baseline \rightarrow 5 MEMIT injections \rightarrow nap \rightarrow 5 more injections \rightarrow full sleep.

Table 7: Perplexity trajectory through lifecycle (Ablation 5, 8B model). MEMIT injections have near-zero perplexity impact. Nap consolidation causes a significant perplexity jump as MEMIT edits are reverted and replaced with LoRA weights. Full sleep partially recovers coherence.

Step	Event	Perplexity	Recall
0	Baseline	5.752	—
1	After fact 1 (MEMIT)	5.777	1.00
2	After fact 2	5.748	1.00
3	After fact 3	5.780	1.00
4	After fact 4	5.751	1.00
5	After fact 5	5.751	1.00
6	After nap	8.622	0.80
7	After fact 6	8.662	0.83
8	After fact 7	8.645	0.71
9	After fact 8	8.767	0.75
10	After fact 9	8.804	0.67
11	After fact 10	8.917	0.70
12	After full sleep	8.027	0.40

Three distinct regimes emerge. **First**, MEMIT injections have near-zero perplexity impact: steps 1–5 fluctuate within ± 0.03 of the 5.752 baseline, confirming that covariance-regularized MEMIT edits preserve model coherence even as facts accumulate.

Second, the nap causes a sharp perplexity jump from 5.75 to 8.62 ($\Delta PPL = +2.87$). This occurs because the nap reverts MEMIT edits and replaces them with a 1-epoch LoRA adapter. The LoRA weights, trained on only 5 Q&A pairs, are sufficient for factual recall (0.80) but introduce distributional perturbation on general text—the model has been slightly “bent” toward the training distribution. Subsequent MEMIT injections on the post-nap model cause further drift (8.62 \rightarrow 8.92 over 5 facts), as MEMIT edits interact with the already-perturbed LoRA weights.

Third, full sleep partially recovers coherence (8.92 \rightarrow 8.03, $\Delta PPL = -0.89$). The deeper consolidation cycle—with experience replay, dreaming, and multi-epoch training—produces a better-integrated adapter than the nap’s single-epoch pass. However, it does not fully recover to baseline, and recall drops further (0.70 \rightarrow 0.40) as sleep consolidation applies its own selective triage (Section 6.4).

The key insight is that MEMIT is remarkably perplexity-neutral while active, but LoRA consolidation carries a coherence cost. This motivates keeping MEMIT edits active as long as possible and consolidating only when capacity pressure or model health requires it.

5.7 Full Lifecycle Results

Table 8 summarizes end-to-end lifecycle results across all three model sizes.

Table 8: Full lifecycle results. Each model executes the complete wake → MEMIT → nap → sleep pipeline. MEMIT recall is via raw completion; sleep validation uses chat-template Q&A.

Model	MEMIT Recall	Capacity@40	Nap Result	Sleep Score	Lifecycle
3B (4-bit)	3/3	0.80@10	2/5 consolidated	2/5	PASS
8B	2/3	0.82@40	2/4 consolidated	5/5	PASS
70B (4-bit)	2/3	0.80@40	Partial	5/5	PASS

All three models complete the full lifecycle with validation approval. The 8B model achieves the best overall performance: highest MEMIT capacity (0.82 at 40 facts) and perfect sleep validation (5/5). The 3B model’s lower sleep score (2/5) reflects the narrow viable learning rate window documented in Baranov (2026). The 70B model achieves perfect sleep validation despite partial nap results, suggesting that LoRA training generalizes well in the chat-template format even when raw completion recall is imperfect.

Key observation: pathway complementarity. MEMIT edits the *raw completion pathway*: given “Viktor lives in”, the model completes with “Portland”. LoRA training edits the *chat-template pathway*: given a question in chat format, the model answers correctly. These are complementary—MEMIT provides instant recall during wake (where raw completion queries can be checked), while LoRA provides robust recall in all interaction formats after consolidation.

