DyhimemSkill: Dynamic Hierarchical Memory and Inference Skills for Long-Form Egocentric Video Question Answering

Anonymous ACL submission

Abstract

Long-form egocentric video question answering (VQA) is uniquely challenging for multimodal large language models (MLLMs) due to continuous viewpoint shifts, visual occlusions, and the temporal fragmentation of relevant events. To overcome the high costs of fine-tuning and the limitations of generic prompting, we present **DyhimemSkill**, a fully training-free framework designed to maximize the inherent reasoning capabilities of frozen MLLMs. Our approach decouples context preservation from query execution via two inference-time abstractions. First, Dynamic Hierarchical Memory tackles temporal fragmentation by semantically grouping local video chunks into a query-agnostic substep–step–goal architecture, seamlessly connecting non-contiguous observations to preserve long-range procedural context. Second, recognizing that different question types demand distinct cognitive strategies, we introduce Inference Skills. Inspired by Agent Skills, these reusable procedures explicitly guide the frozen MLLM on how to gather evidence, adapt visual focus, invoke auxiliary tools, and verify candidates. Evaluations across HD-EPIC, EgoSchema, and QAEgo4D demonstrate that DyhimemSkill consistently improves frozen MLLM backbones, validating our dynamic memory and inference skills as a robust, transferable framework for egocentric VQA.

1 Introduction

Long-form egocentric video understanding is a crucial step toward embodied agents that can reason about human activities from a first-person perspective. However, reasoning over these videos is inherently challenging due to continuous viewpoint shifts, severe hand occlusions, and the sparse, non-contiguous distribution of relevant events over time. Video question answering (VQA) effectively exposes these difficulties, requiring models to not only retrieve sparse evidence from long recordings but also perform complex spatiotemporal reasoning—such as connecting repeated object interactions, inferring multi-step procedures, or grounding spatial relations and gaze from partial views.

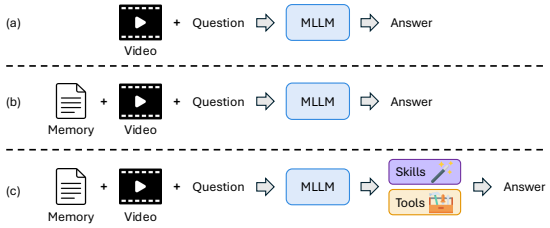

Figure 1: **Comparison of inference pipelines.** (a) Standard: video and question are directly fed to an MLLM. (b) Dyhimem: dynamic hierarchical memory provides temporally grounded procedural context. (c) DyhimemSkill: task-specific skills provide specialized inference protocols and optional tool-derived evidence.

To tackle these challenges, recent approaches have increasingly relied on computationally expensive fine-tuning using large-scale egocentric datasets (Ye et al., 2025; Luo et al., 2025; Pei et al., 2026). While this can adapt models to first-person views, it requires significant resources and fundamentally alters foundational models. To overcome these limitations, we propose a fully training-free framework to maximize the inherent reasoning capabilities of existing, frozen multimodal large language models (MLLMs). Recognizing that different question types demand distinct evidence formats, our approach effectively decouples the inference process: it utilizes a dynamic hierarchical memory to robustly preserve the procedural long-range context of the video, coupled with inference skills that guide the frozen MLLM to actively retrieve and apply that context for each specific query.

Specifically, our framework, DyhimemSkill, explicitly decouples the VQA pipeline into two independent stages: building a query-agnostic memory of the video, and applying a tailored reasoning strategy at inference time. Dynamic Hierarchi-

cal Memory builds a query-agnostic foundation by summarizing local video chunks into timestamped substeps and semantically grouping them into step-level memories and a video-level goal. This organizational structure directly addresses a critical challenge in egocentric videos: temporal fragmentation. For example, a camera wearer might begin preparing a recipe, switch to an unrelated task, and return to the cooking process minutes later. A standard model relying solely on raw video frames struggles to link these temporally distant moments. By semantically clustering non-contiguous substeps into a single, cohesive step memory, our approach inherently bridges these gaps, robustly preserving the long-range procedural context and yielding significant performance gains even before any question-specific processing is applied.

However, while dynamic memory captures procedural context, a text-based summary alone cannot resolve every egocentric challenge. For instance, text memory is inherently limited for 3D spatial reasoning questions—such as determining the camera-relative direction of a fixture—which require precise geometric understanding rather than procedural text. Furthermore, even with perfect memory, a frozen MLLM can easily attend to the wrong granularity if not explicitly guided on how to process the multimodal input for specific tasks. To address these gaps, we introduce Inference Skills, providing reusable procedural guidelines for different question protocols. We categorize these into four skill families: memory-guided skills that isolate relevant hierarchical levels to prevent distraction; input-adaptation skills that redirect the model's visual focus (e.g., toward fine-grained hand motions rather than high-level goals); tool-assisted skills that construct auxiliary evidence (e.g., rendering 3D top-down maps) when text memory is insufficient; and per-choice verification skills for robust candidate evaluation. Together, these skills explicitly dictate how the frozen MLLM should gather, verify, and reason over evidence. A comparison of inference pipeline against standard approaches is provided in Figure 1.

