TASK-FOCUSED MEMORIZATION FOR MULTIMODAL AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-term memory is essential for multimodal agents to build coherent experience, accumulate world knowledge, and achieve continual learning. Beyond module design and basic fidelity, the key challenge lies in determining what to memorize. Multimodal agents, such as embodied agents, continuously perceive, reason, and act in real or virtual environments, receiving an unbounded stream of multimodal observations. From this combinatorial explosion of information, an agent must selectively retain content that is relevant to its role in the environment and valuable for future tasks. To this end, we frame memory generation as a learnable memorization policy and introduce TaskMem (**Task**-focused **Mem**orization Policy Learning), a reinforcement-learning-based framework that enables the policy to dynamically adjust its focus to the demands of real tasks encountered in the environment. TaskMem adopts a two-phase training paradigm: Phase One learns how to memorize by optimizing memory quality under fundamental fidelity requirements; Phase Two occurs after deployment, where the agent learns what to memorize by tuning an adapter on its base MLLM, using recent environment tasks to define a reward model that guides the memorization policy toward task-relevant content. To evaluate our approach, we reformulate VideoMME, EgoLife, and EgoTempo into streaming benchmarks that simulate a realistic setting in which an agent processes streaming observations and handles tasks arriving online. Questions must be answered from memory alone, without access to raw video. Built on Qwen3-VL-30B-A3B, TaskMem improves VQA accuracy by 6.3%, 7.0%, and 5.8% on these benchmarks, respectively.

1 INTRODUCTION

Long-term memory is a cornerstone of intelligence (Graves et al., 2014; Hendrycks et al., 2025). This is especially critical for multimodal agents, such as embodied AI agents (Yang et al., 2024; Team et al., 2025), which perceive and act within physical or virtual environments. These agents must continuously perceive and reason over unbounded streams of visual, auditory, and spatial information in dynamic real-world environments. Long-term memory is therefore essential for maintaining coherence across modalities over time (Song et al., 2024; He et al., 2024a), accumulating world knowledge (Long et al., 2025; Wang et al., 2024b), supporting continual learning (Wang et al., 2024a; 2025), and improving complex, long-horizon decision-making (Wang et al., 2024b; Long et al., 2025).

The core challenge of memorization lies in whether an agent can **autonomously decide what to memorize**. Although multimodal agents can perceive and understand vast amounts of information, a fundamental question remains: which information should be stored in long-term memory? This relates to the AI Frame Problem (Shanahan, 2004), which concerns identifying contextually relevant information without being overwhelmed by the combinatorial explosion of possibilities. Extending this perspective, an agent must decide not only present relevance but future usefulness. Consequently, memory selection should adapt dynamically to the agent's role and tasks within its environment. As shown in Figure 1, a robot assigned housework should remember the house layout, whereas one receiving user-related instructions should prioritize the user's preferences, habits, and emotions. In this sense, memory is not merely a passive storage system, but an active, goal-driven process. An effective agent should continuously retain information that maximizes future utility.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

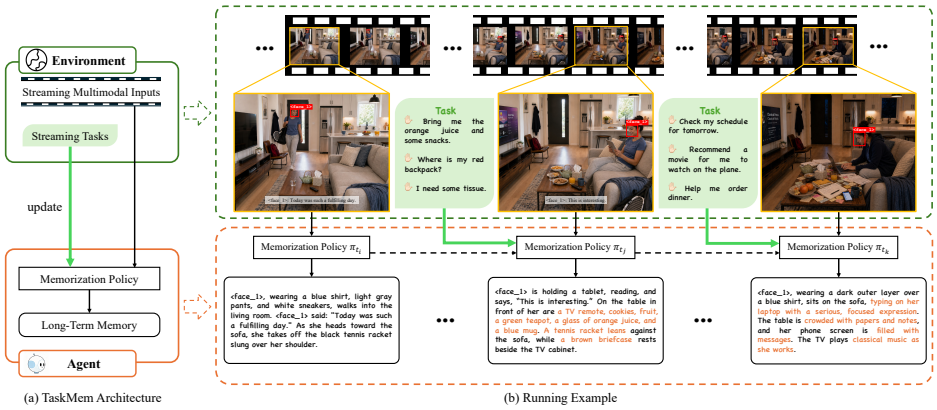


Figure 1: Architecture of TaskMem with a running example. The memorization policy online updates memory generation based on real tasks from the environment, producing task-relevant content.

Recent works propose diverse frameworks for multimodal agents with long-term memory (Chen et al., 2026a; Long et al., 2025). Most treat memory construction as a process independent of task execution. In these frameworks, multimodal large language models (MLLMs) (Comanici et al., 2025; Xu et al., 2025) are used to generate memory content, alongside system-level mechanisms for memory storage (He et al., 2024a; Song et al., 2024), consolidation (Chen et al., 2026a; Liu et al., 2025), and error correction (Chen et al., 2026b). However, memory generation itself remains largely heuristic, relying on prompt engineering (Fan et al., 2024; Chen et al., 2026a) or post-training with predefined templates (Long et al., 2025; Liu et al., 2025), and does not explicitly optimize what information should be memorized. Consequently, memory formation may misalign with real-world task demands.

To bridge this gap, we frame memory generation as a learnable memorization policy rather than a fixed summarization step. Given streaming multimodal inputs and recent memory history, the policy decides what information to store at each moment.

As shown in Figure 1, we introduce TaskMem, a reinforcement learning (RL)-based framework that optimizes the memorization policy so that memories are both intrinsically correct and relevant to tasks in the agent's deployment environment. TaskMem adopts a two-phase optimization. In **Phase One**, the policy learns how to memorize by optimizing memory quality under fundamental requirements such as correctness, non-redundancy, and format compliance, which we formulate as a multi-objective RL problem. In **Phase Two**, the policy learns what to memorize through online learning in an actual environment. This raises three challenges: (1) sparse feedback from only a few recent tasks; (2) catastrophic forgetting of capabilities acquired in Phase One; (3) computational constraints, as updates must not affect serving. To address them, we tune a lightweight 2,048-parameter adapter on top of the base MLLM, with a reward model that transforms sparse, task-level signals into denser supervision by constructing augmented pairwise preference data, thereby guiding the policy toward task-relevant memory while preserving the general capabilities learned in Phase One.

We evaluate TaskMem on VideoMME (Fu et al., 2025), EgoLife (Yang et al., 2025), and EgoTempo (Plizzari et al., 2025) by recasting each VQA benchmark into sequential task streams that simulate a multimodal agent perceiving and processing tasks sequentially within an environment. Each question is revealed only after its video is processed and must be answered from memory alone, so accuracy reflects memory quality. Built on Qwen3-VL-30B-A3B (Bai et al., 2025a), Phase One alone improves VQA accuracy over the base model, and Phase Two further aligns memory with the environment, overall improving accuracy by 6.3%, 7.0%, and 5.8% on the three benchmarks.

The main contributions of this paper are summarized as follows:

- We frame memory generation as a learnable policy that autonomously decides what to memorize from streaming multimodal inputs, addressing the core challenge of memory selection and transforming memory from passive storage into an active, goal-driven process.
- We propose TaskMem, an RL–based framework that optimizes the memorization policy to generate task-relevant memory within an environment.

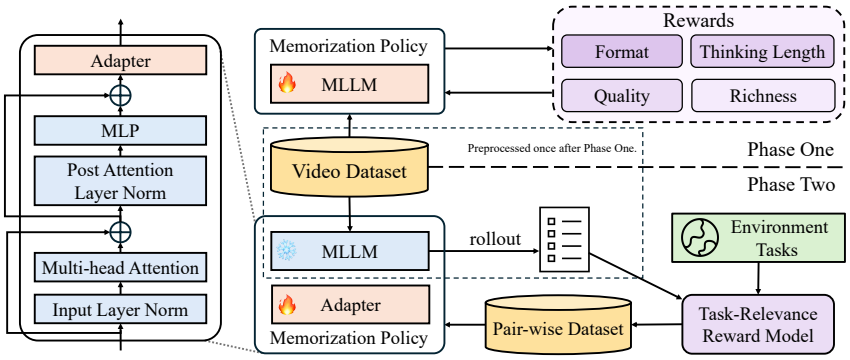


Figure 2: Two-phase training in TaskMem: Phase One optimizes the memorization policy for fundamental capabilities; Phase Two optimizes the policy to generate task-relevant content.

- In streaming VQA experiments, both Phase One and Phase Two training demonstrate consistent improvements across VideoMME, EgoLife, and EgoTempo benchmarks.

2 Approach

2.1 Problem Formulation

Following mainstream work on long-term memory (Long et al., 2025; Yeo et al., 2025; Yang et al., 2025), we formulate the memorization process as a task that takes a streaming video as input and generates memory content. As a representative case, we focus on episodic memory, which captures temporally ordered, event-centric experiences of a multimodal agent. The same formulation can extend to semantic and visual memory.

At each step t, the agent observes a new video segment v_t and maintains the memories generated so far. The memorization policy conditions on a sliding-window context consisting of the k most recent video segments and the memories generated for the first $k-1$ segments in this window, denoted as $q_t = (v_{t-k+1:t}, m_{t-k+1:t-1})$. Given the context q_t, the memorization policy $\pi_\theta(m_t|q_t)$ determines a memory m_t for the segment v_t. A good episodic memory is faithful to the current video segment, coherent with previous memories, non-redundant, and useful for future tasks in the environment.

Maintaining consistent identities across video segments requires linking faces and voices to previously seen individuals (He et al., 2024b). TaskMem directly annotates videos with persistent identities: detected faces are enclosed in bounding boxes labeled with global face IDs, and ASR transcripts are overlaid as time-aligned subtitles tagged with speaker IDs. Unlike prior work that appends lengthy tool outputs as extra text (Long et al., 2025), this design shortens the context without sacrificing identity reasoning accuracy. Implementation details are in Appendix B.

In RL terminology, the full trajectory of the memorization policy is $\tau = (v_{t-k+1:t}, m_{t-k+1:t})$. A trajectory-level reward $r(\tau)$ is provided at the last token of the trajectory. The objective is to learn a memorization policy π_θ that maximizes the expected return: $\mathbb{E}_{\tau\sim\pi_\theta}[r(\tau)]$. For notational simplicity, in the remainder of the paper, we drop the absolute time index and re-index the sliding-window context as $q = (v_1, \ldots, v_k, m_1, \ldots, m_{k-1})$ and the corresponding full trajectory as $\tau = (q, m_k)$.

2.2 Phase One: How to Memorize

Phase One optimizes the memorization policy before deployment under fundamental requirements such as factual accuracy, non-redundancy, and proper formatting. Memory generation is an open-ended, long-form task. Prior work mainly relies on supervised fine-tuning (SFT) (Long et al., 2025; Yang et al., 2025), which has two limitations. First, performance is capped by the quality of the models used to generate synthetic training data. Second, the maximum likelihood objective does not explicitly enforce global-level goals. To overcome these limitations, we therefore adopt RL to train memory generation from scratch, as shown in Figure 2. RL optimizes directly from reward signals, requiring only raw video rather than curated SFT data. Additionally, large multimodal and language

models are often better at evaluation than generation (Saunders et al., 2022; Bai et al., 2022). For example, the tasks that judge factual alignment with a video or detect redundancy are often easier than producing accurate and non-redundant content. Using these critic strengths as reward signals, therefore, yields a higher-quality policy and a strong foundation for Phase Two.

Optimization Algorithm. We adopt Group Sequence Policy Optimization (GSPO) for RL training given its stability and efficiency with sequence-level rewards (Zheng et al., 2025b;a). For each input $q = (v_1, \cdots, v_k, m_1, \cdots, m_{k-1})$, the memorization policy π_θ rolls out a group of G trajectories $\tau_{i=1}^{G}$, where $\tau_i = (q, m_{k,i})$. The reward of the i-th rollout, denoted $r_{\text{mc}}(\tau_i)$, is given by reward models. We then compute the advantage of each trajectory by normalizing rewards within the group:

$$\hat{A}_i = \frac{r_{\text{mc}}(\tau_i) - \text{mean}(\{r_{\text{mc}}(\tau_i)\}_{i=1}^{G})}{\text{std}(\{r_{\text{mc}}(\tau_i)\}_{i=1}^{G})}. \tag{1}$$

The optimization objective is:

$$\mathcal{J}_{\text{GSPO}}(\theta) = \mathbb{E}_{q\sim\mathcal{D},\{\tau_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \left[\frac{1}{G} \sum_{i=1}^{G} \min\left(s_i(\theta)\hat{A}_i, \text{clip}(s_i(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_i \right) \right], \tag{2}$$

where the importance ratio $s_i(\theta)$ is defined as $s_i(\theta) = \left(\frac{\pi_\theta(m_{k,i}|q)}{\pi_{\theta_{\text{old}}}(m_{k,i}|q)} \right)^{\frac{1}{|m_{k,i}|}}$.

Multi-Objective Reward Design. We define the trajectory-level reward $r_{\text{mc}}(\tau)$ as follows:

$$r_{\text{mc}}(\tau) = r_{\text{fmt}}(\tau) + r_{\text{len}}(\tau) + r_{\text{qual}}(\tau) + r_{\text{rich}}(\tau). \tag{3}$$

Here, r_{fmt} evaluates whether the output follows the predefined format. Following the ReAct (Yao et al., 2023) paradigm, the policy generates reasoning before memory, and r_{len} is a soft penalty on overlong intermediate reasoning to regularize computational overhead. r_{qual} measures the quality of the generated memory in terms of accuracy, non-redundancy, and style.

Optimizing solely for quality can lead the memorization policy to hack the objective, generating outputs that are accurate but lack substantive content (see examples in Table 13). To address this issue, we introduce a richness reward $r_{\text{rich}}(\tau)$, which explicitly encourages the generation of content-rich memories. Richness is defined relatively within each group $\{\tau_i\}_{i=1}^{G}$. A reward model ranks each sampled memory by richness, and these rankings are then converted into scalar rewards. Detailed definitions of all reward components, including prompts and scoring rules, are in Appendix C.

2.3 Phase Two: What to Memorize

Once deployed in an environment, the agent should generate task-relevant memory. To achieve this, Phase Two employs online learning driven by environmental feedback, allowing the agent to periodically update its parameters for timely adaptation. The core intuition mirrors how humans refine behavior based on experience: the agent leverages recent tasks to model the likely distribution of future tasks and accordingly adjusts what to memorize next. Specifically, within each update window, TaskMem utilizes the most recent n tasks to optimize memory generation.

Phase Two training presents several challenges. (1) Sparse feedback: TaskMem must adapt its memorization policy using only a few recent tasks (e.g., ten questions), limiting the learning signal. (2) Catastrophic forgetting: Updating the policy parameters risks overwriting or degrading previously learned capabilities. (3) Computational efficiency: Since adaptation occurs at deployment time, updates must be both fast and resource-efficient. We next describe how our design addresses them.

Feedback Augmentation. To address sparse feedback, we introduce a reward model that takes a set of example tasks and two candidate memories as input. The model first infers the underlying intent of the example tasks, and then determines which memory is more relevant to the corresponding environment, or whether both are similarly relevant.

We pre-construct a dataset of rollouts sampled from the Phase One memorization policy. Specifically, for each input $q = (v_{1:k}, m_{1:k-1})$, we roll out a set of candidate memories. At deployment time, this pre-computed dataset allows us to efficiently construct pairwise preference data (q, m_k^w, m_k^l) by

leveraging the reward model to compare candidate memories. These pairwise comparisons are then used to optimize the memorization policy, encouraging it to generate more relevant memories.