6 Discussion

6.1 Why the Dual System Works

The CLS theory mapping is not merely metaphorical—it predicts the system’s behavior. The hippocampus (MEMIT) provides high-fidelity encoding that degrades under interference: adding more facts reduces recall of earlier ones, exactly as observed in our capacity experiments. The neocortex (LoRA) provides slow but stable integration: facts trained into LoRA weights survive model restarts and generalize to new query formats. The nap mechanism (hippocampal replay) transfers fragile episodic traces into stable semantic representations, exactly as NREM sleep replay is theorized to do in biological systems (Diekelmann and Born, 2010).

6.2 The 8B Sweet Spot

The 8B model outperforms both the 3B and 70B models on most metrics. This is not a monotonic scaling effect—it reflects a balance between model capacity and quantization. The 3B model has insufficient MLP dimension for many simultaneous MEMIT edits. The 70B model has ample capacity but introduces quantization artifacts (NF4) and multi-GPU device boundaries that interfere with MEMIT’s float-precision edits. The 8B model, running unquantized in bfloat16, provides the cleanest editing substrate.

6.3 Multi-GPU Challenges

Scaling to 70B on dual H100s revealed several implementation challenges:

- **Device mismatch:** MEMIT computes on the device of each layer’s MLP weights. In a model distributed across GPUs, tensors must be explicitly moved to the correct device at 5+ locations in the MEMIT pipeline.

- **LoRA optimizer:** PyTorch’s fused multi-tensor AdamW operations fail across device boundaries. Setting `foreach=False` falls back to per-parameter updates.
- **Model preparation:** The standard `prepare_model_for_kbit_training()` function corrupted 4-bit weights on multi-GPU setups. Replacing it with `enable_input_require_grads()` resolved the issue.
- **Adapter fusion:** Merging adapters and saving triggers a full model materialization that can crash on meta-device tensors. For nap cycles, we skip fusion and use the in-memory merged model directly.

6.4 Nap as Selective Consolidation Filter

A consistent pattern across ablations 1, 2, 4, and 5 is that recall *decreases* after nap consolidation. This is not a deficiency—it is the intended behavior of a selective consolidation filter, and it maps directly onto biological sleep function.

In biological systems, not all hippocampal traces transfer to neocortex during NREM replay ([Diekelmann and Born, 2010](#); [Rasch and Born, 2013](#)). Sleep acts as a triage mechanism: strongly-encoded, emotionally salient, or frequently-rehearsed memories are preferentially consolidated, while weakly-encoded or redundant traces decay. This is adaptive—a system that indiscriminately transfers everything would overload long-term storage with noise.

Our nap mechanism exhibits the same behavior. In Ablation 2, Batch B (0.80 post-nap recall) consolidates better than Batch A (0.20) because Batch B’s MEMIT edits were more recent and had stronger encoding signal at nap time. In Ablation 4, MEMIT+Nap recall drops to 0.60 after the first nap, as the single-epoch LoRA training provides a narrow bandwidth for consolidation. In Ablation 5, recall decreases from 1.00 (pre-nap, MEMIT active) to 0.80 (post-nap, LoRA only), with the one failed fact being the least distinctively encoded.

This selectivity has three desirable properties:

1. **Quality filtering:** Only facts with strong, unambiguous training signal survive consolidation. Weakly-encoded or contradictory facts are discarded rather than corrupting the LoRA weights.
2. **Capacity efficiency:** LoRA weight space is finite. Selective consolidation allocates it to high-confidence memories rather than exhausting it on marginal traces.
3. **Recoverability:** Facts that fail nap consolidation are not permanently lost—they can be re-injected via MEMIT and given another consolidation opportunity during deep sleep, which uses multi-epoch training with experience replay and achieves higher transfer rates.

The system thus operates as a multi-stage memory pipeline: MEMIT provides a high-capacity, high-fidelity buffer (60+ facts at 0.83 recall); nap consolidation acts as a first-pass filter, transferring the strongest 60–80% of facts to LoRA; and deep sleep provides a second, more thorough consolidation pass with richer training signal.

6.5 Where the Biological Analogy Holds and Breaks

The analogy holds for: dual learning rates, capacity-dependent interference, replay-mediated consolidation, health-based sleep triggers, and—as demonstrated in Ablations 1, 2, 4, and 5—selective consolidation during sleep, where strongly-encoded traces are preferentially transferred while weak traces decay. It breaks for: continuous biological learning (our system has a sharp inference/training boundary), multiple memory systems (we have only two pathways vs. the brain’s many), and active forgetting (we lack a targeted erasure mechanism, relying instead on passive decay during consolidation). The analogy is strongest as a *design framework*—it predicted not only that combining fast

and slow systems would outperform either alone, but also that consolidation should be selective rather than exhaustive, both of which the ablation results confirm.