Our contributions are summarized as follows:

- We propose *DyhimemSkill*, a fully training-free framework that synergizes Dynamic Hierarchical Memory and Inference Skills to maximize the capabilities of frozen MLLMs for long-form egocentric video QA.
- We design Dynamic Hierarchical Memory, which constructs a query-agnostic substep–step–goal architecture by semantically grouping local narrations, inherently connecting non-contiguous observations to preserve long-range procedural context.
- We define Inference Skills as reusable input-side procedures—categorized into four distinct families—that explicitly guide the model's cognitive process.
- We demonstrate consistent, significant gains across HD-EPIC (Perrett et al., 2025), EgoSchema (Mangalam et al., 2023), and QAEgo4D (Bärmann and Waibel, 2022), proving that memory and skills provide crucial, complementary improvements.

2 Related Work

2.1 Egocentric Video Understanding

Egocentric video understanding is important for embodied agents, robotics, assistive systems, and AR/VR applications. Compared with third-person videos, egocentric videos involve frequent camera motion, viewpoint changes, occlusions, close-range hand-object interactions, and long temporal dependencies. Recent benchmarks evaluate these challenges through long-form reasoning, temporal grounding, episodic memory, and fine-grained interaction understanding (Mangalam et al., 2023; Bärmann and Waibel, 2022; Jia et al., 2022; Perrett et al., 2025). Recent egocentric MLLM approaches improve first-person video understanding through data- and model-centric adaptation (Ye et al., 2025; Luo et al., 2025; Pei et al., 2026). In contrast, DyhimemSkill keeps the backbone MLLM frozen and instead improves egocentric VQA through inference-time dynamic memory and question-specific skills.

2.2 Hierarchical Video Understanding

Many human activities exhibit an inherent hierarchical structure, where short-term local actions naturally compose higher-level steps and overarching goals (Grauman et al., 2022; Song et al., 2023; Grauman et al., 2024). Recent works have leveraged this structural prior; for instance, Video ReCap (Islam et al., 2024) performs recursive captioning on hour-long videos across multiple temporal granularities, while OpenHOUSE (Kang et al., 2025) extends this concept to streaming egocentric videos by generating hierarchical descriptions

in an online fashion. Although these works establish hierarchy as a useful abstraction for long egocentric videos, Dyhimem utilizes this structure for a distinctly different purpose. Instead of training models to generate hierarchical captions, Dyhimem constructs a reusable, inference-time memory scaffold explicitly designed for VQA. By summarizing fixed local chunks into substeps and dynamically grouping them by semantic relevance into higher-level step memories, our approach bridges temporally fragmented observations to unify non-contiguous evidence.

2.3 Skills for LLM/MLLM Agents

Recent agentic AI systems increasingly use skills as reusable packages of inference-time procedural knowledge (Zhang et al., 2025a). In this context, SkillsBench (Li et al., 2026) formalizes Agent Skills as structured artifacts that augment LLM agents at inference time by providing procedures, workflows, domain conventions, optional resources, and verification logic. This characterization explicitly distinguishes them from plain prompts, factual retrieval, or instance-specific answers. We adapt this powerful abstraction to egocentric VQA. In our setting, an *inference skill* is a reusable procedure for a class of questions that specifies how a frozen MLLM should use hierarchical memory, select visual evidence, invoke optional tools, transform tool outputs, verify candidate answers, and select the final response. Thus, skills do not update model parameters; they provide inference-time scaffolding for collecting, transforming, and checking multimodal evidence. In DyhimemSkill, dynamic hierarchical memory stores semantically organized evidence, while inference skills determine how that evidence should be used for each question class.

3 Method

DyhimemSkill is a training-free framework for long-form egocentric video question answering with frozen MLLMs. It consists of two inference-time components: *Dynamic Hierarchical Memory* and *Inference Skills*. Dynamic Hierarchical Memory constructs a reusable, query-agnostic representation of a long egocentric video, while Inference Skills specify how a frozen MLLM should use memory, visual evidence, optional tools, and answer candidates for different question classes. No backbone MLLM is fine-tuned on any evaluated benchmark.

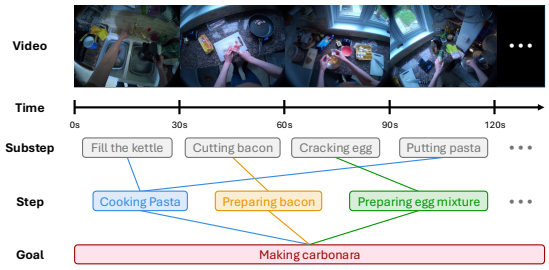

Figure 2: **Overview of dynamic hierarchical memory building.** Short video segments are summarized into substeps, grouped into higher-level steps, and consolidated into a video-level goal.

3.1 Overview

Our core design explicitly separates *what is stored* from *how it is used*. Given a long egocentric video V, we first construct a query-agnostic hierarchical text memory $\mathcal{M}(V)$. Built once per video, this memory contains three distinct levels: a video-level goal, semantically grouped step memories, and timestamped 30-second substeps.

Subsequently, at inference time, when presented with a multiple-choice question q, we deterministically map the query to a specialized inference skill according to its benchmark protocol. The selected skill constructs a set of structured input-side evidence—integrating targeted context from the memory $\mathcal{M}(V)$, visual inputs from sampled frames, and optional auxiliary tool outputs—which is then processed by the frozen answer MLLM to produce the final response.

3.2 Dynamic Hierarchical Memory

Dynamic Hierarchical Memory (Dyhimem) converts a raw egocentric video into a hierarchical text memory through three stages (Figure 2).

Substep generation. We divide each video into fixed-length temporal chunks. For each chunk, a frozen MLLM generates a temporally grounded substep narration describing visible actions, manipulated objects, object states, and scene context. When captioning the current chunk, previously generated substeps are provided as a history window in the prompt, allowing the frozen MLLM to maintain temporal continuity and resolve references across neighboring chunks.