Model Architecture. To address catastrophic forgetting and computational efficiency, we adopt a simple parameter-efficient tuning (Hu et al., 2023; Ding et al., 2023) approach. As shown in Figure 2, we introduce an additive, trainable vector adapter at a selected transformer layer, defined as

$$h_o \leftarrow h_o + a. \tag{4}$$

Here, $h_o \in \mathbb{R}^d$ is the layer output and $a \in \mathbb{R}^d$ is a learnable vector, only inserted at a single layer.

Optimization Algorithm. After augmenting the task signals to pairwise data, we adopt the Direct Preference Optimization (DPO) algorithm (Rafailov et al., 2023). The training objective is

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(q, m_k^w, m_k^l)\sim\mathcal{D}} \left[\log \sigma \left(\beta \log \frac{\pi_\theta(m_k^w|q)}{\pi_{\text{ref}}(m_k^w|q)} - \beta \log \frac{\pi_\theta(m_k^l|q)}{\pi_{\text{ref}}(m_k^l|q)} \right) \right]. \tag{5}$$

Here, π_{ref} is the policy obtained from Phase One training. During Phase Two, we update only the adapter parameters, while keeping the backbone model fixed.

3 Experiments

We reformulate VQA benchmarks into streaming task-specific environments by grouping QA pairs of the same type, assessing whether TaskMem learns to generate task-relevant memory. We first describe the two-phase training details and dynamics, showing how Phase One establishes basic capabilities and Phase Two learns to produce task-relevant memories. We then report VQA performance from memory alone, demonstrating its effectiveness for tasks in the environment.

3.1 Training Details

3.1.1 Phase One

We fine-tune Qwen3-VL-30B-A3B (Bai et al., 2025a) as the memorization policy using GSPO in two stages, which differ only in how the historical memories $m_{1:k-1}$ are constructed as training context. Because on-policy rollout is expensive, the first stage uses Gemini-2.5-Pro (Comanici et al., 2025) to synthesize trajectories $m_{1:k-1}$ for video clips $v_{1:k}$ sampled from an in-house long-video dataset $\mathcal{D}$ (see prompt in Table 14). We call this stage off-policy history training. However, this introduces a train–test distribution gap, since at deployment every historical memory is produced by the memorization policy itself. To close this gap, we then perform on-policy history training, continuing GSPO with histories rolled out from the current policy (Algorithm 1). Our training uses 326 long videos from $\mathcal{D}$, with an average of 25.15 clips extracted per video. See Appendix E for training hyperparameters, reward model implementation, and other details.

Training Stabilization. We extend Zhu et al. (2025) to a multi-valued reward setting by decomposing the learning signal into two components: Positive Sample Reinforce (PSR), which reinforces responses with positive advantage, and Negative Sample Reinforce (NSR), which penalizes those with negative advantage. We find that once the policy achieves a relatively high average reward, NSR introduces training instability, as shown in Figure 3. To mitigate this, we adopt a simple strategy: NSR is disabled for samples with positive reward, which improves stability. Let $\ell_i = \min(s_i(\theta)\hat{A}_i, \text{clip}(s_i(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_i)$. The resulting objective is:

$$\mathcal{J}_{\text{GSPO}}(\theta) = \mathbb{E}_{q\sim\mathcal{D},\, \{\tau_i\}_{i=1}^{G}\sim\pi_{\theta_{\text{old}}}(\cdot|q)} \frac{1}{G} \Bigg[\underbrace{\sum_{\hat{A}_i>0} \ell_i}_{\text{PSR}} + \underbrace{\sum_{\hat{A}_i<0} \mathbb{I}[r_{\text{mc}}(\tau_i) < 0]\ell_i}_{\text{NSR}} \Bigg], \tag{6}$$

where $\mathbb{I}[\cdot]$ returns 1 if true and 0 otherwise.

Richness Reward Discussion. Figure 3 compares Phase One training with and without the richness reward. Without the richness reward, training appears more stable and achieves a higher reward, but the generated memories rapidly become shorter. This indicates reward hacking: the policy satisfies quality checks with shorter, less substantive outputs. The richness reward mitigates this behavior by favoring candidates with more non-redundant and useful content (Table 13).

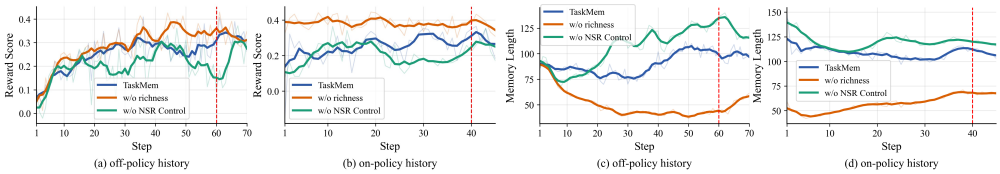


Figure 3: Training performance comparison across three settings: TaskMem, TaskMem w/o the richness reward, and TaskMem w/o NSR control. (a) and (b) report reward trajectories during off-policy and on-policy history training; (c) and (d) report the corresponding memory-length changes.

Table 1: Comparison of Phase Two training strategies on memory quality and task relevance.

Method	**Accuracy** ↑	**Non-redundancy** ↑	**Loss Ratio** ↓	**Tie Ratio**	**Win Ratio** ↑
TaskMem Phase One	92.39 ± 0.47	83.59 ± 0.43	13.82 ± 1.15	72.80 ± 0.83	13.38 ± 0.37
w/ full-parameter tuning	91.29	78.21	2.00	58.00	40.00
w/ 40-step adapter	92.93	81.74	3.06	61.72	35.22
w/ scaled adapter	91.54	80.96	3.17	58.52	38.31

3.1.2 PHASE TWO

We first construct a dataset of (video, rollouts) pairs, with rollouts sampled from the Phase One policy π_0. During deployment, the agent collects real tasks and leverages the reward model to transform sparse feedback into pairwise preference data. We then perform DPO on this dataset, training only a lightweight adapter while keeping the base MLLM frozen (Appendix F).

To track and evaluate training dynamics, we build a validation set consisting of trajectories $(v_{1:k}, m_{1:k-1})$. The video clips are disjoint from the training set, and the historical memories are generated by π_0. We evaluate performance using three metrics: (1) Accuracy, whether the generated memory m_k aligns with the video content; (2) Non-redundancy rate, whether m_k avoids duplicating information already present in the historical memory; (3) Relevance win/tie/loss ratio, how often m_k is more relevant, equally relevant, or less relevant to tasks in the environment compared with the memory generated by π_0. Implementation details for these metrics are provided in Appendix F.4.

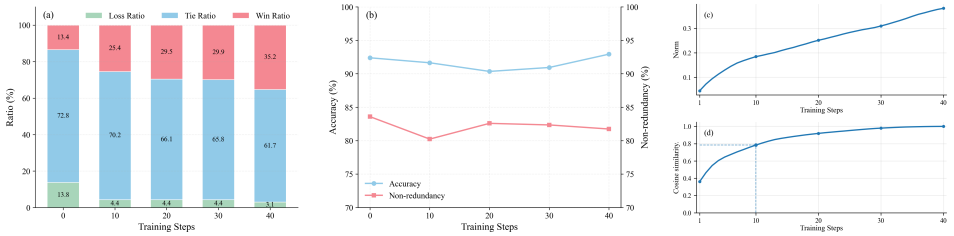


Figure 4: Phase Two training dynamics for object recognition on VideoMME, using ten streaming-task questions as feedback: (a) task relevance improves; (b) memory quality stays stable; (c) adapter norm grows steadily; (d) cosine similarity to the final 40-step adapter.

Training Dynamics. As shown in Figure 4, Phase Two improves task relevance while maintaining stable accuracy and low redundancy. The adapter direction converges early, with cosine similarity between step-10 and step-40 reaching 0.8, indicating that later training primarily increases its norm. Motivated by this, we propose a simple acceleration strategy: directly scale the step-10 adapter to a target norm (0.3) to approximate the final adapter. As shown in Table 1, the scaled adapter achieves performance comparable to step-40, while reducing training data and time by approximately 75%.

Layer Ablation. We examine how adapter placement across layers affects performance. Figure 5 reports the accuracy, non-redundancy rate, and win ratio for adapters inserted at different layers on the VideoMME object recognition task. The results show that placing adapters in shallow and middle layers is more effective than in deep layers, and we place the adapter at layer 22 in all experiments.

Adapter Training Discussion. We motivate the use of an adapter vector from three perspectives. First, it is lightweight and parameter-efficient, requiring only 2,048 trainable parameters. As shown in Table 1, compared with full-parameter training, the adapter training preserves accuracy and non-redundancy rates close to the original model, indicating resistance to catastrophic forgetting.



Figure 5: Ablation study on adapter placement across layers on VideoMME object recognition task.

Second, from an inference perspective, the adapter can function as a group of parameters loaded only at inference time, leaving the base model weights unchanged. It incurs no additional serving cost. Conceptually, the adapter can be viewed as a form of parametric personalized memory.

Third, prior works show that LLM behaviors can be steered at inference time by adding a vector to layer activations (Turner et al., 2023; Panickssery et al., 2023; Arditi et al., 2024; Chen et al., 2025b). This is because high-level concepts are approximately encoded as linear directions in activation space (Elhage et al., 2022; Park et al., 2023). Following the same intuition, we learn such a direction directly, enabling the resulting vector to capture task-specific representations in activation space.

3.2 VQA Evaluation Setup

We evaluate TaskMem on VideoMME (Fu et al., 2025), EgoLife (Yang et al., 2025), and EgoTempo (Plizzari et al., 2025) by reformulating each VQA benchmark as streaming task-specific environments. For each benchmark, video-question pairs are grouped by question type, and each group is treated as one task environment. Within each task environment, videos are segmented into 10-second clips and processed sequentially. The memorization policy operates over a sliding window of the most recent $k = 5$ clips and their preceding memories. The question is revealed only after its corresponding video has been processed, ensuring memory generation is independent of the question. Dataset statistics are provided in Appendix D.1. To isolate memory quality, answers are generated solely from memory, without access to the original video. We use GPT-4o as the answer generator. Given a question and memory, it first checks whether sufficient information is present; if not, it returns "insufficient information"; otherwise, it produces an answer. Due to context-length limits, we evaluate only the short and medium subsets of VideoMME. The prompt is provided in Table 20.

For each task stream, the memorization policy is initialized with the Phase One policy (π_0). The first 10 questions are answered with memories from π_0 and then used for Phase Two training. The remaining questions are answered with memories from the updated policy.

We report Accuracy (Acc.), Coverage (Cov.), and Precision (Prec.) for each benchmark. Accuracy is overall downstream utility; Coverage is how often memory suffices to answer; and Precision is reliability among answerable questions. Detailed computation is provided in Appendix D.2.

Baselines. We compare against two categories. (1) Base MLLMs: Gemini-1.5-Pro (Team et al., 2024), Gemini-2.5-Pro (Comanici et al., 2025), GPT-5.2 (OpenAI, 2025), and Qwen3-VL-30B-A3B (Bai et al., 2025a). Each model follows the streaming protocol of TaskMem, receiving identical video clips but conditioning on its own previously generated memories. (2) Memory Frameworks: EgoGPT (Yang et al., 2025), HippoMM (Lin et al., 2025), and M3-Agent (Long et al., 2025). Their episodic memories are produced by the official pipelines and passed to the same answer generator. Baseline setup and prompts are detailed in Appendix D.3.

3.3 Main Results for VQA

Table 2 compares TaskMem with base MLLMs and existing memory frameworks on VideoMME, EgoLife, and EgoTempo, with per-task breakdowns in Appendix G.1. Compared with Qwen3-VL-30B-A3B, TaskMem achieves accuracy gains of 6.3, 7.0, and 5.8 points on three benchmarks, and obtains the best performance on VideoMME and EgoLife. On EgoTempo, TaskMem trails GPT-5.2 in accuracy but achieves the highest precision, suggesting that although GPT-5.2 has stronger video

Table 2: Results on VideoMME, EgoLife, and EgoTempo. Best in bold, second-best underlined.

Method	VideoMME			EgoLife			EgoTempo		
	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)
EgoGPT	44.3	58.7	75.5	19.2	28.2	68.1	15.0	33.5	44.9
HippoMM	48.9	66.6	73.5	30.4	43.4	70.0	15.8	30.8	51.1
M3-Agent	62.5	77.7	80.4	21.8	30.8	70.8	16.0	36.3	44.2
Gemini-1.5-Pro	55.3	65.9	83.9	39.4	51.6	76.4	19.7	34.3	57.4
Gemini-2.5-Pro	63.2	74.8	<u>84.4</u>	<u>43.8</u>	56.6	<u>77.4</u>	25.8	42.3	61.0
GPT-5.2	<u>67.3</u>	80.8	83.3	34.8	48.2	72.2	**32.1**	51.4	<u>62.4</u>
Qwen3-VL-30B-A3B	61.6	74.7	82.5	38.4	52.4	73.3	22.3	38.9	57.2
TaskMem	**67.9**	79.3	**85.6**	**45.4**	56.4	**80.5**	<u>28.1</u>	44.1	**65.3**

Table 3: Ablation study of Phase One and Phase Two training.

Method	VideoMME			EgoLife			EgoTempo		
	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)
Qwen3-VL-30B-A3B	61.6	74.7	82.5	38.4	52.4	73.3	22.3	38.9	57.2
+ Task Prompt	64.2	77.3	83.0	<u>40.0</u>	51.0	<u>78.4</u>	<u>24.5</u>	42.9	57.1
TaskMem Phase One	<u>64.4</u>	75.4	<u>85.3</u>	39.6	51.0	77.6	23.7	37.5	<u>63.3</u>
TaskMem	**67.9**	79.3	**85.6**	**45.4**	56.4	**80.5**	**28.1**	44.1	**65.3**

understanding for fine-grained egocentric activities, our trained memory is more reliable and less prone to hallucination. Appendix G.2 shows these gains persist with a different QA answer generator.

We compare TaskMem against a prompt-only baseline, where Qwen3-VL-30B-A3B is given the same recent tasks and prompted to generate task-relevant memory (see prompt in Table 15). As shown in Table 3, prompting improves over the base model but still underperforms TaskMem, indicating that parameter updates are more effective. Ablating the two training phases shows that Phase One improves accuracy and precision by learning more reliable memory, while Phase Two further boosts accuracy by generating more task-relevant memory.

To verify that Phase Two learns task-specific memory focus rather than a generally stronger adapter, we conduct a cross-task transfer test: we fix the evaluation task to Object Recognition VQA and replace its adapter with those trained with other-task feedback (Counting, OCR, Attribute Perception). If Phase Two merely produced a universally better adapter, all variants would improve Object Recognition; if it learns task-specific focus, only the matched adapter should help. As shown in Table 4, only the Object Recognition adapter improves, indicating that Phase Two acquires task-specific focus, rather than a generic cross-task capability.

Table 4: Adapter analysis on object recognition task.

Method	Acc.(↑)	Cov.	Prec.(↑)
TaskMem Phase One	<u>65.7</u>	74.7	<u>87.9</u>
w/ Counting adapter	65.0	74.0	87.8
w/ OCR adapter	65.0	75.7	85.9
w/ Attribute adapter	65.0	74.3	87.4
w/ Object adapter	**69.7**	78.7	**88.6**

3.4 Case Study

Task-Focused Memory in Phase Two. We compare how episodic memory evolves from Phase One to Phase Two through several case studies. As shown in Figure 6, Phase One produces accurate but general event-narration memory, while Phase Two, trained with object-recognition questions, shifts toward fine-grained object details. Additional qualitative cases are provided in Table 21.

Complete Episodic Memory on a Long Video. Table 22 presents a complete episodic memory generated by TaskMem for a five-minute video, demonstrating three advantages: (1) detailed yet low-hallucination content, where the policy generates a 4620-token memory with rich and accurate descriptions; (2) coherent streaming generation, where the memory maintains smooth narrative flow across clip boundaries and uses face IDs, e.g., `[face_1]`, for entity tracking; (3) task relevance, where the memory is generated under a counting task and emphasizes local counts within individual clips, helping solve the global counting problem, a representative and challenging long-horizon task.