7 Limitations

Single-run experiments. All reported numbers are from single runs without error bars. The experiments should be repeated with multiple random seeds to establish confidence intervals. We prioritized breadth of ablations over statistical rigor in this initial report.

Template-based fact extraction. Facts are extracted using regex patterns for personal information (name, location, occupation). This misses complex or implicit facts. Model-based extraction would be more general but slower.

No selective forgetting. Once information is consolidated into LoRA weights, there is no mechanism to remove it short of rolling back to a prior checkpoint. The system can only add memories, not delete them.

Blocking sleep. The model goes offline during nap and sleep cycles. A production system would require background consolidation or a secondary model for handling requests during sleep.

MEMIT edits raw completion only. MEMIT-injected facts are only accessible through raw text completion (“Viktor lives in” → “Portland”), not through chat-template queries (“Where does Viktor live?”). During wake, the system can route recall checks through the raw pathway, but this is a design limitation compared to LoRA’s format-agnostic recall.

Quantization interaction. MEMIT edits float-precision weight matrices on models with 4-bit quantized weights. The dequantize-edit workflow works but introduces a mixed-precision boundary that may affect edit quality on highly quantized models.

8 Conclusion

We presented a dual-system architecture for lifelong learning in language models that combines MEMIT for instant factual injection with LoRA sleep-wake consolidation for permanent storage. The system draws on Complementary Learning Systems theory, mapping MEMIT to hippocampal fast encoding, LoRA to neocortical slow integration, and nap cycles to selective sleep-mediated consolidation.

The architecture was validated across three model scales (3B, 8B, 70B) on consumer and data-center hardware, with all sizes completing the full wake-nap-sleep lifecycle with validation approval. Ablation experiments demonstrate that the dual system provides capabilities unavailable to either component alone: MEMIT delivers instant recall (5/5 facts within 4 seconds) while LoRA provides persistent, format-agnostic recall after consolidation (4/5 post-restart). Covariance-regularized MEMIT with cross-edit null-space constraints achieves perfect retention (1.00) of previously injected facts through both filler conversations and new injections. On the 8B model, MEMIT sustains 0.83 recall at 60 facts with near-zero perplexity impact ($\Delta PPL < 0.03$), demonstrating that direct weight editing can be remarkably non-destructive when properly regularized.

A key finding is that nap consolidation functions as a selective triage filter rather than a lossless transfer mechanism—mirroring biological sleep, where not all hippocampal traces transfer to neocortex. This selectivity is adaptive: it allocates finite LoRA capacity to strongly-encoded facts while allowing marginal traces to decay or await deeper consolidation.

The 8B model emerged as the sweet spot, achieving the highest MEMIT capacity and perfect sleep validation (5/5), benefiting from unquantized bfloat16 weights that provide the cleanest substrate for both MEMIT edits and LoRA training. The lambda regularization sweep confirmed the 8B model’s robustness: perfect recall at all tested values (0.01–1.0) with negligible perplexity variation.

Future work includes: multi-seed experiments for statistical rigor; model-based fact extraction to handle complex knowledge; selective forgetting mechanisms; non-blocking background consolidation; tuning the nap’s consolidation bandwidth to control the selectivity–retention trade-off; and online learning that blurs the wake-sleep boundary, moving toward the brain’s continuous learning regime.