Semantic step grouping. To form higher-level step memories, we encode all substep narrations into vector embeddings using a frozen sentence embedding model. We then compute pairwise cosine similarities between substep embeddings and apply

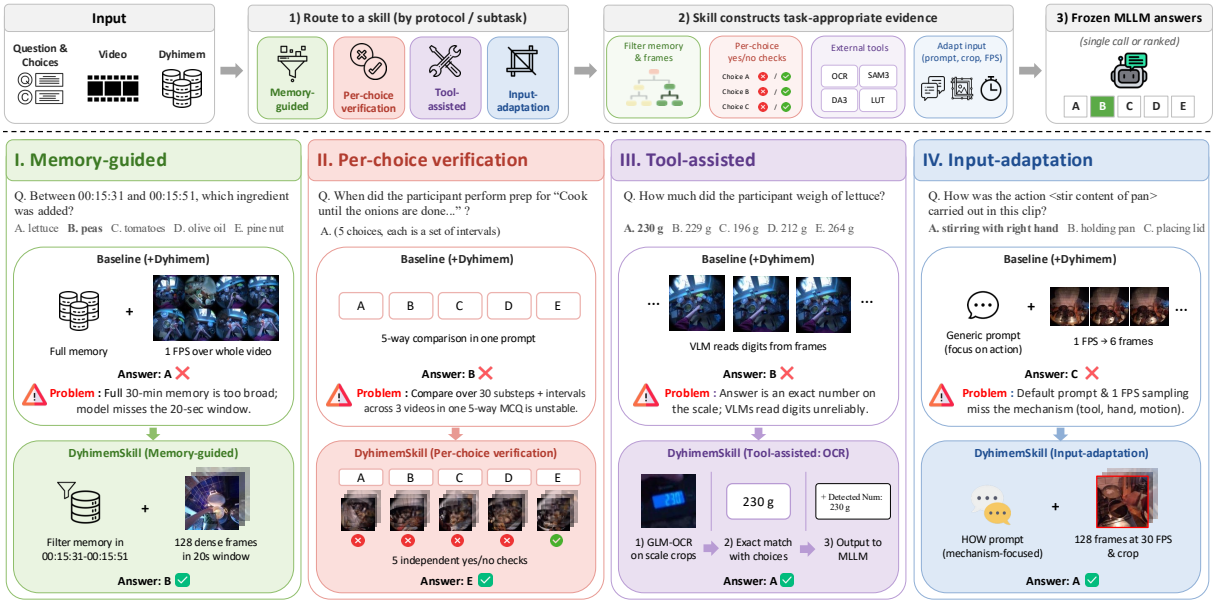

Figure 3: **Overview of DyhimemSkill.** DyhimemSkill routes each question to a protocol-defined inference skill, which constructs task-appropriate evidence from dynamic hierarchical memory, video frames, and optional external tools before querying a frozen MLLM. The examples illustrate four skill families: memory-guided, per-choice verification, tool-assisted, and input-adaptation.

average-linkage agglomerative clustering over the resulting cosine-distance matrix. This grouping is semantic rather than strictly temporal: substeps can be assigned to the same step even when they are non-contiguous in time. This property is important in egocentric video, where a camera wearer may switch tasks, revisit an object, or resume an earlier step after unrelated actions.

Step and goal generation. For each cluster of substeps, a frozen LLM/MLLM generates a compact step summary. The overall goal is then summarized from these step memories. The final memory contains three granularities: timestamped substeps, semantically grouped steps, and an overall goal. We provide an example of the constructed Dyhimem in Appendix 16.

3.3 Inference Skills

Dynamic hierarchical memory provides reusable video context, but it does not specify how a frozen MLLM should use that context for each question. We therefore define *Inference Skills*: deterministic inference-time procedures that construct task-appropriate evidence for a class of questions. A skill may specify the prompt format, relevant memory fields, temporal window, visual crop, auxiliary tools, candidate verification rule, or answer-selection procedure. Skills do not update model parameters and operate entirely through input-side evidence construction.

Figure 3 shows the overall inference pipeline and representative example for each skill family. Given a question and multiple-choice candidates, the system first routes the question to a protocol-defined skill family. The selected skill then constructs task-specific evidence from Dyhimem, video frames, and, when needed, external tools. Finally, the frozen answer MLLM predicts the answer from the constructed evidence either in a single call or through ranked candidate-wise calls.

Most questions are routed deterministically using a question protocol, either provided by the benchmark or manually defined by grouping similar question types. If the matched protocol has a registered skill, the corresponding skill is applied. Otherwise, the system falls back to Dyhimem inference, using the hierarchical memory without additional skill-specific evidence construction. We organize inference skills into four families.

Memory-guided skills. Memory-guided skills are used when the answer is likely to be present in Dyhimem, but the full memory is too broad or distracting. Instead of passing the entire video-level hierarchy, the skill narrows the memory and frame evidence to the portion required by the question. For example, in fine-grained temporal retrieval tasks, a question asks which ingredient was added between `00:15:31` and `00:15:51`. The baseline receives the full 30-minute memory and sparse

video frames, even though only a 20-second interval matters. The memory-guided skill parses the timestamp from the question text, filters Dyhimem to substeps overlapping that interval, and samples dense frames only within the queried window. This design targets cases where the required evidence is already encoded in Dyhimem, but the frozen MLLM needs input-side guidance to attend to the correct abstraction level, entity, or temporal region.