Video (sampled frames)



Representative task queries

"Which galaxies are depicted in the video?"; "Which object does the holder made in this video visually resemble?"; "Which team scored in the video?"; "What does the chef in the video end up cutting with a knife?"; "What is the second class the boys are taking in the video?"

Episodic memory (Phase One)

The video transitions to the exterior of a thatched-roof building with a swing and chairs on a sandy area, followed by a view of a tented room containing a large bed with white linens, a fan, and a bathroom area with a wooden table, sink, and round mirror. <face_3> stands on a dirt path in the arid landscape, pointing at the ground and stating, "So this is an elephant poop," with sparse trees and dry terrain visible in the background.

Episodic memory (Phase Two)

The video transitions to the exterior of a thatched-roof building with a swing and wooden chairs on a sandy area, surrounded by trees. Inside a tented structure, a person adjusts a screen while another stands nearby. The tent's interior features a large bed with white and pink bedding, a fan, wooden furniture, and large windows overlooking the savanna. A bathroom area within the tent displays a wooden sink with toiletries, a round mirror, and towels hanging. <face_3> then stands on a dirt path, pointing at the ground and stating, "So this is an elephant poop," while wearing a cap and backpack, with the arid landscape and thatched structures in the background.

Figure 6: Episodic memory from Phase One and Phase Two. Highlights mark newly added details.

4 Related Work

Long-term Memory in Multimodal Agents. Multimodal agents require external memory to preserve information beyond the context window (Hu et al., 2025b; Huang et al., 2026). Prior work explores diverse memory paradigms, including memory banks and sparse memories for long-video understanding (He et al., 2024a; Song et al., 2024), structured memories built from textual, object-centric, episodic, semantic, and visual representations (Fan et al., 2024; Long et al., 2025), and advanced systems with continual consolidation, structured retrieval, or neuro-symbolic reasoning, personalization, verification, or agent benchmarks (Chen et al., 2026a; Liu et al., 2025; Jiang et al., 2026). As multimodal observations accumulate, however, maintaining memory quality becomes difficult and can degrade downstream retrieval and reasoning (Lu et al., 2026; Salama et al., 2025). Existing approaches predefine what to store either through prompting (Chen et al., 2026a; Jiang et al., 2026) or post-training (Long et al., 2025; Liu et al., 2025). However, they overlook a critical aspect: what to memorize should dynamically adapt to the environment. An effective multimodal agent should leverage environmental feedback to continuously refine memory formation for both accuracy and task relevance. Enabling such continual adaptation is the central focus of our work.

Test-Time Training. Test-time training (TTT) adapts model parameters at inference time using data related to the current test instance, improving alignment with the test distribution. Originating from local learning (Bottou & Vapnik, 1992), TTT has recently been extended to LLMs through retrieval-based adaptation (Hardt & Sun, 2024) and active data selection (Hübotter et al., 2025). It is also closely related to continual learning (Hung et al., 2019; Douillard et al., 2022), where models are incrementally updated from a data stream (Chen et al., 2025a). A central challenge is catastrophic forgetting (Kirkpatrick et al., 2017; French, 1999; Hu et al., 2025a), where adapting to new data may degrade previously acquired capabilities. Existing TTT methods typically rely on direct adaptation signals from task inputs or self-supervised objectives (Hardt & Sun, 2024; Akyürek et al., 2025). In contrast, our setting involves indirect feedback: a multimodal agent observes task rather than explicit signals of memory quality. TaskMem bridges this gap by converting sparse environment feedback into task-conditioned preferences for memorization.

5 Conclusion

In this paper, we introduce TaskMem, a reinforcement learning framework that trains a memorization policy to generate task-relevant memory. The framework adopts a two-phase design: Phase One optimizes the policy with multi-objective rewards to produce accurate, non-redundant, well-formatted, and content-rich episodic memories. Phase Two further aligns the policy toward more task-relevant content through tuning a lightweight adapter on the base MLLM. Under a streaming VQA setting, TaskMem achieves the best performance on VideoMME and EgoLife, and remains competitive on EgoTempo. Future work will extend TaskMem beyond episodic memory to semantic and visual memory, and apply adaptive memorization to more interactive embodied environments.

AI USE STATEMENT

We used large language models to improve the clarity and readability of the paper, with all edited text reviewed by the authors. In addition, we used LLMs to assist in generating training data and in evaluation: Gemini-2.5-Pro synthesizes the historical memories used as training context, Gemini-2.5-Flash and GPT-4o serve as reward models, and GPT-4o is used as the answer generator in our evaluation. We take full responsibility for the final content of this work.

REFERENCES

Ekin Akyürek, Mehul Damani, Adam Zweiger, Linlu Qiu, Han Guo, Jyothish Pari, Yoon Kim, and Jacob Andreas. The surprising effectiveness of test-time training for few-shot learning. In *International Conference on Machine Learning*, pp. 942–963. PMLR, 2025.

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. *Advances in Neural Information Processing Systems*, 37:136037–136083, 2024.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025a.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Ming-Hsuan Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *CoRR*, abs/2502.13923, 2025b.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

Léon Bottou and Vladimir Vapnik. Local learning algorithms. *Neural computation*, 4(6):888–900, 1992.

Chunliang Chen, Ming Guan, Xiao Lin, Jiaxu Li, Luxi Lin, Qiyi Wang, Xiangyu Chen, Jixiang Luo, Changzhi Sun, Dell Zhang, and Xuelong Li. Telemem: Building long-term and multimodal memory for agentic AI. *CoRR*, abs/2601.06037, 2026a.

Haoran Chen, Micah Goldblum, Zuxuan Wu, and Yu-Gang Jiang. Adaptive retention & correction: Test-time training for continual learning. In *The Thirteenth International Conference on Learning Representations*, 2025a.

Runjin Chen, Andy Arditi, Henry Sleight, Owain Evans, and Jack Lindsey. Persona vectors: Monitoring and controlling character traits in language models. *arXiv preprint arXiv:2507.21509*, 2025b.

Zhisheng Chen, Tingyu Wu, Zijie Zhou, Zhengwei Xie, Ziyan Weng, and Yingwei Zhang. Polarmem: A training-free polarized latent graph memory for verifiable multimodal agents. *CoRR*, abs/2602.00415, 2026b.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, et al. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature machine intelligence*, 5(3):220–235, 2023.

Arthur Douillard, Alexandre Ramé, Guillaume Couairon, and Matthieu Cord. Dytox: Transformers for continual learning with dynamic token expansion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9285–9295, 2022.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. Toy models of superposition. *arXiv preprint arXiv:2209.10652*, 2022.

Yue Fan, Xiaojian Ma, Rujie Wu, Yuntao Du, Jiaqi Li, Zhi Gao, and Qing Li. Videoagent: A memory-augmented multimodal agent for video understanding. In *Computer Vision - ECCV 2024 - 18th European Conference, Milan, Italy, September 29-October 4, 2024, Proceedings, Part XXII*, Lecture Notes in Computer Science, pp. 75–92. Springer, 2024.

Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3 (4):128–135, 1999.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 24108–24118, 2025.

Alex Graves, Greg Wayne, and Ivo Danihelka. Neural turing machines. *arXiv preprint arXiv:1410.5401*, 2014.

Moritz Hardt and Yu Sun. Test-time training on nearest neighbors for large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

Bo He, Hengduo Li, Young Kyun Jang, Menglin Jia, Xuefei Cao, Ashish Shah, Abhinav Shrivastava, and Ser-Nam Lim. MA-LMM: memory-augmented large multimodal model for long-term video understanding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 13504–13514. IEEE, 2024a.

Yichen He, Yuan Lin, Jianchao Wu, Hanchong Zhang, Yuchen Zhang, and Ruicheng Le. Storyteller: Improving long video description through global audio-visual character identification. *arXiv preprint arXiv:2411.07076*, 2024b.

Dan Hendrycks, Dawn Song, Christian Szegedy, Honglak Lee, Yarin Gal, Erik Brynjolfsson, Sharon Li, Andy Zou, Lionel Levine, Bo Han, et al. A definition of agi. *arXiv preprint arXiv:2510.18212*, 2025.

Jinwu Hu, Zitian Zhang, Guohao Chen, Xutao Wen, Chao Shuai, Wei Luo, Bin Xiao, Yuanqing Li, and Mingkui Tan. Test-time learning for large language models. In *Forty-second International Conference on Machine Learning*, 2025a.

Yuyang Hu, Shichun Liu, Yanwei Yue, Guibin Zhang, Boyang Liu, Fangyi Zhu, Jiahang Lin, Honglin Guo, Shihan Dou, Zhiheng Xi, et al. Memory in the age of ai agents. *arXiv preprint arXiv:2512.13564*, 2025b.

Zhiqiang Hu, Lei Wang, Yihuai Lan, Wanyu Xu, Ee-Peng Lim, Lidong Bing, Xing Xu, Soujanya Poria, and Roy Lee. Llm-adapters: An adapter family for parameter-efficient fine-tuning of large language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 5254–5276, 2023.

Wei-Chieh Huang, Weizhi Zhang, Yueqing Liang, Yuanchen Bei, Yankai Chen, Tao Feng, Xinyu Pan, Zhen Tan, Yu Wang, Tianxin Wei, Shanglin Wu, Ruiyao Xu, Liangwei Yang, Rui Yang, Wooseong Yang, Chin-Yuan Yeh, Hanrong Zhang, Haozhen Zhang, Siqi Zhu, Henry Peng Zou, Wanjia Zhao, Song Wang, Wujiang Xu, Zixuan Ke, Zheng Hui, Dawei Li, Yaozu Wu, Langzhou He, Chen Wang, Xiongxiao Xu, Baixiang Huang, Juntao Tan, Shelby Heinecke, Huan Wang, Caiming Xiong, Ahmed A. Metwally, Jun Yan, Chen-Yu Lee, Hanqing Zeng, Yinglong Xia, Xiaokai Wei, Ali Payani, Yu Wang, Haitong Ma, Wenya Wang, Chenguang Wang, Yu Zhang, Xin Wang, Yongfeng Zhang, Jiaxuan You, Hanghang Tong, Xiao Luo, Xue Liu, Yizhou Sun, Wei Wang, Julian J. McAuley, James Zou, Jiawei Han, Philip S. Yu, and Kai Shu. Rethinking memory mechanisms of foundation agents in the second half: A survey. *CoRR*, abs/2602.06052, 2026.

Jonas Hübotter, Sascha Bongni, Ido Hakimi, and Andreas Krause. Efficiently learning at test-time: Active fine-tuning of llms. In *The Thirteenth International Conference on Learning Representations*, 2025.

Ching-Yi Hung, Cheng-Hao Tu, Cheng-En Wu, Chien-Hung Chen, Yi-Ming Chan, and Chu-Song Chen. Compacting, picking and growing for unforgetting continual learning. *Advances in neural information processing systems*, 32, 2019.

Rongjie Jiang, Jianwei Wang, Gengda Zhao, Chengyang Luo, Kai Wang, and Wenjie Zhang. Advancing multimodal agent reasoning with long-term neuro-symbolic memory. *arXiv preprint arXiv:2603.15280*, 2026.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114 (13):3521–3526, 2017.

Yueqian Lin, Jingyang Zhang, Qinsi Wang, Hancheng Ye, Yuzhe Fu, Yudong Liu, Hai Li, Yiran Chen, et al. Hippomm: Hippocampal-inspired multimodal memory for long audiovisual event understanding. *arXiv preprint arXiv:2504.10739*, 2025.

Junming Liu, Yifei Sun, Weihua Cheng, Haodong Lei, Yirong Chen, Licheng Wen, Xuemeng Yang, Daocheng Fu, Pinlong Cai, Nianchen Deng, Yi Yu, Shuyue Hu, Botian Shi, and Ding Wang. Memverse: Multimodal memory for lifelong learning agents. *CoRR*, abs/2512.03627, 2025.

Lin Long, Yichen He, Wentao Ye, Yiyuan Pan, Yuan Lin, Hang Li, Junbo Zhao, and Wei Li. Seeing, listening, remembering, and reasoning: A multimodal agent with long-term memory. *CoRR*, abs/2508.09736, 2025.

Yihao Lu, Wanru Cheng, Zeyu Zhang, and Hao Tang. Mma: Multimodal memory agent. *arXiv preprint arXiv:2602.16493*, 2026.

OpenAI. Gpt-5.2. https://openai.com/index/introducing-gpt-5-2/, 2025.

Nina Panickssery, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering llama 2 via contrastive activation addition. *arXiv preprint arXiv:2312.06681*, 2023.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*, 2023.

Chiara Plizzari, Alessio Tonioni, Yongqin Xian, Achin Kulshrestha, and Federico Tombari. Omnia de egotempo: Benchmarking temporal understanding of multi-modal llms in egocentric videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 24129–24138, 2025.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Rana Salama, Jason Cai, Michelle Yuan, Anna Currey, Monica Sunkara, Yi Zhang, and Yassine Benajiba. Meminsight: Autonomous memory augmentation for LLM agents. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pp. 33136–33152. Association for Computational Linguistics, 2025.

William Saunders, Catherine Yeh, Jeff Wu, Steven Bills, Long Ouyang, Jonathan Ward, and Jan Leike. Self-critiquing models for assisting human evaluators. *arXiv preprint arXiv:2206.05802*, 2022.

Murray Shanahan. The frame problem. https://plato.stanford.edu/entries/frame-problem/, 2004.

Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, Yan Lu, Jenq-Neng Hwang, and Gaoang Wang. Moviechat: From dense token to sparse memory for long video understanding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pp. 18221–18232. IEEE, 2024.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

Gemini Robotics Team, Saminda Abeyruwan, Joshua Ainslie, Jean-Baptiste Alayrac, Montserrat Gonzalez Arenas, Travis Armstrong, Ashwin Balakrishna, Robert Baruch, Maria Bauza, Michiel Blokzijl, et al. Gemini robotics: Bringing ai into the physical world. *arXiv preprint arXiv:2503.20020*, 2025.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering. *arXiv preprint arXiv:2308.10248*, 2023.

Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE transactions on pattern analysis and machine intelligence*, 46(8):5362–5383, 2024a.

Zihao Wang, Shaofei Cai, Anji Liu, Yonggang Jin, Jinbing Hou, Bowei Zhang, Haowei Lin, Zhaofeng He, Zilong Zheng, Yaodong Yang, et al. Jarvis-1: Open-world multi-task agents with memory-augmented multimodal language models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(3):1894–1907, 2024b.

Zixuan Wang, Bo Yu, Junzhe Zhao, Wenhao Sun, Sai Hou, Shuai Liang, Xing Hu, Yinhe Han, and Yiming Gan. Karma: Augmenting embodied ai agents with long-and-short term memory systems. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1–8. IEEE, 2025.

Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, et al. Qwen3-omni technical report. *arXiv preprint arXiv:2509.17765*, 2025.

Jingkang Yang, Shuai Liu, Hongming Guo, Yuhao Dong, Xiamengwei Zhang, Sicheng Zhang, Pengyun Wang, Zitang Zhou, Binzhu Xie, Ziyue Wang, et al. Egolife: Towards egocentric life assistant. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 28885–28900, 2025.

Yijun Yang, Tianyi Zhou, Kanxue Li, Dapeng Tao, Lusong Li, Li Shen, Xiaodong He, Jing Jiang, and Yuhui Shi. Embodied multi-modal agent trained by an llm from a parallel textworld. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26265–26275. IEEE, 2024.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

Woongyeong Yeo, Kangsan Kim, Jaehong Yoon, and Sung Ju Hwang. Worldmm: Dynamic multimodal memory agent for long video reasoning. *CoRR*, abs/2512.02425, 2025. doi: 10.48550/ARXIV.2512.02425. URL https://doi.org/10.48550/arXiv.2512.02425.