References

- Anonymous. Dreaming is all you need. *arXiv preprint arXiv:2409.01633*, 2024.
- Anonymous. Language models need sleep: Learning to self modify and consolidate memories. *OpenReview*, 2025.
- V. Baranov. Sleep-wake consolidation for lifelong conversational memory in local language models. *arXiv preprint*, 2026. v1 of this work.
- A. Carta et al. Wake-sleep consolidated learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. Also arXiv:2401.08623.
- T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer. QLoRA: Efficient finetuning of quantized LLMs. In *Advances in Neural Information Processing Systems*, volume 36, 2023.
- S. Diekelmann and J. Born. The memory function of sleep. *Nature Reviews Neuroscience*, 11(2): 114–126, 2010.
- A. Hannun, J. Digani, A. Katharopoulos, and R. Collobert. MLX: Efficient and flexible machine learning on Apple silicon. Apple Machine Learning Research. <https://github.com/ml-explore/mlx>, 2023.
- M. Y. Harun, J. Gallardo, T. L. Hayes, R. Kemker, and C. Kanan. SIESTA: Efficient online continual learning with sleep. *Transactions on Machine Learning Research*, 2023.
- E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.
- J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017.

- D. Kumaran, D. Hassabis, and J. L. McClelland. What learning systems do intelligent agents need? Complementary Learning Systems theory updated. *Trends in Cognitive Sciences*, 20(7):512–534, 2016.
- P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, and D. Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474, 2020.
- H. Li, L. Ding, M. Fang, and D. Tao. Revisiting catastrophic forgetting in large language model tuning. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2024.
- Z. Li and D. Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2017.
- Y.-S. Liang and W.-J. Li. InfLoRA: Interference-free low-rank adaptation for continual learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23638–23647, 2024.
- Y. Luo, Z. Yang, F. Meng, Y. Li, J. Zhou, and Y. Zhang. An empirical study of catastrophic forgetting in large language models during continual fine-tuning. *arXiv preprint arXiv:2308.08747*, 2023.
- J. L. McClelland, B. L. McNaughton, and R. C. O'Reilly. Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, 102(3):419–457, 1995.
- K. Meng, D. Bau, A. Andonian, and Y. Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, pages 17359–17372, 2022.
- K. Meng, A. S. Sharma, A. J. Andonian, Y. Belinkov, and D. Bau. Mass-editing memory in a transformer. In *International Conference on Learning Representations*, 2023.
- E. Mitchell, C. Lin, A. Bosselut, C. Finn, and C. D. Manning. Fast model editing at scale. *arXiv preprint arXiv:2110.11309*, 2022.
- B. Rasch and J. Born. About sleep's role in memory. *Physiological Reviews*, 93(2):681–766, 2013.
- A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016.
- H. Shi, Z. Xu, H. Wang, W. Qin, W. Wang, Y. Wang, and H. Wang. Continual learning of large language models: A comprehensive survey. *ACM Computing Surveys*, 2024.
- T. Tadros, G. P. Krishnan, R. Ramyaa, and M. Bazhenov. Sleep-like unsupervised replay reduces catastrophic forgetting in artificial neural networks. *Nature Communications*, 13:7742, 2022.
- W. Wang, L. Dong, H. Cheng, X. Liu, X. Yan, J. Gao, and F. Wei. Augmenting language models with long-term memory. In *Advances in Neural Information Processing Systems*, volume 36, 2023a.
- X. Wang, T. Chen, Q. Ge, H. Xia, R. Bao, R. Zheng, Q. Zhang, T. Gui, and X. Huang. O-LoRA: Orthogonal subspace learning for language model continual learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10658–10671, 2023b.

Y. Wang, Y. Gao, X. Chen, H. Jiang, S. Li, J. Yang, Q. Yin, Z. Li, X. Li, B. Yin, J. Shang, and J. McAuley. MemoryLLM: Towards self-updatable large language models. In *International Conference on Machine Learning*, 2024.

M. A. Woodbury. Inverting modified matrices. *Memorandum Report*, 42:336, 1950.

T. Wu, L. Luo, Y.-F. Li, S. Pan, T.-T. Vu, and G. Haffari. Continual learning for large language models: A survey. *arXiv preprint arXiv:2402.01364*, 2024.

Y. Yao, P. Wang, B. Tian, S. Cheng, Z. Li, S. Deng, H. Chen, and N. Zhang. Editing large language models: Problems, methods, and opportunities. *arXiv preprint arXiv:2305.13172*, 2023.

J. Zheng, X. Cai, S. Qiu, and Q. Ma. Spurious forgetting in continual learning of language models. In *International Conference on Learning Representations*, 2025.