Per-choice verification skills. When each candidate represents a complex hypothesis, direct multi-way classification often becomes unreliable, making per-choice verification a more robust alternative. Instead of asking the model to compare all choices at once, the skill decomposes the multiple-choice problem into independent yes/no verification calls. For each candidate, the model receives candidate-specific evidence and answers whether that candidate satisfies the question. The final answer is selected by ranking the candidates using the yes/no score margin. For instance, in complex temporal localization tasks, each answer choice might consist of several time intervals distributed across multiple videos. A single 5-way prompt requires the model to compare more than thirty candidate moments at once. The skill instead samples frames and memory substeps for each candidate interval set, asks whether the target preparation step occurs in that set, and chooses the interval set with the strongest yes-over-no evidence. This design is useful when the model can make reliable local binary judgments but struggles with global cross-choice comparison.

Tool-assisted skills. To overcome the limitations of generic MLLM prompting in extracting reliable evidence, the framework incorporates tool-assisted skills. These skills invoke external tools prior to generating the final response, converting their outputs into structured evidence (textual, numeric, spatial, or visual) to guide the frozen answer model. Specific integrations include OCR for reading frame-level numeric measurements, nutrition databases for dietary values, object detection and segmentation for precise localization, 3D reconstruction for camera-aligned spatial reasoning, and an advanced narration generator to capture subtle object motions. To illustrate, answering an ingredient weight question requires reading the precise value displayed on a digital scale (e.g., 230 g). While baseline MLLMs often confuse similar digits like 230 and 229, the OCR skill avoids this by cropping scale regions and directly matching the detected numbers to the choices. Consequently, this tool augmentation ensures accuracy whenever a task necessitates exact measurements, external facts, geometric structures, or detailed tracking—areas where an unconstrained MLLM would otherwise fail.

Input-adaptation skills. Input-adaptation skills are used when the needed evidence is already visible, but the default prompt or frame sampling causes the model to attend to the wrong aspect of the clip. These skills only adapt the input format, including the system prompt, crop, temporal coverage, frame budget, or visual annotation. For example, in an fine-grained "how" question, the answer depends on the physical mechanism of an action: which utensil is used, which hand moves, and what motion pattern is performed. The default prompt and 1-FPS sampling may focus on the action label itself, or may be misled by a coarse memory narration such as "stirring with a wooden spoon." The input-adaptation skill replaces the prompt with a mechanism-focused directive, filters memory to the local clip neighborhood, and samples dense frames at the native FPS so that the utensil and hand motion are visible. Fundamentally, this strategy is highly effective when the challenge lies not in missing data, but in guiding the model's visual attention.

The four families differ in what they change at inference time. Detailed protocol-to-skill assignments and implementation details for all benchmarks are provided in Appendix G.

4 Experiments

4.1 Benchmarks and metric.

We evaluate DyhimemSkill on three long-form egocentric video question answering benchmarks: HD-EPIC (Perrett et al., 2025), EgoSchema (Mangalam et al., 2023), and QAEgo4D (Di and Xie, 2024). We report accuracy for all benchmarks; for HD-EPIC, we additionally report Macro-7, the average accuracy across its seven macro-categories.

HD-EPIC (Perrett et al., 2025) contains an egocentric VQA benchmark with 26,550 questions across 30 subtasks, grouped into seven macro-categories: Recipe, Ingredient, Nutrition, Fine-grained Action, 3D Perception, Object Motion, and Gaze.

EgoSchema (Mangalam et al., 2023) is a long-range egocentric multiple-choice QA benchmark with over 5,000 human-curated question-answer pairs spanning diverse human activities. We use

Model	Recipe	Ingredient	Nutrition	Action	3D	Motion	Gaze	Avg.
Sample Human Baseline	96.7	96.7	85.0	92.5	93.8	92.7	75.0	90.3
VideoLlama2 (Cheng et al., 2024)	30.8	25.7	32.7	27.2	25.7	28.5	21.2	27.4
LongVA (Zhang et al., 2024a)	29.6	30.8	33.7	30.7	32.9	22.7	24.5	29.3
LLaVA-Video (Zhang et al., 2024b)	36.3	33.5	38.7	43.0	27.3	18.9	29.3	32.4
Gemini Pro (Team et al., 2024)	60.5	46.2	34.7	39.6	32.5	20.8	28.7	37.6
Qwen3-VL-32B (Bai et al., 2025a)	49.9	35.5	40.3	49.6	37.3	30.1	34.6	39.6
Qwen3-VL-235B (Bai et al., 2025a)	55.0	38.5	47.0	53.4	35.8	35.7	36.8	43.2
GLM-4.6V (Hong et al., 2025)	47.5	40.2	40.0	44.3	32.8	32.0	33.8	38.7
From Pixel to Graphs (Taluzzi et al., 2025)	64.6	54.7	38.0	48.3	42.6	30.2	31.2	44.2
OpenMMEgo (Luo et al., 2025)	-	-	-	-	-	-	-	42.4
Egocentric Co-Pilot (Yang et al., 2026)	-	-	-	-	-	-	-	46.2
Qwen2.5-VL-7B (Bai et al., 2025b)	35.1	34.5	37.3	41.8	29.7	30.8	34.0	34.8
+ DyhimemSkill	53.9	52.8	51.3	48.3	39.0	41.0	35.5	46.0
Qwen3-VL-8B (Bai et al., 2025a)	48.0	33.8	30.7	45.1	35.5	38.5	32.5	37.7
+ DyhimemSkill	77.6	60.5	56.3	52.4	50.2	49.1	46.3	56.1
Qwen3.5-27B (Qwen Team, 2026)	52.6	43.0	38.0	53.2	36.4	29.6	38.9	41.7
+ DyhimemSkill	85.5	69.2	61.0	63.7	46.7	37.3	53.6	59.6

Table 1: **HD-EPIC VQA (Perrett et al., 2025) results.** We compare DyhimemSkill with baselines reported in HD-EPIC, recent egocentric systems evaluated on HD-EPIC, and additional frozen MLLM baselines. Rows marked with "+ DyhimemSkill" add dynamic hierarchical memory and inference skills to the same frozen backbone, without finetuning the backbone model.

the official 500-question subset, referred to as the $\text{EgoSchema}_{\texttt{subset}}$.