Chujie Zheng, Kai Dang, Bowen Yu, Mingze Li, Huiqiang Jiang, Junrong Lin, Yuqiong Liu, Hao Lin, Chencan Wu, Feng Hu, et al. Stabilizing reinforcement learning with llms: Formulation and practices. *arXiv preprint arXiv:2512.01374*, 2025a.

Chujie Zheng, Shixuan Liu, Mingze Li, Xiong-Hui Chen, Bowen Yu, Chang Gao, Kai Dang, Yuqiong Liu, Rui Men, An Yang, et al. Group sequence policy optimization. *arXiv preprint arXiv:2507.18071*, 2025b.

Xinyu Zhu, Mengzhou Xia, Zhepei Wei, Wei-Lin Chen, Danqi Chen, and Yu Meng. The surprising effectiveness of negative reinforcement in llm reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

A LIMITATIONS

In this work, we focus on validating TaskMem in the setting of episodic memory, where the agent records temporally ordered event-centric experiences from streaming multimodal observations. Extending task-focused memorization to broader memory forms, such as semantic memory, visual memory, and structured user-centric memory, remains an important direction for future work. In addition, environmental feedback in practical agent scenarios can be highly diverse. Our current Phase Two design adopts QA-style downstream tasks as a relatively accessible and measurable feedback signal for learning task relevance. Future work may incorporate other forms of feedback, such as natural language instructions, interaction outcomes, tool-use traces, task success signals, and implicit user preferences, to further improve the generality and adaptability of the memorization policy.

B IMPLEMENTATION DETAILS OF TOOLS

Here, we provide the implementation details of the tools for representation extraction introduced in Section 2.1.

Facial Recognition. For face recognition, we follow the previous work (Long et al., 2025) to obtain bounding boxes and filter low-quality faces. Then, we draw the detected bounding boxes in the video. For each 1-second clip in the video, we select the middle frame, draw bounding boxes around all detected faces, and extend each bounding box forward and backward in time by 0.5 seconds, respectively. We also annotate the corresponding face ID in the upper-left corner of each bounding box.

Voice Identification. For speech recognition, we use Gemini-2.5-Pro to extract audio segments and identify speakers. Speaker identification is performed based on the current clip. The model matches the recognized audio segments with the face ID of their most likely corresponding speaker. Meanwhile, for segments where the speaker cannot be identified, the model retrieves similar voices from the global memory and labels them with the corresponding face ID. Finally, the speech content, together with the speaker's face ID, is displayed as subtitles at the bottom of the video. The prompts used for speech recognition and speaker identification are shown in Table 5 and Table 6.

The Prompt for Speech Recognition

You are given a video with a total duration of 10 seconds. Your task is to perform Automatic Speech Recognition (ASR) and audio diarization on the provided video. Extract all speech segments with accurate timestamps and segment them by speaker turns (i.e., different speakers should have separate segments), but without assigning speaker identifiers.

Return a JSON list where each entry represents a speech segment with the following fields:
- start_time: Start time in seconds, represented as a floating-point number, accurate to 0.1s.
- end_time: End time in seconds, represented as a floating-point number, accurate to 0.1s.
- asr: The transcribed text for that segment.

Example Output:
```json
[
    {"start_time": 5.3, "end_time": 6.9, "asr": "Hello, everyone."},
    {"start_time": 9.2, "end_time": 11.6, "asr": "Welcome to the meeting."}
]
```

Strict Requirements:

- Ensure precise speech segmentation with accurate timestamps.
- Adjacent sentences need to be separated. Each list item can only have one sentence.
- Preserve punctuation and capitalization in the ASR output.

(Continued on next page)

- Skip the speeches that can hardly be clearly recognized or extremely SHORT in time.
- Return only the valid JSON list (which starts with "["and ends with "]") without additional explanations.
- If the video contains no speech, return an empty list ("[]").

Now generate the JSON list based on the given video:

Table 5: Prompt used for automatic speech recognition.

The Prompt for Speaker Identification.

You are given a video. Your task is to match the subtitle with the <face_id> of its speaker.
The subtitle to be matched is given in the following JSON list:
```json
{subtitles}
```

The returned list must have the same length as the input JSON list.
Each item in the list shall include an additional string field named "speaker", with the value determined as follows:
- If the corresponding subtitle is definitively associated with a <face_id>, set "speaker" to that <face_id>;
- Otherwise, set "speaker" to the string literal "unknown".

Example Output:
```json
[
    {"start_time": 5.3, "end_time": 6.9, "asr": "Hello, everyone.", "speaker": "<face_1>"},
    {"start_time": 9.2, "end_time": 11.6, "asr": "Welcome to the meeting.", "speaker": "unknown"}
]
```

Now generate the JSON list based on the given video:

Table 6: Prompt used for automatic speaker identification.

C Reward Design in Phase One

The reward components are already scaled in their definitions and are summed directly to obtain r_{mc}. We provide the detailed reward definitions used in Phase One training. The reward is computed with a format-gated procedure. If the generated trajectory violates the required output format, it receives only a format penalty, with all other reward components set to zero. Otherwise, it is evaluated using the thinking length, quality, and richness rewards. This hierarchical design treats format correctness as a prerequisite, since malformed outputs cannot be reliably parsed or evaluated. For format-valid outputs, the quality reward serves as the primary learning signal, encouraging memories to be factually grounded, contextually coherent, non-redundant, well-formed, and compliant with the memory budget. The thinking-length reward discourages unnecessarily long intermediate reasoning traces, while the richness reward provides a small auxiliary signal for high-quality candidates, favoring non-redundant, content-rich episodic information. This prevents the policy from exploiting the quality reward by producing short but uninformative memories.

Format reward. The model is required to output an intermediate reasoning trace and a final memory in a predefined structure. The reasoning trace must be enclosed within the required thinking tags, and

the final memory must follow the specified output schema. We use a binary format reward:

$$r_{\text{fmt}}(\tau) = \begin{cases} 0, & \text{if the output format is valid,} \\ -1.5, & \text{otherwise.} \end{cases} \tag{7}$$

Thinking length reward. The thinking length reward is applied only when the output format is valid. It penalizes overlong intermediate reasoning traces, which helps control computational overhead and discourages the policy from allocating excessive tokens to reasoning rather than memory content. Let ℓ_{think} denote the token length of the reasoning trace, and let L_{think} be the length threshold. We define:

$$r_{\text{len}}(\tau) = \begin{cases} 0, & \ell_{\text{think}} \le L_{\text{think}}, \\ \lambda_{\text{len}} \dfrac{\ell_{\text{think}} - L_{\text{think}}}{L_{\text{think}}}, & L_{\text{think}} < \ell_{\text{think}} < 2L_{\text{think}}, \\ \lambda_{\text{len}}, & \ell_{\text{think}} \ge 2L_{\text{think}}, \end{cases} \tag{8}$$

where $\lambda_{\text{len}} = -1.0$. In all experiments, we set $L_{\text{think}} = 1200$.

Quality reward. The quality reward is the primary learning signal for Phase One. It evaluates whether the generated memory is factually grounded in the video, coherent with the previous memory context, non-redundant, textually well-formed, and compliant with the memory token budget. A candidate is considered quality-valid only if it satisfies all quality criteria. We use a binary quality reward:

$$r_{\text{qual}}(\tau) = \begin{cases} 0.5, & \text{if the memory satisfies all quality criteria,} \\ -0.5, & \text{otherwise.} \end{cases} \tag{9}$$

Richness reward. The richness reward is applied only to candidates that are both format-valid and quality-valid. It encourages the policy to generate memories with more non-redundant and content-rich information, rather than short but acceptable outputs.

Given a rollout group $\{\tau_i\}_{i=1}^{G}$, where $\tau_i = (q, m_{k,i})$, let $\mathcal{V}$ denote the subset of valid candidates:

$$\mathcal{V} = \{i \in \{1, \dots, G\} \mid r_{\text{fmt}}(\tau_i) = 0 \text{ and } r_{\text{qual}}(\tau_i) \ge \gamma_{\text{qual}}\}. \tag{10}$$

Here, $r_{\text{fmt}}(\tau_i) = 0$ indicates that the output satisfies the required format, and γ_{qual} is the threshold for quality validity. Here γ_{qual} is set as 0.

Let $G' = |\mathcal{V}|$. For each $i \in \mathcal{V}$, a reward model ranks $m_{k,i}$ by richness among the valid candidates, where $\text{rank}(m_{k,i}) = 1$ indicates the richest memory. When $G' > 1$, we convert the rank into a normalized score:

$$u_i = 1 - \frac{\text{rank}(m_{k,i}) - 1}{G' - 1}. \tag{11}$$

The richness reward is then defined as

$$r_{\text{rich}}(\tau_i) = \begin{cases} \lambda_{\text{rich}} u_i, & i \in \mathcal{V} \text{ and } G' > 1, \\ 0, & \text{otherwise,} \end{cases} \tag{12}$$

where $\lambda_{\text{rich}} = 0.05$.

D Dataset and Evaluation Details

D.1 Datasets

We evaluate our approach on three VQA benchmarks:

- **VideoMME (Fu et al., 2025):** The videos in VideoMME are sourced from YouTube and provide comprehensive coverage of diverse video types. Its question categories span twelve tasks, such as object recognition, action reasoning, and counting. Because the memory length of long videos may exceed GPT-4o's maximum token limit, we use only the short and medium subsets of VideoMME. In total, these subsets contain 600 videos and 1,800 QA pairs.

- **EgoLife (Yang et al., 2025):** EgoLife consists of egocentric videos depicting practical, everyday activities. Its questions cover five task types: event recall, relation map, entity log, habit insight, and task master. The dataset contains 500 VQA samples.
- **EgoTempo (Plizzari et al., 2025):** EgoTempo is an egocentric VQA benchmark featuring temporal understanding. The benchmark contains 500 VQA samples spanning ten task types, such as locating object, spatial relationship, and future action prediction. Four samples were excluded due to video retrieval failures, leaving 496 samples in our evaluation.

D.2 Evaluation Metric Computation

For each question, the answer generator is given only the generated memory. It first determines whether the memory contains sufficient evidence to answer the question. If sufficient evidence is available, the answer generator selects one of the candidate options; otherwise, it returns an insufficient-information response. This design prevents the answer generator from making arbitrary guesses when the memory does not preserve the information required by the question. For multiple-choice evaluation, we encode insufficient-information responses as an additional option E, which is treated as an unanswered case rather than a regular answer option.

Let N denote the total number of questions, N_{correct} denote the number of correctly answered questions, and N_{ans} denote the number of answered questions, i.e., the number of predictions not equal to option E. We compute the metrics as

$$\text{Accuracy} = \frac{N_{\text{correct}}}{N}, \qquad \text{Coverage} = \frac{N_{\text{ans}}}{N}, \qquad \text{Precision} = \frac{N_{\text{correct}}}{N_{\text{ans}}}.$$

Accuracy measures the overall fraction of correctly answered questions, Coverage measures how often the generated memory contains enough information to attempt an answer, and Precision measures correctness among the answered questions.

D.3 Baseline Setup

All baselines are evaluated under the same streaming memory-only VQA protocol. During memory generation, each method has access to the video stream and aligned ASR transcripts, but not to future questions. After the video is processed, the generated memories are used as the only input to the answer generator.

Base MLLMs. We evaluate Gemini-1.5-Pro (Team et al., 2024), Gemini-2.5-Pro (Comanici et al., 2025), GPT-5.2 (OpenAI, 2025), and Qwen3-VL-30B-A3B (Bai et al., 2025a) as streaming memory generators under the same setup as TaskMem. At each step, the model receives the current clip, recent clips, and previously generated memories within the same sliding window as TaskMem, and is prompted to generate memory for the current clip. Prompt templates are provided in Table 14.

Memory Frameworks. We further compare with EgoGPT (Yang et al., 2025), HippoMM (Lin et al., 2025), and M3-Agent (Long et al., 2025) as memory framework baselines. For EgoGPT and M3-Agent, we use the official implementations and released checkpoints. For HippoMM, we follow its official memory construction pipeline and use Qwen2.5-VL-7B-Instruct (Bai et al., 2025b) as the underlying vision-language model. All memory framework baselines process videos sequentially as non-overlapping 30-second clips, and their generated memories are ordered by timestamp before being passed to the answer generator.

E Phase One Training Details

E.1 On-policy history training

Algorithm 1 details the on-policy history training procedure used in Phase One. The algorithm maintains a dynamic batch of partial video trajectories. Each trajectory stores the currently visible clips, the generated memories, the corresponding long-video identifier, and a clip counter c_j. At each iteration, the current policy samples candidate memory actions, which are scored by the reward models and optimized with the GSPO objective. After the policy update, the batch is refreshed by extending, retaining, or resampling trajectories according to c_j, so that training covers both longer temporal contexts and diverse videos.

Algorithm 1 Phase One On-Policy History Training Algorithm

Require: Video dataset $\mathcal{D}$, batch size B, keeping probability p, maximum video clips $K = 5$, clip thresholds n_{min}, n_{max}, training steps T.
Ensure: Memory policy π_θ.

for $j = 1$ **to** B **do** ▷ Initialization
  Sample a video v from $\mathcal{D}$ and a random starting point i.
  $x_j \leftarrow (v_i), \quad c_j \leftarrow 1$ ▷ c_j: processed clip counter
end for

for $t = 1$ **to** T **do** ▷ Training loop
  for each $x_j = (v_i, \cdots, v_{i+k}, m_i, \cdots, m_{i+k-1})$ in batch **do** ▷ GSPO optimization on current batch
    Sample a group of memories $M_{i+k}^{(j)} = \{m_{i+k,1}, \cdots, m_{i+k,G}\}$.
    Compute reward and advantage of each memory in $M_{i+k}^{(j)}$.
  end for
  Update policy parameters θ using the GSPO algorithm.

  for each (x_j, c_j) in batch **do** ▷ Batch data evolution
    if $c_j < n_{\text{min}}$ **then**
      EXTEND(j) ▷ Always retain below lower threshold
    else if $c_j > n_{\text{max}}$ **then**
      RESAMPLE(j) ▷ Always discard above upper threshold
    else
      With probability p: EXTEND(j); otherwise RESAMPLE(j)
    end if
  end for
end for

return π_θ

function EXTEND(j) ▷ Retain and extend
  Select a $\hat{m}_{i+k}$ uniformly from $M_{i+k}^{(j)}$.
  if $k + 1 < K$ **then**
    $x_j \leftarrow (v_i, \cdots, v_{i+k+1}, m_i, \cdots, m_{i+k-1}, \hat{m}_{i+k})$
  else ▷ Slide window to keep context $\leq K$
    $x_j \leftarrow (v_{i+1}, \cdots, v_{i+k+1}, m_{i+1}, \cdots, m_{i+k-1}, \hat{m}_{i+k})$
  end if
  $c_j \leftarrow c_j + 1$
end function

function RESAMPLE(j) ▷ Discard and re-sample
  Sample new video $v \sim \mathcal{D}$ and a random starting point i.
  $x_j \leftarrow (v_i), \quad c_j \leftarrow 1$
end function

E.2 Reward Model Implementation

The format and thinking length rewards are rule-based. We first parse the model output into an intermediate reasoning trace and a final memory field. The format checker verifies whether the reasoning trace is enclosed within the required thinking tags and whether the final memory is a valid JSON object following the required schema in Table 14. If this check fails, the remaining reward components are skipped. Otherwise, the thinking length reward is computed deterministically from the token length of the reasoning trace according to Eq. 8.