QAEgo4D (Di and Xie, 2024) is a long egocentric video QA benchmark with temporally grounded questions. We use $\text{QAEgo4D}_{\texttt{test-mc}}$, a 500-question multiple-choice test split.

4.2 Implementation Details

Backbones and memory builder. All MLLMs used in our experiments are frozen. We evaluate several open-source MLLM backbones, including Qwen2.5-VL (Bai et al., 2025b), Qwen3-VL (Bai et al., 2025a), Qwen3.5 (Qwen Team, 2026), and GLM-4.6V (Hong et al., 2025). No backbone MLLM is fine-tuned on any evaluated benchmark.

Memory construction. Across HD-EPIC, EgoSchema, and QAEgo4D, we build Dyhimem with the same code, prompts, and hyperparameters. Videos are split into 30-second chunks, and Qwen3-VL-8B generates timestamped substep captions from 1 FPS frames using greedy decoding. When captioning chunk i, we include up to the previous W=10 generated substeps in the prompt as a sliding history window, namely substeps $\max(1, i-W), \ldots, i-1$. Substeps are embedded with Qwen3-Embedding-0.6B (Zhang et al., 2025b) and grouped by average-linkage agglomerative clustering over precomputed cosine distances, using a similarity threshold τ=0.68 ($1-\tau = 0.32$ distance threshold). Clusters are summarized into steps and then into a video-level goal.

Auxiliary tools. For tool-assisted skills, external tools are used only to construct input-side evidence. We use Depth Anything 3 (Lin et al., 2025) for 3D reconstruction, SAM3 (Carion et al., 2025) and Grounding DINO (Liu et al., 2024) for object detection and segmentation, GLM-OCR (Duan et al., 2026) for text and measurement reading, the USDA FoodData Central database (U.S. Department of Agriculture, Agricultural Research Service, 2019) for nutrition lookup, and Qwen3-VL-32B (Bai et al., 2025a) for narration generator. The outputs of these tools are converted into prompts, rendered images, structured text, or answer candidates for the frozen answer model.

Inference settings. We compare three settings: Baseline, which directly feeds sampled video frames and the question to the MLLM; +Dyhimem, which adds Dyhimem; and +DyhimemSkill, which further applies inference skills. All memory-building and answering calls use greedy decoding. Appendix E provides prompts, hyperparameters, and additional implementation details.

4.3 Results

Results on HD-EPIC. Table 1 compares DyhimemSkill with baselines reported in HD-EPIC, recent egocentric video understanding systems, and additional frozen MLLM baselines. HD-EPIC remains challenging for general-purpose video-language models: Gemini Pro achieves 37.6, and even Qwen3-VL-235B reaches only 43.2, far below the sample human baseline of 90.3. Recent egocentric-oriented methods improve this range by introducing additional structure or task-specific training. From Pixels to Graphs (Taluzzi et al.,

Model	EgoSchema$_{subset}$	QAEgo4D$_{test\text{-}mc}$
EgoThinker	66.6	55.2
Qwen2-VL-7B	62.2	56.6
+ DyhimemSkill	70.4	57.2
Qwen2.5-VL-3B	58.0	50.4
+ DyhimemSkill	69.4	53.2
Qwen2.5-VL-7B	68.4	55.2
+ DyhimemSkill	73.8	58.0
Qwen3-VL-8B	72.0	56.6
+ DyhimemSkill	77.6	62.2

Table 2: **EgoSchema$_{subset}$ (Mangalam et al., 2023) and QAEgo4D$_{test\text{-}mc}$ (Di and Xie, 2024) Results**. Rows marked with "+ DyhimemSkill" apply dynamic hierarchical memory and inference skills to the same frozen backbone.

Category	Baseline	+30 sec	+Dyhimem
Recipe	48.0	69.5	**71.0**
Ingredient	33.8	43.0	**44.8**
Nutrition	30.7	35.3	**39.3**
Action	45.1	45.0	**48.3**
3D	35.5	**41.7**	40.9
Motion	38.5	37.0	**39.0**
gaze	32.5	32.2	**40.7**
Avg.	37.7	43.4	**46.3**

Table 3: **Memory format ablation on HD-EPIC using Qwen3-VL-8B.** "+30 sec" uses flat 30-second substep summaries, while "+Dyhimem" uses the full goal–step–substep hierarchy.

Model	Baseline	+Dyhimem	+DyhimemSkill
HD-EPIC			
Qwen2.5-VL-7B	34.8	37.2	**46.0**
Qwen3-VL-8B	37.7	46.3	**56.1**
Qwen3.5-27B	41.7	48.5	**59.6**
EgoSchema$_{subset}$			
Qwen2-VL-7B	62.2	68.6	**70.4**
Qwen2.5-VL-3B	58.0	67.2	**69.4**
Qwen2.5-VL-7B	68.4	70.2	**73.8**
Qwen3-VL-8B	72.0	73.4	**77.6**
QAEgo4D$_{test\text{-}mc}$			
Qwen2-VL-7B	56.6	**59.4**	57.2
Qwen2.5-VL-3B	50.4	**54.0**	53.2
Qwen2.5-VL-7B	55.2	57.4	**58.0**
Qwen3-VL-8B	56.6	60.6	**62.2**

Table 4: **Component analysis of DyhimemSkill.** Results show the cumulative effect of adding dynamic hierarchical memory and inference skills across three egocentric video QA benchmarks.

2025), built on Gemini 2.0 Flash (Google), uses structured scene and knowledge graphs and reports 44.2. OpenMMEgo (Luo et al., 2025) and Egocentric Co-Pilot (Yang et al., 2026) are based on Qwen2.5-VL-7B fine-tuning and report 42.4 and 46.2, respectively.

In contrast, DyhimemSkill keeps the answer MLLM frozen and improves performance through inference-time memory construction and skill-guided evidence construction. With the same Qwen2.5-VL-7B backbone scale used by the fine-tuning-based systems, DyhimemSkill reaches 46.0, matching or outperforming these recent egocentric models without updating model parameters. With stronger frozen backbones, the gains become larger: DyhimemSkill reaches 56.1 with Qwen3-VL-8B and 59.6 with Qwen3.5-27B, outperforming the strongest prior egocentric-oriented result in the table by 9.9 and 13.4 points, respectively.

Across the frozen backbones evaluated with our full framework, DyhimemSkill improves over the corresponding baseline by 11.2 points for Qwen2.5-VL-7B, 18.4 points for Qwen3-VL-8B, and 17.9 points for Qwen3.5-27B. These results suggest that dynamic hierarchical memory and inference skills provide a strong training-free alternative to task-specific fine-tuning, and that they remain complementary to backbone scaling.

Results on other egocentric video QA benchmarks. Table 2 evaluates DyhimemSkill on EgoSchema$_{subset}$ and QAEgo4D$_{test\text{-}mc}$ using multiple Qwen backbones. We also compare with EgoThinker, an egocentric fine-tuned Qwen2-VL-7B model. With the same Qwen2-VL-7B backbone, DyhimemSkill achieves 70.4 on EgoSchema$_{subset}$ and 57.2 on QAEgo4D$_{test\text{-}mc}$, outperforming EgoThinker by 3.8 and 2.0 points, respectively.

For each benchmark, we apply the corresponding question-protocol skills on top of Dyhimem. DyhimemSkill improves all evaluated frozen backbones on both datasets, with the strongest configuration reaching 77.6 on EgoSchema$_{subset}$ and 62.2 on QAEgo4D$_{test\text{-}mc}$. These results indicate that dynamic hierarchical memory and inference skills are broadly useful across long-form egocentric QA settings with different question distributions.

4.4 Ablation Study

We conduct ablation studies to analyze the two main components of DyhimemSkill: dynamic hierarchical memory and inference skills. Table 3 isolates the effect of memory organization, while Table 4 analyzes the cumulative effect of first adding dynamic hierarchical memory and then applying inference skills.

Memory format ablation. Table 3 compares different memory formats on HD-EPIC using Qwen3-VL-8B. Here, +30 sec stores 30-second substep summaries as a flat chronological memory, while Dyhimem builds a hierarchy by semantically clustering substeps into higher-level procedural steps.

The flat memory improves over the baseline, showing that dense temporal summaries provide

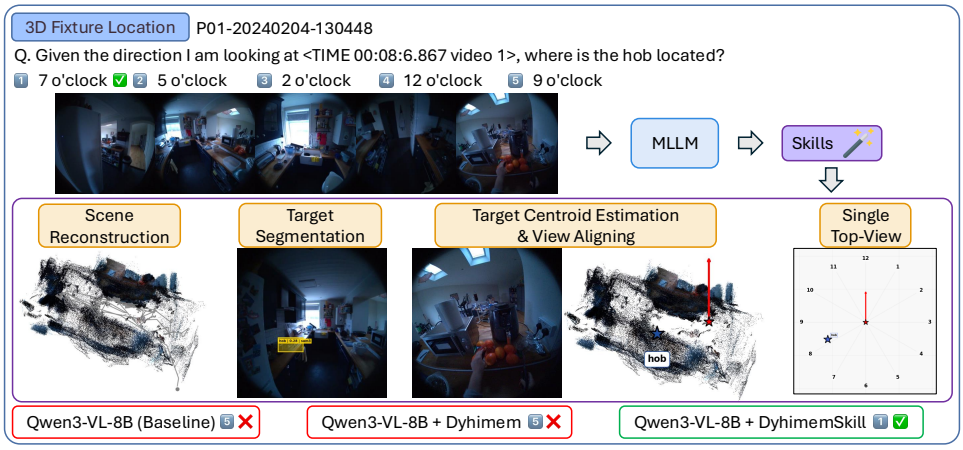

Figure 4: **Qualitative examples of DyhimemSkill.** For 3D fixture localization, the skill renders a camera-aligned top-view map with the target fixture localized, providing structured spatial evidence for clock-direction prediction.

useful long-range context. The hierarchical memory further improves overall performance and outperforms the flat format on most categories. This indicates that the benefit of Dyhimem comes not only from converting video into textual summaries, but also from semantically organizing those summaries into a procedural hierarchy.