The quality reward is implemented using external evaluators together with rule-based checks. We use Gemini-2.5-Flash to assess faithfulness to the visual content and contextual coherence, and GPT-4o to check textual validity and non-redundancy. Compliance with the memory token budget is enforced deterministically. A candidate is marked as quality-valid only if it passes all these checks. The prompts are given in Tables 16 and 17.

The richness reward is implemented by prompting GPT-4o to rank the format-valid and quality-valid memories within each rollout group according to the amount of non-redundant, visually grounded, and content-rich episodic information they preserve. The ranking is converted into scalar rewards according to Eq. 12, with the prompt shown in Table 18. All evaluator prompts are fixed across experiments, and all reward queries use deterministic decoding.

E.3 Training Hyperparameters of GSPO

We show the training hyperparameters of GSPO for Phase One in Table 7.

Table 7: Hyperparameters used in Phase One GSPO training.

Parameter Name	Value
Batch Size B	32
Mini Batch Size	8
GPU with 80GB memory	32
Number of Samples in a Group G	8
Learning Rate	1×10^{-6}
Maximum Video Clips K	5
Keeping Probability p	0.15
Lower Clip Threshold $n_{\min}$	10
Upper Clip Threshold $n_{\max}$	20
Training Steps T	100
GSPO Clipping Parameter ϵ_{clip}	0.28
Memory Token Budget L_{mem}	150

F Phase Two Training Details

F.1 Rollout Cache Construction

After the Phase One policy π_0 is fixed, we precompute a rollout cache for Phase Two training. For each streaming context $q = (v_{1:k}, m_{1:k-1})$, where the historical memories are generated by π_0, we sample N candidate memories from π_0 for the current clip and store $\langle q, \mathcal{Y}(q) \rangle$, where $\mathcal{Y}(q) = \{m_{k,j}\}_{j=1}^{N}$. This cache is constructed once and kept fixed during Phase Two. When adapting to a task environment, we use the recent questions observed in that environment to construct task-relevance preferences over the cached candidates, without resampling candidates or using ground-truth answers.

F.2 Task-Relevance Reward Model

We define the Phase Two task reward model as a task-relevance pairwise preference evaluator. Given a set of recent downstream questions $\mathcal{T}$ from the current environment and two candidate memories $x, y \in \mathcal{Y}(q)$ generated for the same context q, the evaluator compares which memory has higher task relevance under the task distribution represented by T:

$$R_{\text{task}}(x, y; \mathcal{T}) \in \{x \succ y,\ y \succ x,\ x \sim y\}.$$

Here, $x \succ y$ indicates that x is judged more task-relevant than y under T, and $x \sim y$ denotes a tie. To reduce presentation-order bias, each candidate pair is evaluated twice with the order swapped. We keep a preference only when the two evaluations are consistent and non-tied; otherwise, the comparison is discarded. Unlike the Phase One rewards, R_{task} is not used as an absolute scalar reward, but only provides relative preferences between candidate memories. In implementation, we instantiate the evaluator by prompting GPT-4o to compare the task relevance of two candidate memories for supporting the questions in $\mathcal{T}$, without access to ground-truth answers. The corresponding prompt is provided in Table 19.

F.3 Preference Data Construction

For each context q, we construct one DPO training pair (q, m_k^w, m_k^l) from the cached candidate memories $\mathcal{Y}(q)$. The selected pair should satisfy two requirements. First, the two memories should have a clear difference in task relevance under the recent task set $\mathcal{T}$. Second, they should avoid substantial differences in basic memory quality, so that the preference mainly reflects task relevance rather than general memory quality differences.

We construct the preference data in two steps. First, we use R_{task} to compare all candidate-memory pairs and keep only reliable win/loss results, discarding tied or order-inconsistent comparisons. Second, for each context, we aggregate these pairwise preferences into a directed graph over candidate memories, where an edge $y \to x$ means that x is preferred to y for the current task distribution. Pairwise preferences may be non-transitive, leading to cycles in the raw graph. We remove comparisons

involved in cycles and retain the resulting DAG. A longer path $y \rightsquigarrow x$ in this DAG indicates a larger task-relevance gap between x and y.

We then select the final DPO pair from the DAG. For a valid pair $\langle x, y \rangle$, we require that x is preferred to y, that the preferred memory does not have lower basic quality than the dispreferred memory, i.e., $r_{\text{qual}}(x) \geq r_{\text{qual}}(y)$, and that the two memories are not both low-quality. Among all valid pairs, we choose the one with the largest path distance as the final preference pair. If multiple pairs have the same distance, we choose the pair with the smaller memory-length difference. The selected pair is recorded as (q, m_k^w, m_k^l).

In our implementation, we sample $N = 8$ candidate memories for each context. After filtering, 29.17% of sampled contexts yield valid preference pairs, covering roughly 100 videos for Phase Two training.

F.4 Training Metric Implementation

Accuracy. Accuracy measures whether Phase Two preserves the fundamental fidelity of memory generation. It is implemented using Gemini-2.5-Flash together with a rule-based length check. Gemini-2.5-Flash judges whether the generated memory is faithful to the visual content and subtitles of the current video clip, while the length check verifies compliance with the predefined memory token budget L_{mem}. A sample is counted as correct only if it passes both checks. The corresponding prompt is given in Table 16.

Non-redundancy rate. Non-redundancy measures whether Phase Two preserves the ability to generate incremental memory rather than repeating historical content. It is implemented using GPT-4o, which compares the generated memory against the historical memories and determines whether it introduces new information without repeating previously stored content. The evaluator also checks whether the memory remains textually well-formed and consistent with the expected style. A sample is counted as non-redundant if GPT-4o returns a positive judgment. The corresponding prompt is given in Table 17.

Relevance win/tie/loss ratio. Relevance measures whether Phase Two improves task-focused memorization relative to the Phase One reference policy. For each validation trajectory and each environment task, we compare the memory generated by the current policy against the memory produced by the reference policy π_0. We use GPT-4o with the same task-relevance prompt as in the task reward model: the evaluator receives the recent environment questions and the two candidate memories, and judges whether the current policy's memory is more task-relevant, equally task-relevant, or less task-relevant than the reference memory. We report the corresponding Win Ratio, Tie Ratio, and Loss Ratio. The corresponding prompt is given in Table 19.

F.5 Hyperparameters

Table 8 lists the hyperparameters used in DPO training.

Table 8: The hyperparameters used in DPO training.

Parameter Name	Value
Batch Size	64
Mini Batch Size	64
Number of 80GB GPUs	32
Learning Rate	1e-3

G Additional Results

G.1 Per-task Results

We provide per-task results on VideoMME, EgoLife, and EgoTempo in Tables 9, 10, and 11, respectively.

Table 9: Results for each task in VideoMME.

Method	Counting Problem			Information Synopsis			Object Recognition			Action Reasoning		
	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)
EgoGPT	22.3	40.9	54.4	75.6	90.0	84.0	44.7	55.0	81.2	37.1	50.5	73.6
HippoMM	25.5	55.9	45.5	75.6	91.9	82.3	53.0	65.7	80.7	38.1	53.3	71.4
M3-Agent	30.5	65.0	46.9	90.6	97.5	92.9	65.3	77.7	84.1	61.0	74.3	82.1
Gemini-1.5-Pro	26.8	49.5	54.1	88.1	96.9	91.0	54.7	61.7	88.6	71.4	84.8	84.3
Gemini-2.5-Pro	34.5	60.5	57.1	91.9	98.1	93.6	65.7	74.3	88.3	73.3	83.8	87.5
GPT-5.2	39.5	68.2	58.0	91.2	97.5	93.6	68.0	80.7	84.3	74.3	84.8	87.6
Qwen3-VL-30B-A3B	34.5	60.0	57.6	90.0	98.1	91.7	62.0	72.7	85.3	61.0	76.2	80.0
TaskMem	42.3	64.1	66.0	93.1	98.8	94.3	69.7	78.7	88.6	69.5	82.9	83.9
	Object Reasoning			**Temporal Perception**			**Attribute Perception**			**Temporal Reasoning**		
	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)
EgoGPT	48.6	65.0	74.8	59.2	77.6	76.3	43.6	54.9	79.4	45.3	69.8	65.0
HippoMM	52.8	71.0	74.3	51.0	73.5	69.4	50.8	62.6	81.1	46.5	79.1	58.8
M3-Agent	75.2	87.4	86.1	75.5	87.8	86.0	60.0	68.7	87.3	58.1	87.2	66.7
Gemini-1.5-Pro	66.4	78.0	85.0	61.2	75.5	81.1	35.9	39.5	90.9	68.6	84.9	80.8
Gemini-2.5-Pro	72.4	85.0	85.2	77.6	87.8	88.4	46.2	53.3	86.5	69.8	86.0	81.1
GPT-5.2	72.4	86.9	83.3	79.6	93.9	84.8	62.1	71.3	87.1	67.4	88.4	76.3
Qwen3-VL-30B-A3B	68.7	80.8	85.0	73.5	89.8	81.8	54.4	64.6	84.1	70.9	87.2	81.3
TaskMem	74.3	86.4	85.9	81.6	93.9	87.0	65.6	71.3	92.1	72.1	84.9	84.9
	Action Recognition			**OCR Problems**			**Spatial Perception**			**Spatial Reasoning**		
	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)
EgoGPT	34.4	47.2	72.9	48.0	62.4	76.9	43.1	58.8	73.3	66.7	77.8	85.7
HippoMM	36.8	51.6	71.3	64.8	76.8	84.4	43.1	64.7	66.7	71.1	86.7	82.1
M3-Agent	50.4	64.8	77.8	78.4	88.8	88.3	52.9	72.5	73.0	82.2	88.9	92.5
Gemini-1.5-Pro	54.0	64.0	84.4	48.0	53.6	89.6	47.1	56.9	82.8	80.0	84.4	94.7
Gemini-2.5-Pro	63.6	74.8	85.0	57.6	64.0	90.0	52.9	64.7	81.8	86.7	95.6	90.7
GPT-5.2	63.6	75.2	84.6	72.0	81.6	88.2	60.8	68.6	88.6	95.6	100.0	95.6
Qwen3-VL-30B-A3B	54.0	68.0	79.4	69.6	76.8	90.6	54.9	62.7	87.5	86.7	93.3	92.9
TaskMem	56.8	70.0	81.1	75.2	86.4	87.0	64.7	72.5	89.2	91.1	95.6	95.3

Table 10: Results for each task in EgoLife.

	RelationMap			EventRecall			EntityLog			HabitInsight			TaskMaster		
	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)
EgoGPT	9.6	20.8	46.2	26.2	37.3	70.2	10.4	13.6	76.5	14.8	27.9	52.9	46.0	54.0	85.3
HippoMM	16.8	34.4	48.8	36.5	50.8	71.9	29.6	37.6	78.7	27.9	41.0	68.0	49.2	60.3	81.6
M3-Agent	2.4	12.8	18.8	34.1	46.8	72.9	19.2	24.8	77.4	19.7	26.2	75.0	42.9	50.8	84.4
Gemini-1.5-Pro	23.2	40.0	58.0	46.0	59.5	77.3	28.0	36.8	76.1	52.5	59.0	88.9	68.3	81.0	84.3
Gemini-2.5-Pro	36.8	53.6	68.7	50.8	65.1	78.0	33.6	43.2	77.8	44.3	52.5	84.4	63.5	76.2	83.3
GPT-5.2	13.6	27.2	50.0	42.1	60.3	69.7	30.4	40.0	76.0	42.6	52.5	81.2	63.5	77.8	81.6
Qwen3-VL-30B-A3B	30.4	49.6	61.3	42.9	57.1	75.0	24.0	36.0	66.7	54.1	60.7	89.2	58.7	73.0	80.4
TaskMem	37.6	54.4	69.1	50.0	61.1	81.8	28.0	36.8	76.1	60.7	63.9	94.9	71.4	82.5	86.5

Table 11: Results for each task in EgoTempo.

	Action-Specific Object			Action Sequence			Locating Object			Spatial Relationship			Object Sequence		
	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)
EgoGPT	13.2	26.0	50.8	22.9	54.2	42.3	10.4	24.0	43.3	6.0	10.0	60.0	23.2	60.0	38.7
HippoMM	10.8	22.0	49.1	16.7	43.8	38.1	16.4	22.0	74.5	18.0	20.0	90.0	17.6	52.0	33.8
M3-Agent	14.4	30.0	48.0	17.1	52.1	32.8	9.6	22.0	43.6	10.0	12.0	83.3	24.4	64.0	38.1
Gemini-1.5-Pro	27.2	38.0	71.6	37.1	60.4	61.4	12.4	14.0	88.6	0.4	2.0	20.0	32.4	66.0	49.1
Gemini-2.5-Pro	30.4	44.0	69.1	47.5	75.0	63.3	19.2	24.0	80.0	6.4	10.0	64.0	36.4	68.0	53.5
GPT-5.2	26.4	40.0	66.0	44.6	66.7	66.9	27.2	38.0	71.6	22.0	30.0	73.3	39.2	78.0	50.3
Qwen3-VL-30B-A3B	29.2	44.0	66.4	37.9	62.5	60.7	11.6	20.0	58.0	6.4	10.0	64.0	31.6	70.0	45.1
TaskMem	30.0	42.0	71.4	53.1	74.3	71.5	20.4	30.0	68.0	8.0	10.0	80.0	37.2	68.0	54.7
	Object-Specific Action			**Temporal Event Ordering**			**Future Action Prediction**			**Counting Actions**			**Counting Objects**		
	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)	Acc.(↑)	Cov.	Prec.(↑)
EgoGPT	17.2	26.0	66.2	8.2	20.4	40.0	22.4	60.0	37.3	12.2	22.4	54.5	14.8	32.0	46.2
HippoMM	14.0	26.0	53.8	7.3	8.2	90.0	29.2	62.0	47.1	8.6	18.4	46.7	18.8	34.0	55.3
M3-Agent	12.0	20.0	60.0	15.5	36.7	42.2	28.0	66.0	42.4	8.6	18.4	46.7	20.8	42.0	49.5
Gemini-1.5-Pro	21.2	30.0	70.7	19.2	42.9	44.8	25.2	50.0	50.4	15.9	30.6	52.0	6.4	10.0	64.0
Gemini-2.5-Pro	27.2	38.0	71.6	29.8	53.1	56.2	26.8	56.0	47.9	15.5	30.6	50.7	19.6	26.0	75.4
GPT-5.2	38.0	48.0	79.2	33.1	57.1	57.9	40.4	76.0	53.2	21.6	36.7	58.9	28.8	44.0	65.5
Qwen3-VL-30B-A3B	18.8	26.0	72.3	29.8	46.9	63.5	28.4	58.0	49.0	16.7	32.7	51.2	12.8	20.0	64.0
TaskMem	24.8	34.0	72.9	29.4	51.0	57.6	36.8	66.0	55.8	22.9	32.7	70.0	21.2	38.0	55.8

Table 12: VideoMME results under different answer generators. We keep the generated memories fixed and only replace the model used to answer questions from memory.

Answer Generator	Model	Acc.(↑)	Cov.	Prec.(↑)
GPT-4o	Qwen3-VL-30B-A3B	61.6	74.7	82.5
	TaskMem	67.9	79.3	85.6
Gemini-2.5-Pro	Qwen3-VL-30B-A3B	68.9	81.7	84.4
	TaskMem	74.5	85.8	86.9

G.2 Robustness to the Choice of Answer Generator

GPT-4o is used in two parts of our pipeline: constructing task-relevance preference data for Phase Two and answering questions from memory during downstream evaluation. This may couple the learned preference signal with the evaluation model. To examine this effect, we conduct an additional robustness check on VideoMME while keeping the generated memories fixed and replacing only the model used for memory-based question answering. This setting preserves the Phase-Two memorization policy but tests whether the learned memories remain useful to an independent answer generator. We compare GPT-4o with Gemini-2.5-Pro under the same memory-based QA protocol.