Component analysis. Table 4 analyzes the cumulative effect of the two components of DyhimemSkill across HD-EPIC, $\text{EgoSchema}_{\texttt{subset}}$, and $\text{QAEgo4D}_{\texttt{test-mc}}$. The baseline uses sampled video frames and the question, the memory-augmented setting adds Dyhimem, and the full framework further applies inference skills. Adding hierarchical memory improves over the baseline in every evaluated setting, confirming that Dyhimem provides useful temporal context before any skill-specific evidence construction.

The skill stage yields the largest and most consistent gains on HD-EPIC, where the protocols cover diverse question types requiring temporal filtering, candidate verification, external tools, and input adaptation. It also consistently improves over the memory-augmented setting on $\text{EgoSchema}_{\texttt{subset}}$. On $\text{QAEgo4D}_{\texttt{test-mc}}$, the effect is more mixed under the fixed skill configuration: larger backbones benefit from the skill stage, while Qwen2-VL-7B and Qwen2.5-VL-3B perform slightly better with the memory-augmented setting. This indicates that inference skills are useful across benchmarks, but their prompts, routing choices, and decision thresholds can interact with backbone scale and instruction-following ability.

Overall, memory and skills play complementary roles. Dyhimem provides reusable temporal context alongside the sampled video frames, while inference skills specify how that context should be selected, reformatted, verified, or augmented for different question protocols. Together, they provide a stronger input-side framework than raw video prompting and, in most settings, further improve over memory-augmented inference.

4.5 Qualitative Result

Figure 4 illustrates how DyhimemSkill handles a 3D fixture localization question. Given a query asking for the clock direction of the hob relative to the camera-facing direction, the skill performs the necessary visual-spatial processing outside the MLLM: it reconstructs the scene, segments the target fixture, estimates its 3D centroid, and renders a camera-aligned top-view map. The MLLM then receives only this final rendered top-view image and predicts the clock direction from the structured spatial evidence. This example highlights how DyhimemSkill turns raw egocentric evidence into a compact, task-aligned representation for a single frozen-MLLM call.

5 Conclusion

We presented DyhimemSkill, a fully training-free framework that maximizes the capabilities of frozen MLLMs for long-form egocentric video QA. By synergizing a query-agnostic Dyhimem—which connects non-contiguous observations to preserve long-range procedural context—with reusable Inference Skills, our approach explicitly guides the model's cognitive process. Extensive evaluations demonstrate consistent, significant gains across three benchmarks.

Limitations and Future work

While our framework demonstrates strong performance on long-form egocentric video QA, the quality of the dynamic hierarchical memory inherently depends on the baseline visual captioner. However, as vision-language models continue to advance, this dependency will naturally yield even higher-quality memories without architecture changes. Furthermore, our current pipeline utilizes human-designed inference skills and deterministic routing. While recent studies (Li et al., 2026) suggest that self-generated or automated skills currently provide negligible or even negative benefits compared to human-curated expertise, achieving autonomous skill orchestration remains an important goal for general embodied agents. We hope our manually crafted skills and routing framework can serve as a robust baseline and reference for future research exploring automated skill generation and optimal prompt discovery. Finally, while we extensively validate our training-free framework on accessible open-weights models, investigating its synergy with closed-source proprietary APIs is left for future exploration.

References

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, and 1 others. 2025a. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025b. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Leonard Bärmann and Alex Waibel. 2022. Where did i leave my keys?—episodic-memory-based question answering on egocentric videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 1560–1568.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, and 1 others. 2025. Sam 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719*.

Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, and 1 others. 2024. VideoLLaMA 2: Advancing Spatial-Temporal Modeling and Audio Understanding in Video-LLMs. *arXiv preprint arXiv:2406.07476*.

Shangzhe Di and Weidi Xie. 2024. Grounded question-answering in long egocentric videos. In *CVPR*.

Shuaiqi Duan, Yadong Xue, Weihan Wang, Zhe Su, Huan Liu, Sheng Yang, Guobing Gan, Guo Wang, Zihan Wang, Shengdong Yan, and 1 others. 2026. Glm-ocr technical report. *arXiv preprint arXiv:2603.10910*.

Google. Gemini 2.0 flash.

Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, and 1 others. 2022. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18995–19012.

Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, and 1 others. 2024. Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19383–19400.

Wenyi Hong, Wenmeng Yu, Xiaotao Gu, Guo Wang, Guobing Gan, Haomiao Tang, Jiale Cheng, Ji Qi, Junhui Ji, Lihang Pan, and 1 others. 2025. Glm-4.5 v and glm-4.1 v-thinking: Towards versatile multimodal reasoning with scalable reinforcement learning. *arXiv preprint arXiv:2507.01006*.

Md Mohaiminul Islam, Ngan Ho, Xitong Yang, Tushar Nagarajan, Lorenzo Torresani, and Gedas Bertasius. 2024. Video recap: Recursive captioning of hour-long videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18198–18208.

Baoxiong Jia, Ting Lei, Song-Chun Zhu, and Siyuan Huang. 2022. Egotaskqa: Understanding human tasks in egocentric videos. *Advances in Neural Information Processing Systems*, 35:3343–3360.

Hyolim Kang, Yunsu Park, Youngbeom Yoo, Yeeun Choi, and Seon Joo Kim. 2025. Open-ended hierarchical streaming video understanding with vision language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20715–20725.