As shown in Table 12, changing the answer generator affects the absolute scores, but the relative improvement of TaskMem over Qwen3-VL-30B-A3B remains consistent. TaskMem improves accuracy by 6.3 points with GPT-4o and by 5.6 points with Gemini-2.5-Pro. Coverage and precision also improve under both generators, indicating that the gains are not tied to a specific QA model.

The higher absolute scores under Gemini-2.5-Pro mainly come from higher coverage: given the same memories, Gemini-2.5-Pro more often identifies sufficient evidence and attempts an answer. Its precision also remains comparable, indicating that the higher coverage is not simply accompanied by indiscriminate answering. Since the generated memories are fixed in this comparison, these differences reflect the behavior of the answer generator rather than changes in memory quality. Despite these generator-dependent differences, TaskMem consistently improves over the base memorization policy under both answer generators, suggesting that its relative advantage is robust to the choice of answer generator.

G.3 Case Studies of Richness Reward

Table 13: Comparison of memory generations between `TaskMem` and `TaskMem` without richness reward (i.e., `w/o richness reward`). Yellow highlights indicate additional details provided by our model. For privacy protection, human faces in the illustrative video frames are masked only for visualization in this paper.

Case 1

`Video` (Illustrated as 7 frames)



`Episodic memory (TaskMem)`

`<face_1>` walks through a sunlit farm with rows of crops covered in plastic and netting, wearing a white and black long-sleeve shirt, black pants, and a cap. Gesturing with his hands as he speaks, `<face_1>` says, “Five years ago, I met a guy named Andrew at a farm-to-table dinner, and today we are at his farm, which he manages all by himself, while working 35 hours a week.” The background includes greenery, a wooden shed, and greenhouses under a clear blue sky.

`Episodic memory (w/o richness reward)`

(Continued on next page)

<face_1> walks through a sunlit farm with rows of crops, gesturing while speaking: “Five years ago, I met a guy named Andrew at a farm-to-table dinner, and today we are at his farm, which he manages all by himself, while working 35 hours a week.”

Case 2

Video (Illustrated as 7 frames)



Episodic memory (TaskMem)

<face_1> stands in a store aisle wearing a cap and white T-shirt, reaching toward shelves with sandwich makers as “COMING UP...” text overlays and price commentary play. The scene shifts to outside a Kmart building where <face_1> and <face_2> (in a white sleeveless dress and sunglasses) stand on a crosswalk; <face_1> gestures while saying, “Good morning and welcome back to the daily vlog. And today we’re going...”

Episodic memory (w/o richness reward)

<face_1> stands outside a Kmart under a bright blue sky, wearing a cap and white T-shirt, while <face_2> stands nearby. <face_1> greets viewers, saying, “Good morning and welcome back to the daily vlog. And today we’re going...”

Case 3

Video (Illustrated as 7 frames)



Episodic memory (TaskMem)

<face_1> stands beside a green train at night, wearing a blue jacket and gray beanie, with snow-covered ground and other people walking in the background. The text states, “Today I’m traveling by train for 80 hours from London two and a half thousand miles north to Lapland in the Arctic Circle.” An aerial view shows a train moving through a snowy landscape with “76 HOURS REMAINING” displayed. The scene shifts to Big Ben behind a black iron fence, then a map highlighting Lapland with <face_1>’s image. A snowy forest appears, followed by a person in a train looking out the window. Finally, the Swedish flag waves as the text reads, “Join me crossing eight countries over the next four days.”

Episodic memory (w/o richness reward)

<face_1> travels by train through snowy landscapes, passing landmarks like Big Ben and a map highlighting Lapland, while stating, “Today I’m traveling by train for 80 hours from London two and a half thousand miles north to Lapland in the Arctic Circle. Join me crossing eight countries over the next four days.”

H Prompt Templates for Training and Evaluation

The prompt for generating episodic memory.

You are given the following content:
- A video with corresponding faces (presented via bounding boxes) and subtitles.
- [Description of the preceding part] of the video. This field can be empty.

(Continued on next page)

Using the provided face IDs, write a detailed and cohesive description of the given video. The description should capture the complete set of observable and inferable events in the video. Your output should incorporate the following categories (but is not limited to these):

1. Characters' Appearance: Describe the characters' appearance, such as their clothing, facial features, or any distinguishing characteristics.
2. Characters' Actions & Movements: Describe specific gestures, movements, or interactions performed by the characters.
3. Characters' Spoken Dialogue: Transcribe or summarize what is spoken by the characters.
4. Characters' Contextual Behavior: Describe the characters' roles in the scene or their interaction with other characters, focusing on their behavior, emotional state, or relationships.

Strict Requirements:
- If a character has an associated face ID in the video, refer to them ONLY using that face ID.
- If characters DO NOT have associated face IDs in the whole video, it's ok not to describe them.
- A character may have multiple face IDs, and the ID currently displayed on the screen should be used for description.
- Ensure the continuity and uniformity of content between adjacent descriptions.
- Directly describe the video content, DO NOT start with 'The video ...'.
- If the video has an incomplete ending plot, the last line is truncated or asr is not a complete sentence, DO NOT describe it.
- The final output must be a dictionary, with the key being "description".

Output format:
```json
{
    "description": "<face_1> is standing outside under a blue sky with clouds. <face_1> gets out of the car and says: 'Hello everyone, welcome to my channel'."
}
```

[Description of the preceding part]:
{preceding_descriptions}

- Generate subsequent descriptions not covered in [Description of the preceding part], maintain coherence with it, and avoid any repetition of similar information.
- If [Description of the preceding part] is empty, describe the video from scratch.
- Generate the description briefly in one or two sentences.
Please output the description.

Table 14: The prompt for generating episodic memory.

The prompt for generating episodic memory with task prompts.

You are given the following content:
- A video with corresponding faces, presented via bounding boxes, and subtitles.
- [Description of the preceding part] of the video. This field can be empty.

Using the provided face IDs, write a detailed and cohesive description of the given video. The description should capture the complete set of observable and inferable events in the video. Your output should incorporate the following categories, but is not limited to these:

1. Characters' Appearance: Describe the characters' appearance, such as their clothing, facial features, or any distinguishing characteristics.
2. Characters' Actions & Movements: Describe specific gestures, movements, or interactions performed by the characters.

(Continued on next page)

3. Characters' Spoken Dialogue: Transcribe or summarize what is spoken by the characters.
4. Characters' Contextual Behavior: Describe the characters' roles in the scene or their interaction with other characters, focusing on their behavior, emotional state, or relationships.

Strict Requirements:
- If a character has an associated face ID in the video, refer to them ONLY using that face ID.
- If characters DO NOT have associated face IDs in the whole video, it's ok not to describe them.
- A character may have multiple face IDs, and the ID currently displayed on the screen should be used for description.
- Ensure the continuity and uniformity of content between adjacent descriptions.
- Directly describe the video content, DO NOT start with 'The video ...'.
- If the video has an incomplete ending plot, the last line is truncated, or ASR is not a complete sentence, DO NOT describe it.
- The final output must be a dictionary, with the key being "description".

- Your generated memory will be used to answer questions of the same type as those below. Please describe as clearly as possible the information in the video that helps answer such questions.
    - {example_question_1}
    - {example_question_2}
    - {example_question_3}
    - {...}

Output format:
```json
{
    "description": "<face_1> is standing outside under a blue sky with clouds. <face_1> gets out of the car and says: \"Hello everyone, welcome to my channel\"."
}
```

[Description of the preceding part]:
{preceding_descriptions}

- Generate subsequent descriptions not covered in [Description of the preceding part], maintain coherence with it, and avoid any repetition of similar information.
- If [Description of the preceding part] is empty, describe the video from scratch.
- Generate the description briefly in one or two sentences.
Please output the description.

Table 15: The prompt for generating episodic memory with task prompts.

The prompt for correctness of episodic memory.

You are provided with a video, a description of its preceding segment, and a generated candidate [Description] for the remaining portion.
Your task is to evaluate:
1. Whether the candidate description is factually accurate based only on visual content and subtitles (ignore audio).
2. Whether it connects coherently and naturally with the preceding description, without using transition words such as "continue".
For any spoken content, verify it solely against the displayed subtitles and disregard audio information.
Assign exactly one label:
1: Correct — The description that meets all of the above criteria.
0: Incorrect — Any description that fails to meet the above criteria.

(Continued on next page)

Output Requirements: Return the result in the following valid JSON format only. Do not generate anything else.

```
{
    "correctness_rationale": "Short explanation for marking this description as 1 or 0",
    "correctness": 1 or 0
}
```

The description of the preceding segment:
{preceding_descriptions}

The [Description] to verify:
{descriptions}

Table 16: The prompt for correctness of episodic memory.

The prompt for redundancy of episodic memory.

You are given the [Context] and a candidate description that are describing new events.

Your task is to evaluate whether the candidate description satisfies the following conditions.

Return label=0 if any condition is satisfied, else 1:
(1) The description repeats any atomic fact already present in the [Context].
(2) It includes any mention of bounding boxes, coordinates, or detection boxes (e.g., "bounding box", "bbox", "x1,y1,x2,y2", "rectangle box around").
(3) It contains meta phrases like: "subtitles said", "the subtitles say", "subtitle reads", "subtitle says", or "according to the subtitles".
(4) The quoted speech contains transcript-style speaker labels like "<face_id> says '<face_id>: Good.'"inside quoted dialogue.
(5) It includes conclusion-based or context-setting statements such as "this video ends with..."or "based on previous videos".

Output Requirements: Return the result in the following valid JSON format only. Do not generate anything else.

```
{
    "label_rationale": "Short explanation for marking this description as 1 or 0",
    "label": 1 or 0
}
```

[Context]:
{preceding_descriptions}

candidate description to verify:
{descriptions}

Table 17: The prompt for redundancy of episodic memory.

The prompt for richness of episodic memory.

You are given a list of descriptions summarized from a video, each associated with a unique ID. Please rank these descriptions based on their usefulness, output their ID from high to low. Usefulness should be determined by the amount of non-redundant, unique information contained in each item; items with more unique and less overlapping information should be ranked higher. Besides, descriptions that include dialogue directly in the narrative (e.g., <face_id> said, "xxx") should be ranked higher than descriptions that reference dialogue by referencing subtitles, captions, or other UI elements. The length of the output list must match the input list exactly.

Output format:
[RANK START]
[2, 1, 3, 6, 4, 5]
[RANK END]

Input Knowledge:
{descriptions}

Output the list of ID:

Table 18: The prompt for the richness of episodic memory.

The prompt for task-relevance pairwise comparison of episodic memories.

You are given two [Description] and some example questions.

Based on the focus of the example questions, your task is to evaluate which description contains information that would be more useful for answering similar questions.

Output the ID of the more useful description. If both descriptions are equally useful (a tie), output -1.

- A set of example questions: {example_questions}
- Two [Description]:
{blocks_text}

Return exactly one JSON object:
```
{
    "more_useful_rationale": "Briefly introduce the reasons for making this judgment",
    "more_useful": "ID of the more useful description or -1"
}
```

Table 19: Prompt for task-relevance comparison of episodic memories.

Prompt for VideoMME QA test.

Based on the following video description, select one option as the answer to the question. Give your reasoning for your answer. Output the option letter A, B, C or D. If you cannot find the answer, output E.

Video Description:
{memory_text}

Question:
{question_with_options}

(Continued on next page)

Reasoning:
[Your reasoning here]

Answer: [A|B|C|D|E]

Prompt for EgoLife QA test.

Based on the following video description, select one option as the answer to the question. The question is asked at the CURRENT time, but the relevant evidence is usually located in the TARGET clips. Only output the option letter A, B, C, D or E. If you cannot find the answer from the description, use E.

Description:
{descriptions}

Question:
{question}

Options:
{options}

Respond ONLY in strict JSON (no markdown, no code fences, no extra text).
The JSON schema is:
{
"cot": "Reasoning for the selected answer in English or Chinese",
"answer": "A|B|C|D|E"
}

Prompt for EgoTempo QA test.

These are descriptions of a video that I want to upload, please answer the question. You need to answer the question in any case and not demand additional context information. Note: All actions mentioned refer to the person recording the video.

Video Description:
{descriptions}

Question:
{question}

If the provided description is insufficient to answer the question, output 'Insufficient Information'.
Answer:

Table 20: The QA test prompts for VideoMME, EgoLife, and EgoTempo.

I Impact Statements

This work is foundational research on task-focused memorization for multimodal agents and is evaluated only in benchmark settings. The face and voice information used in this paper is employed solely for benchmark evaluation and representation alignment, such as assigning face IDs and speaker IDs within benchmark videos. We do not use these signals to identify real-world individuals, infer sensitive personal attributes, or support deployment in surveillance or user-profiling scenarios.

Our experiments are conducted on existing public benchmarks, including VideoMME, EgoLife, and EgoTempo. The face and voice processing components are used only as auxiliary tools for constructing benchmark inputs and evaluating long-term memory quality. For privacy protection, human faces in illustrative figures are masked when shown in the paper, and we do not release any additional personally identifying information.

Since this work focuses on a general memorization framework rather than a deployed application, we do not identify a direct path to specific negative societal impacts. If similar memorization mechanisms are adapted to real-world personal videos or interactive agents, appropriate consent, data minimization, and user control over stored memories should be considered.

J CASE STUDY

J.1 QUALITATIVE COMPARISON OF MEMORY EVOLUTION FROM PHASE ONE TO PHASE TWO

We present several case studies to compare how memory evolves from Phase One to Phase Two. Table 21 reports additional cases on different tasks, which together qualitatively demonstrate the effectiveness of our method.

Table 21: Qualitative comparison of episodic memory generations before and after the Phase Two training. Yellow highlights indicate additional fine-grained details generated after training.

Case 2 (Object Reasoning)

Representative task questions

"What is the most likely role of the blonde woman in the video, clad in a blue T-shirt and black shorts?"; "What is the role of the woman in the video with short hair, wearing a black top, and donning a white scarf?"; "In which country is the food featured in the video recognized worldwide?"; "Which player was the winner of the game?"; "Which of the following elements is not present in the "Starry Sky"?"

Video (Illustrated as sampled frames)



Episodic memory (Phase One)

<face_1> holds the sandwich, teasing, "You're gonna eat the whole thing before I even get to try it." They grin, adding, "Oh, well, we're gonna have to fight to the death for it." The scene shifts to a person in a red dress sitting at the same table, taking a large bite of the hoagie with evident enjoyment, their hands gripping the sandwich as they savor the flavors.

Episodic memory (Phase Two)

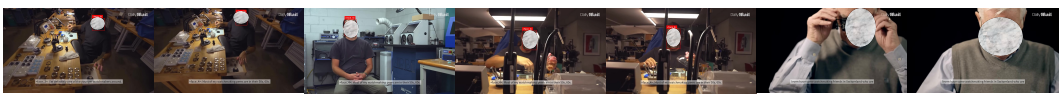
<face_1> remains seated at the table, holding the hoagie and stating, "You're gonna eat the whole thing before I even get to try it," with a playful yet teasing tone. Another voice responds, "Oh, well, we're gonna have to fight to the death for it," adding a lighthearted competitive vibe to the interaction. The scene then shifts to a woman with long red hair, wearing a red dress and a white wristwatch, seated at a similar checkered-tablecloth table. She takes a large bite of the hoagie, her expression focused on savoring the flavor, with the sandwich's layers of smoky brisket and jalapeno sausage visible as she eats. The background shows a warmly lit restaurant interior, maintaining the casual, communal dining atmosphere.

Case 3 (Attribute Perception)

Representative task questions

"From which country does the first-place athlete in the video originate?"; "What color are the foxes in the video?"; "Which genre best describes the performance based on the activity and costumes?"; "What is the shape of the paper shown in the video?"; "What can be inferred about the tone of the play?"

Video (Illustrated as sampled frames)



Episodic memory (Phase One)

(Continued on next page)

<face_4> sits in the workshop, stating, “Most of my watchmaking peers are in their 50s, 60s.” The scene then cuts to an older man adjusting a magnifying glass against a black background, saying, “I even have some watchmaking friends in Switzerland who are”.

Episodic memory (Phase Two)

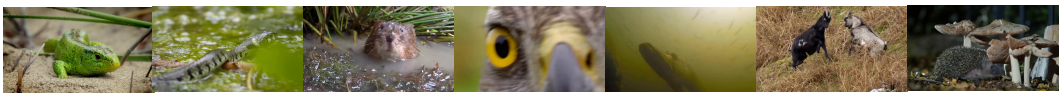
<face_4> sits in a workshop surrounded by machinery, wearing a dark shirt and khaki pants, with hands clasped as they speak: “Most of my watchmaking peers are in their 50s, 60s.” The scene transitions to an older man with white hair, dressed in a light blue shirt and gray vest, adjusting a magnifying glass while stating, “I even have some watchmaking friends in Switzerland who are”. The older man’s focused expression and deliberate hand movements emphasize the precision of watchmaking craftsmanship.

Case 4 (Counting Problem)

Representative task questions

“How many red flags appear in the video?”; “How many outfits did the male protagonist change in total in the video?”; “What is the total number of bird species that are visible in the video?”; “How many tricks are performed in this video?”; “How many people join in a toast at the end of the video?”

Video (Illustrated as sampled frames)



Episodic memory (Phase One)

A vibrant green lizard with a yellow head rests on sandy ground, its patterned scales blending with the surroundings. A snake glides through greenish water, partially submerged among aquatic plants. A beaver emerges from murky water, navigating reeds and vegetation with wet fur glistening. The intense yellow eye of an owl fills the frame, feathers ruffled. A fish swims in murky water, its form barely visible. Two goats clash horns in a field of dry, golden grass, their movements vigorous. Finally, a young otter lounges on seaweed, damp fur resting near the shoreline.

Episodic memory (Phase Two)

A vibrant green lizard with intricate patterns rests on sandy ground, its head slightly raised as it observes its surroundings amid thin green stems and dry grass. The scene shifts to a snake with a patterned body gliding through murky water, its head above the surface among aquatic plants. A water vole with brown fur and prominent whiskers emerges from murky water, surrounded by aquatic vegetation. A close-up reveals an owl’s striking yellow eye with detailed feather patterns, capturing its intense gaze. Underwater, a fish swims in yellowish water with other fish in the background. Two goats—one black, one white—clash horns in tall dry grass, their bodies tense during the struggle. A young otter with wet brown fur lies on seaweed-covered rocks, eyes wide and alert. Finally, a hedgehog with spiny brown fur sniffs around large mushrooms in a forest, surrounded by fallen leaves and dark foliage.

J.2 A Complete Episodic Memory Example on a Long Video

In this subsection, we present a complete episodic memory generated by TaskMem for a five-minute long video, shown in Table 22.

Table 22: A complete episodic memory example on a long video. Square-bracketed tokens (e.g. [face_1]) are used for entity tracking. The example shows that the generated memory is accurate, coherent, and task-relevant. Task-relevant memories are highlighted in blue. The video and question are from VideoMME, and the original video is available at: https://www.youtube.com/watch?v=uC9qU3X1JgM.

Question. How many persons does Superman fight versus?

Options. A. 6. B. 4. C. 5. D. 7.

(Continued on next page)

Ground-truth Answer. C. 5.

Model Prediction. C. 5.

Generated Memory.

00:00:00–00:00:10

A shirtless man with dark hair stands with his back to the camera, facing a group of superheroes at a memorial site featuring a reflecting pool, scattered flowers, and city buildings under twilight. The group includes [face_1] in a red suit, [face_2] in a dark armored suit, and [face_3] (Wonder Woman) in golden-blue armor with a tiara, sword, and shield. [face_3] turns to face the shirtless man and speaks: "He's back." The scene includes police cars and onlookers in the background, emphasizing a moment of tension or reunion.

00:00:10–00:00:20

The shirtless man identified as [face_4] stands with a serious expression, facing the group of superheroes at the memorial site. [face_2] (Aquaman) holds a trident, clad in his armored suit, while [face_3] (Wonder Woman) remains in her golden-blue armor with a sword and shield. [face_1] (The Flash) in his red suit and [face_5] (Cyborg) with a glowing chest stand alongside them. The reflecting pool, scattered flowers, and cityscape under twilight persist in the background, maintaining the scene's tense atmosphere as the group collectively observes [face_4]

00:00:20–00:00:30

[face_3] (Wonder Woman) in golden-blue armor with a tiara, sword, and shield turns to face [face_4], stating, "He's back." Her expression is intense. [face_4], shirtless with dark hair, stands with a stern look toward the group. The scene cuts to [face_2] (Aquaman) in his armored suit, long hair and beard framing a serious expression, followed by [face_1] (The Flash) in his red suit with a lightning bolt emblem, also solemn. [face_4] remains tense, then the group appears in a blue-lit environment where their muscular forms are briefly highlighted, suggesting a power-related moment, while the memorial's reflecting pool and cityscape persist in the background.

00:00:30–00:00:40

A close-up of [face_4]'s intense eyes fills the frame, emphasizing tension. The scene shifts to [face_5] (Cyborg) with a glowing chest, stating, "Scanning him." [face_1] (The Flash) in his red suit reacts with, "What?" as [face_5] continues, "Arthur, you need to relax. Your adrenaline's spiking," while the cityscape and memorial site remain in the background, maintaining the scene's urgent atmosphere.

00:00:40–00:00:50

The shirtless man [face_4] stands on the memorial steps, facing Wonder Woman and The Flash. He asks, "Or show our bellies?" with a tense expression. Cyborg [face_5] intervenes, saying "Victor, stop" while raising a hand in a restraining gesture. Wonder Woman and The Flash remain alert, their postures tense as they respond to the interaction. The memorial's reflecting pool, cityscape, and scattered flowers persist in the background, maintaining the scene's urgent tone.

00:00:50–00:01:00

[face_5] (Cyborg) with a glowing chest and metallic armor speaks urgently, saying, "Victor, it's my armor's defense system. It's stronger since the interface," while raising a defensive hand. [face_3] (Wonder Woman) intervenes, shouting, "Victor, no!" as [face_4] (shirtless man on the memorial steps) adds, "You're gonna cause an alarm," his expression tense. The Flash ([face_1]) and Aquaman ([face_2]) stand alert nearby, their postures rigid, while the cityscape and memorial pool remain in the background, heightening the scene's urgency.

00:01:00–00:01:10

The shirtless man [face_4] places his hand on his pants as Cyborg [face_5] fires a red energy beam toward him, causing an explosion on the memorial steps. Smoke billows as an aerial view reveals the chaos, with the group scattered and the memorial site in disarray. Wonder Woman [face_3] and The Flash [face_1] react with shock, while Aquaman [face_2] looks on in concern. The cityscape and reflecting pool remain visible, amplifying the scene's tension as the aftermath of the blast unfolds.

00:01:10–00:01:20

(Continued on next page)

Wonder Woman [face_3] shouts "Kal-El, no!" as Cyborg [face_5] fires an intense red energy beam, triggering a massive explosion that illuminates the memorial site. The Flash [face_1] and Aquaman [face_2] brace themselves against the blast's force, while the shirtless man [face_4] stands amid rising smoke and debris. Flames and light fill the frame as the explosion's shockwave disrupts the surrounding area, with the city skyline and reflecting pool now partially obscured by the chaos.

00:01:20–00:01:30

The Flash [face_1] rises from the ground, expression bewildered, as he mutters, "He's confused." Wonder Woman [face_3] exclaims, "He doesn't know who he is," while scanning the smoldering memorial site. The Flash adds, "That cemetery," his voice tense. Meanwhile, the shirtless man [face_4] hoists a massive stone from the steps, muscles straining, as Aquaman [face_2] and Wonder Woman ready their weapons, bracing for further conflict amid the lingering smoke and scattered debris.

00:01:30–00:01:40

Wonder Woman [face_3] sprints toward the conflict, her golden armor gleaming as she calls out, "Arthur, we need to restrain him." Aquaman [face_2] follows, trident raised, while the shirtless man [face_4] struggles against their combined effort. The Flash [face_1] dashes in, red suit flashing, as Wonder Woman and Aquaman grapple with the muscular figure. Smoke from the earlier explosion lingers in the air, mixing with the city's glow as the trio maneuvers around the memorial steps, their movements urgent and coordinated in the chaos.

00:01:40–00:01:50

Wonder Woman [face_3] lies on the memorial steps, her shield scattered nearby, as she regains her footing. She rises swiftly, lasso in hand, as the shirtless man [face_4] unleashes a golden energy beam toward her. The Flash [face_1] zips in to intercept, creating a streak of red, while Aquaman [face_2] charges with his trident. The city's glow illuminates the scene as smoke from the earlier blast lingers. Wonder Woman shouts, "You got it!" as she dodges the beam, her golden armor reflecting the light. The shirtless man [face_4] grins, his muscles taut, as he continues the assault, the energy beam slicing through the air. Aquaman and The Flash work to flank him, but the muscular figure moves with impossible speed, his bare chest glistening under the city lights.

00:01:50–00:02:00

[face_4] maintains his stance, emitting a powerful golden energy beam from both hands as he targets [face_3]. [face_3] grips the glowing lasso, twisting her body to deflect the beam while shouting, "Kal-El?" The beam cuts through the air, but she uses the lasso to redirect its force. [face_1] rushes in as a red blur, attempting to intercept [face_4], while [face_2] charges forward with his trident, aiming to flank him. Smoke from the earlier explosion lingers, mixing with the city's glow as debris scatters across the memorial steps. [face_4] grins intensely, his muscles straining under the effort, undeterred by the coordinated attacks from Wonder Woman and Aquaman.

00:02:00–00:02:10

[face_3] grips the glowing lasso, her expression intense as she declares, "Kal-El, the last son of Krypton." She pulls the lasso taut, redirecting the golden energy beam while urging, "Remember who you are." [face_4] stands firm, muscles rippling under the city lights, his gaze locked on [face_3] as he responds with a strained voice. The memorial site chaos continues, debris scattering as [face_1] (The Flash) zips past in a red blur, attempting to aid [face_3]. Smoke from previous explosions mingles with the urban glow, highlighting the tension between the characters. [face_3] presses forward, lasso glowing brighter, as she demands, "Tell me who you"—her voice cutting through the din of battle.

00:02:10–00:02:20

Wonder Woman [face_3] and Aquaman [face_2] maintain a firm grip on [face_4], their combined strength overpowering his resistance as he struggles against their restraint. The Flash [face_1] zips in, adding his speed to secure [face_4], whose golden energy beam flickers and fades. Wonder Woman, lasso still glowing, repeats, "Remember who you are," her voice cutting through the chaos. Smoke from earlier explosions drifts over the memorial steps, where debris litters the ground. Aquaman's trident glints as he holds [face_4] steady, while Wonder Woman's expression shifts from urgency to determination as they finally subdue him.

00:02:20–00:02:30

(Continued on next page)

Wonder Woman [face_3] keeps the glowing lasso taut around [face_4], her expression intense as she leans close. [face_4] trembles, his golden energy fading, while Aquaman [face_2] holds him firm with a trident. The Flash [face_1] stands nearby, red suit crackling with electricity. Wonder Woman murmurs, "You are Kal-El, the last son of Krypton," as [face_4]'s eyes flicker with recognition, his muscular frame slackening under their restraint. Debris and smoke linger on the memorial steps, city lights reflecting off the tension between them.

00:02:30–00:02:40

[face_1] (The Flash) bursts into the scene, red suit crackling with electric energy as he zips toward [face_4]. Lightning trails behind him, casting sharp shadows on the memorial steps. He unleashes a pulse of blue energy, disrupting [face_4]'s fading golden aura. [face_4] wobbles, muscles straining as [face_3] (Wonder Woman) tightens the glowing lasso around his wrist and [face_2] (Aquaman) anchors him with a trident. Wonder Woman's voice rings out: "You're not alone anymore." The Flash steps back, breathless, as [face_4]'s eyes dim, his body collapsing under their combined hold. Debris scatters across the wet pavement, city lights reflecting off the tension as the trio secures [face_4], ensuring he cannot resist further.

00:02:40–00:02:50

The trio stands amidst the debris of the memorial site, city lights casting a blue glow over the scene. [face_3] (Wonder Woman) releases the lasso, her hand resting on [face_4]'s shoulder as he breathes heavily, his golden aura gone. [face_2] (Aquaman) lowers his trident, while [face_1] (The Flash) gestures toward the skyline, saying, "We need to move. The city's in chaos." They begin walking away, the city's glow reflecting off their determined expressions as smoke lingers in the background.

00:02:50–00:03:00

[face_4] and [face_1] engage in a fierce battle, with [face_4] delivering heavy punches while [face_1] evades using lightning-fast movements, electric arcs flaring around his red suit. [face_1] counters with a blast of blue energy, forcing [face_4] to stagger as debris scatters across the memorial steps. Wonder Woman [face_3] and Aquaman [face_2] watch from the sidelines, weapons poised, as [face_4] attempts a spinning kick that [face_1] sidesteps, retaliating with a rapid strike that sends [face_4] crashing into a shattered stone pillar. The city skyline glows behind them, reflecting off the tension in their clash as lightning continues to crackle around [face_1]'s form.

00:03:00–00:03:10

[face_19] stands in a cityscape of towering skyscrapers, smoke rising from a burning vehicle and a police car in the background. Dressed in a dark, armored suit with a cape, [face_19] surveys the chaos, his cowl shadowing his face. Nearby, [face_4] appears shirtless, muscles taut and skin glistening with sweat, as city lights reflect off his form. [face_19] shifts slightly, assessing the scene amid scattered debris and flickering urban lights, while [face_4] maintains a tense posture, the aftermath of the earlier battle lingering in the air.

00:03:10–00:03:20

[face_19] (Batman) stands amidst smoldering wreckage and flickering city lights, his armored suit glistening under the evening sky. He faces [face_4] (Clark), who is shirtless, muscles taut, and sweat glistening on his skin. [face_4] meets [face_19]'s gaze and says, "I know you," his voice steady amid the chaos. Behind them, burning vehicles and scattered debris mark the battle's aftermath, while Wonder Woman approaches with her lasso coiled and shield raised. The city's glow reflects off the wet pavement, highlighting the tension between the two figures as they stand locked in a silent understanding.

00:03:20–00:03:30

[face_3] (Wonder Woman) stands near a reflective water feature, her golden tiara gleaming under the city's twilight glow. She declares, "We do this," as she grips her lasso and shield, her expression resolute. Facing [face_4] (shirtless, sweat glistening on his muscles), she launches a swift punch, prompting [face_4] to evade with an agile sidestep. The two clash in a flurry of movements—Wonder Woman's lasso whips through the air, while [face_4] counters with rapid strikes, debris scattering as their battle intensifies against the backdrop of burning vehicles and towering skyscrapers.

00:03:30–00:03:40

(Continued on next page)

[face_3] lies on the wet pavement, her golden tiara askew and shield resting beside her, the aftermath of the clash with [face_4] evident in her strained posture. [face_4] stands over her, chest heaving, sweat dripping down his muscular frame as he glances toward [face_19] (Batman), who approaches from the smoldering wreckage. [face_19]'s armored suit gleams under city lights, his cowl shadowing his face as he surveys the scene. The "METROPOLIS" police car is visible in the background, its lights pulsing amid smoke from burning vehicles. [face_4] maintains a tense stance, ready to react as [face_19] closes the distance, the city skyline reflecting off the rain-slicked ground.

00:03:40–00:03:50

[face_19] crouches beside the "METROPOLIS" police car, smoke rising from nearby wreckage. He speaks urgently, "Alfred, I need the big guns," as [face_4] steps forward. [face_4] grabs [face_19]'s cowl with both hands, pulling it taut before lifting [face_19] off the ground. The city skyline looms behind them, skyscrapers reflecting the chaos as [face_4] holds [face_19] aloft, muscles straining under the evening light.

00:03:50–00:04:00

[face_4] grips [face_19]'s cowl tightly, his voice steady as he says, "You knew this." [face_19] responds, "I had to," while struggling against the hold. [face_4] leans in, intensity in his eyes, declaring, "You won't let me live." [face_19] meets his gaze, replying, "The world needs you." As the tension crackles, [face_4] releases [face_19], who stumbles back, adjusting his cowl. The city skyline looms behind them, skyscrapers reflecting the chaos of burning vehicles and flickering lights, while the two stand locked in a moment of unresolved conflict.

00:04:00–00:04:10

[face_4] stands with a tense expression, his voice firm as he declares, "It doesn't need you." [face_19] remains in the grip of the moment, one hand still near his cowl as he processes the words, the city's chaos unfolding behind them with smoke rising from distant fires and skyscrapers towering into the twilight.

00:04:10–00:04:20

[face_4] grips [face_19]'s cowl firmly, leaning in with a challenging expression as he asks, "Do you bleed?" [face_19] remains tense, one hand still near his cowl while the mask shifts slightly to reveal a portion of his face, eyes narrowed in response. The city skyline, dotted with illuminated skyscrapers and lingering smoke from earlier clashes, frames the confrontation. [face_19] does not speak immediately, the weight of the question hanging in the air as the two figures stand locked in a silent battle of wills.

00:04:20–00:04:30

Inside a luxury car, [face_21] sits in the backseat wearing a brown leather jacket and glasses, the "MAYBACH" emblem visible on the door. The scene shifts to [face_4], shirtless with dark pants, standing against the city skyline as he engages in a tense confrontation. Meanwhile, [face_22]—a woman with long red hair in a dark coat over a blue shirt—runs across the grassy area toward a "METROPOLIS" police car marked "8202," with an officer holding a weapon behind her. Her expression is urgent as she approaches the vehicle, the backdrop of illuminated skyscrapers and lingering smoke from earlier conflicts underscoring the chaotic atmosphere.

00:04:30–00:04:40

[face_22] stands beside the "METROPOLIS" police car marked "8202," her posture rigid with urgency as she faces [face_4], who approaches from the grassy expanse. The city skyline, dotted with glowing skyscrapers and wisps of smoke from earlier chaos, looms behind them. [face_4], shirtless and clad in dark pants, moves with deliberate purpose, his expression tense as the unresolved tension from their earlier exchange hangs heavy in the air. The officer with the weapon remains in the background, adding to the scene's charged atmosphere.

00:04:40–00:04:50

[face_22] faces [face_4] with a pleading expression, her voice trembling as she says, "Please." She stands close to the "METROPOLIS" police car, her dark coat slightly disheveled from running. [face_4] remains rigid, his shirtless torso glistening under the city lights as he stares back, the tension between them palpable. The backdrop of towering skyscrapers, glowing windows, and lingering smoke from earlier chaos underscores the urgency of the moment, while the officer with the weapon stays positioned behind the police car, adding to the scene's intensity.

00:04:50–00:05:00

(Continued on next page)

[face_4] stands close to [face_22], his shirtless torso glistening under the city lights as he meets her pleading gaze. [face_22]'s hand brushes against his shoulder, tears tracing her cheeks as she whispers, "Please," her voice trembling against the backdrop of the "METROPOLIS" police car and towering skyscrapers shrouded in smoke.

J.3 Hard Case Analysis

We conduct an error analysis to investigate the hard cases for our approach. Specifically, we randomly sampled 50 incorrectly answered questions and manually inspected their corresponding generated memories. For each case, we examined whether the memory preserved sufficient evidence to answer the question, whether the preserved evidence was factually accurate, and whether the QA model still failed even when relevant evidence was available. Across the sampled cases, we identify three major failure modes: (1) missing fine-grained visual evidence, (2) inaccurate or misleading visual descriptions, and (3) insufficient relational and temporal integration. Representative examples are shown in Tables 23–25.

Overall, these findings suggest that future work should incorporate richer visual information into the memory. Key open questions include how to structure and organize visual memory, how to determine which visual details are worth retaining, and how to preserve spatial, relational, and temporal cues from cross-modal information.

Table 23: Hard case caused by missing fine-grained evidence in the generated memory.

Question: What is the number of the first lipstick she used?

Options: A. 959. B. 656. C. 858. D. 666.

Ground-truth Answer: B

Model Prediction: E

Generated Memory.

00:00:00–00:00:10
The video opens with text reading "Beauty Secrets" followed by "Adriana Lima's Two-Minute Party Makeup." [face_1] appears in a bathroom with tiled walls and a shower area, wearing a black-and-white robe with hair tied back. She waves both hands, says "Hi everyone. My name is Adriana. I love makeup, so this is what I do," then adjusts her hair and holds up a BB Cream product labeled "BB Cream" while speaking.
00:00:10–00:00:20
[face_1] holds up a primer bottle labeled "Master Prime" while stating "Primer," then mixes the product on her hand before applying it to her face. She explains, "You see, I mix it up like that. You just go for it. I always like the glow," as she spreads the primer across her skin with her fingers, maintaining a focused expression while demonstrating the makeup technique in the bathroom setting.
00:00:20–00:00:30
[face_1] continues speaking in the bathroom, stating, "I don't like when your skin has a matte finish," while using hand gestures to emphasize her point. She then picks up a white cup, takes a sip, and says, "English breakfast tea. Okay with honey, that's what I like," maintaining a conversational tone as she shares her personal preference for tea while remaining in the tiled bathroom setting.
00:00:30–00:00:40
[face_1] holds up a Maybelline "The Nudes" eyeshadow palette, saying, "The nudes, and I'm mixing these two colors." She uses a brush to blend two shades from the palette, then applies the mixture under her eye with precise hand movements. Maintaining a focused expression, she continues explaining the makeup technique while demonstrating the application in the tiled bathroom environment.
00:00:40–00:00:50

(Continued on next page)

[face_1] holds a highlighter palette, selecting a shade while stating, “This one here and then I go.” She applies the highlighter to her cheekbones and forehead with a brush, emphasizing a glowing finish. Next, she picks up a mascara wand, says “Mascara,” and carefully applies it to her upper lashes, ensuring even coverage. Throughout, she maintains a focused demeanor, continuing her makeup tutorial in the tiled bathroom setting.

00:00:50–00:01:00

After applying mascara, [face_1] remarks, “You see my eyes is so weird coming in,” while adjusting her gaze. She then shifts focus to her eyebrows, stating, “Oh my eyebrows, I’m just gonna leave it naturally and I can just brush with my fingers,” as she uses her fingers to groom them. Next, [face_1] picks up a contour palette, declares, “I’m gonna use the contour,” and prepares to apply the product, maintaining her instructional tone in the consistent bathroom setting with tiled walls and a shower visible in the background.

00:01:00–00:01:10

[face_1] transitions to applying blush, holding up a blush palette and stating, “a little bit of the blush.” She uses a brush to apply the product to her cheeks, focusing on achieving a natural flush. Throughout the process, she maintains her instructional tone, occasionally adjusting her position to ensure even application while remaining in the consistent bathroom setting with tiled walls and a shower visible in the background.

00:01:10–00:01:20

[face_1] blends blush on her cheeks with her hand, saying, “And if it’s a little pinkish, I kind of like it,” while ensuring an even application. She then transitions to lipsticks, stating, “I’m obsessed with lipsticks,” as she readies the next step in her makeup routine within the tiled bathroom environment. [**Missing:** The memory only states that she transitions to lipsticks, but it does not record the visible number or shade label of the first lipstick, which is the key evidence required by the question.]

00:01:20–00:01:30

[face_1] holds up a lipstick, saying “Let’s see,” before applying it to her lips with careful precision. She then picks up a lip liner, preparing to define her lip shape while maintaining her focused demeanor. Throughout, she remains in the tiled bathroom setting, continuing her makeup tutorial with an instructional tone as she experiments with different lipstick shades to achieve her desired look.

00:01:30–00:01:40

After applying lip liner, [face_1] adjusts her hair while stating, “This can be a cocktail look, guys,” and emphasizes, “Look, you just change the lipstick.” She continues refining her lip makeup with deliberate movements, maintaining focus on achieving the desired shade and shape. The bathroom setting remains consistent, with tiled walls and a shower visible in the background, as she concludes this segment of her makeup tutorial with a practical demonstration of lipstick customization.

00:01:40–00:01:50

After applying lipstick and exclaiming “Ta-da,” [face_1] smiles broadly, then says, “Love you all and see you soon” while pointing toward the camera. The scene concludes with a white screen displaying “Vogue.com” as [face_1] finishes her makeup tutorial in the consistent bathroom setting with tiled walls and a shower visible in the background.

Table 24: Hard case caused by incorrect or misleading information in the generated memory.

Question: On which shirt does the man show drawing imaginary lines?

Options: A. The shirt of white. B. The shirt of black. C. The shirt of blue. D. The shirt of yellow.

Ground-truth Answer: B

Model Prediction: C

Generated Memory:

00:00:00–00:00:10

(Continued on next page)

A white short-sleeve shirt with blue vertical stripes lies flat on a wooden surface. Text on the screen reads, "Today I'm going to show you how to fold a shirt in under two seconds." A pair of hands enters the frame, swiftly folding the shirt by aligning the sleeves and pressing the fabric. The folded shirt is placed back on the surface, neatly arranged with the collar visible. New text appears: "Start by taking a short sleeve shirt" as the hands adjust the folded garment.
00:00:10–00:00:20
A dark blue short-sleeve polo shirt with a small emblem on the chest is placed flat on the wooden surface. The hands adjust the shirt to lie smoothly, ensuring it is properly positioned. Text on the screen reads, "Start by taking a short sleeve shirt and lying it out on its back." The hands then gesture to indicate imaginary lines: one halfway between the top and bottom of the shirt, and another between the center line and the outside, as the voiceover explains the folding preparation steps.
00:00:20–00:00:30
The hands point to the intersection of the imaginary lines on the dark blue polo shirt, labeling the crosspoint as A, the top as B, and the bottom as C while the voiceover states, "We'll call the point where the lines cross A, the top point B, and the bottom point C." The hands then shift to pinch the shirt at point A with the left hand, following the instruction, "Start by pinching the shirt at point A with your left hand," as the preparation for the folding process continues.
00:00:30–00:00:40
The hands continue the folding process by lifting point B with the right hand while maintaining a pinch at point A with the left hand, then crossing the right hand over to grasp point C. After securing these points, the hands swiftly unfold the arms and use the wooden surface to fold the shirt backward, aligning the fabric neatly. The voiceover instructs, "Next, quickly unfold your arms, and finally use the table to fold the shirt back," as the shirt is transformed into a compact shape through precise hand movements.
00:00:40–00:00:50
The hands place a yellow short-sleeve shirt with a graphic design onto the wooden surface, smoothing it out before folding. Following the same technique, the hands pinch point A, lift point B, and cross to grasp point C, then unfold and use the table to fold the shirt neatly. Next, a light blue short-sleeve shirt is laid out, adjusted, and folded using the identical method. As the final shirt is folded, text appears on screen: "If you like this video, maybe you'd like to" while the hands complete the fold, leaving the shirt compact and neatly arranged on the wooden surface.
00:00:50–00:01:00
The video transitions to an end screen featuring a neatly folded blue dress shirt on the left. On the right, four video thumbnails are displayed with titles: "How To Make A Mini Bow And Arrow," "How To Make The Coca-Cola Truck," "How To Make A Light Bulb Vase," and "How To Make A Coca-Cola Can Rose." Text above the thumbnails reads "Click To View," while below the shirt, the text states, "If you like this video, maybe you'd like to take a look at some of my others by clicking on the links on the right-hand side." A yellow "Subscribe" button appears at the bottom left, with "DaveHax.com" beneath it. The message "Thanks for watching." is also visible, concluding the video.
00:01:00–00:01:10
The person, wearing a blue and yellow striped shirt, demonstrates folding a dark blue polo shirt on a wooden surface. They mark points A, B, and C on the shirt, pinch at A with the left hand, lift B with the right hand, cross the right hand to grasp C, then unfold and use the table to fold the shirt neatly. This process repeats with a yellow graphic short-sleeve shirt and a light blue short-sleeve shirt, each folded using the same technique. The video concludes with an end screen featuring a neatly folded blue dress shirt, four video thumbnails titled "How To Make A Mini Bow And Arrow," "How To Make The Coca-Cola Truck," "How To Make A Light Bulb Vase," and "How To Make A Coca-Cola Can Rose," along with text prompting viewers to click for more content, a "Subscribe" button, and "DaveHax.com" at the bottom.

Table 25: Hard case where the generated memory captures local visual cues but fails to integrate their relational and temporal structure.

Question: How does the girl perceive her future with regard to decision-making and control?

(Continued on next page)

Options: A. She doesn't care about her future. B. She wishes to make decision by herself. C. She feels she will be controlled by her mom. D. She has no hope for the future.
Ground-truth Answer: C
Model Prediction: B

Generated Memory:

00:00:00–00:00:10
[face_1] stands against a light blue backdrop, wearing a red shirt and holding a drawing featuring a cartoon figure and hearts. The text "When I try to hide" appears as [face_1] glances toward [face_2]. [face_2], in a green shirt with short dark hair, is partially visible on the left. The text "there you are" displays as [face_2]'s face becomes fully visible, showing a neutral expression. [face_1] looks down, shifts to a slight frown, then smiles while still holding the drawing. The scene transitions to [face_2] holding a purple "ALGEBRA" book with a parabola diagram against an orange background, with the text "you were so much more" appearing as [face_2] holds the book and [face_1] is visible from the back on the right.

00:00:10–00:00:20
The video transitions to a three-panel layout. The left panel displays a drawing of a purple-haired pony with hearts and clouds. The middle panel shows [face_1] in a dark dress with a white collar, holding papers and wearing a somber expression. The right panel depicts two figures embracing. The scene shifts to two panels: the left shows open books with lined pages, while the right panel features [face_1] in a light gray shirt, looking concerned. A close-up follows of a drawing with a purple-haired pony, a pencil, and the text "call me fighter". Finally, [face_1] stands against an orange background in a black shirt, pointing with both hands as the text "call me FIGHTER" appears in bold letters.

00:00:20–00:00:30
[face_1] stands beside a wall where the pony drawing is taped, hands raised in a presenting gesture as the text "call me LOVER" appears. The scene shifts to [face_1] in a black shirt against an orange background, making peace signs with both hands while the text "call me LOVER" is displayed in large letters. Next, [face_1] holds multiple books with a neutral expression, and the text "a drink OR" shows up. The text then changes to "I'll take you" as [face_1]'s expression softens into a smile. Finally, [face_1] appears with light blonde hair and a black shirt, standing before an orange backdrop with red puppet strings; one hand is raised as if controlling the strings, with the text "I'll take you" still visible.

00:00:30–00:00:40
The video transitions to a brown background where [face_1] stands beside a hand-drawn sign reading "Thanks to my patrons ssaparova_ soso_coaster" with hearts, and "Thanks for watching" below. [face_1] smiles warmly, pointing at the sign with one hand while the other hand rests by their side, concluding the video with a grateful message.

Analysis. The generated memory captures several local visual cues, but it fails to preserve the key relational and temporal information needed to answer the question. This suggests that the failure is not due to missing low-level perception, but rather insufficient integration of visual cues into task-relevant semantic memory.