Xiangyi Li, Yimin Liu, Wenbo Chen, Shenghan Zheng, Xiaokun Chen, Yifeng He, Yubo Li, Bingran You, Haotian Shen, Jiankai Sun, and 1 others. 2026. SkillsBench: Benchmarking how well agent skills work across diverse tasks. *arXiv preprint arXiv:2602.12670*.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. 2025. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, and 1 others. 2024. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pages 38–55. Springer.

Hao Luo, Zihao Yue, Wanpeng Zhang, Yicheng Feng, Sipeng Zheng, Deheng Ye, and Zongqing Lu. 2025. OpenMMEgo: Enhancing egocentric understanding for LMMs with open weights and data. *arXiv preprint arXiv:2500.00000*. NeurIPS 2025. Replace arXiv ID with the final identifier if available.

Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. 2023. Egoschema: A diagnostic benchmark for very long-form video language understanding. In *Advances in Neural Information Processing Systems*, volume 36.

Baoqi Pei, Yifei Huang, Jilan Xu, Yuping He, Guo Chen, Fei Wu, Jiangmiao Pang, and Yu Qiao. 2026. Egothinker: Unveiling egocentric reasoning with spatio-temporal cot. *Advances in Neural Information Processing Systems*, 38:44140–44168.

Toby Perrett, Will Price, Jorge Cartucho, and Dima Damen. 2025. Hd-epic: A highly-detailed egocentric video dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23901–23913.

Qwen Team. 2026. Qwen3.5: Towards native multimodal agents.

Yale Song, Eugene Byrne, Tushar Nagarajan, Huiyu Wang, Miguel Martin, and Lorenzo Torresani. 2023. Ego4d goal-step: Toward hierarchical understanding of procedural activities. *Advances in neural information processing systems*, 36:38863–38886.

Agnese Taluzzi, Davide Gesualdi, Riccardo Santambrogio, Chiara Plizzari, Francesca Palermo, Simone Mentasti, and Matteo Matteucci. 2025. From pixels to graphs: using scene and knowledge graphs for hd-epic vqa challenge. *Preprint*, arXiv:2506.08553.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, and 1 others. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

U.S. Department of Agriculture, Agricultural Research Service. 2019. FoodData Central.

Sicheng Yang, Yukai Huang, Weitong Cai, Shitong Sun, Fengyi Fang, You He, Yiqiao Xie, Jiankang Deng, Hang Zhang, Jifei Song, and 1 others. 2026. Egocentric co-pilot: Web-native smart-glasses agents for assistive egocentric ai. In *Proceedings of the ACM Web Conference 2026*, pages 8862–8873.

Hanrong Ye, Haotian Zhang, Erik Daxberger, Lin Chen, Zongyu Lin, Yanghao Li, Bowen Zhang, Haoxuan You, Dan Xu, Zhe Gan, and 1 others. 2025. Mmego: Towards building egocentric multimodal llms for video qa. In *International Conference on Learning Representations*, volume 2025, pages 71705–71723.

Barry Zhang, Keith Lazuka, and Mahesh Murag. 2025a. Equipping agents for the real world with agent skills. *Anthropic Engineering Blog*.

Peiyuan Zhang, Kaichen Zhang, Bo Li, Guangtao Zeng, Jingkang Yang, Yuanhan Zhang, Ziyue Wang, Haoran Tan, Chunyuan Li, and Ziwei Liu. 2024a. Long context transfer from language to vision. *arXiv preprint arXiv:2406.16852*.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, and 1 others. 2025b. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*.

Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. 2024b. Video instruction tuning with synthetic data. *arXiv preprint arXiv:2410.02713*.

A Potential Risks

While our training-free framework, DyhimemSkill, significantly enhances the reasoning capabilities of egocentric video agents, we acknowledge several potential risks associated with its deployment. Although the framework bypasses fine-tuning costs, the repeated inference calls required for memory generation and skill execution increase latency and energy consumption, posing challenges for real-time deployment on low-power wearable devices. Additionally, despite being grounded in visual evidence, frozen MLLMs remain susceptible to hallucinations—such as misinterpreting procedural steps or object states—which could lead to unsafe consequences if the system's outputs are used to guide physical robotic actions.

B License

Artifact	Type	License
HD-EPIC	Dataset	CC BY-NC 4.0
EgoSchema	Dataset	MIT License
QAEgo4D	Dataset	MIT License
Qwen-VL Series	MLLM	Apache License 2.0
GLM-4.6V	MLLM	MIT License
GLM-OCR	Tool	MIT License
SAM3	Tool	Meta SAM License
Grounding DINO	Tool	Apache License 2.0
Depth Anything 3	Tool	Apache License 2.0

Table 5: **Summary of licenses for key artifacts.** We strictly adhere to the licensing terms and usage policies of all external datasets, models, and tools utilized in our framework.

C Artifact Use and Intended Use.

We use all external datasets, models, and tools only for research and evaluation purposes, in accordance with their original licenses, access conditions, and intended uses. Derived artifacts produced by our framework, including generated substep narrations, hierarchical memory entries, tool outputs, and intermediate predictions, are used solely within the research context of this work and are not intended for use outside the scope permitted by the corresponding source artifacts.

D Benchmark Details

We evaluate on three egocentric multiple-choice VideoQA benchmarks, summarized in Table 6. The benchmarks differ in video duration, question design, and required reasoning skills, providing complementary evaluation settings for long-form egocentric video understanding.

Benchmark	Duration	#Videos	#QA	Type
HD-EPIC	15.9 ($\pm$ 14.5) min	156	26,550	MC
$\text{QAEgo4D}_{\texttt{test-mc}}$	8.3 min	148	500	MC
$\text{EgoSchema}_{\texttt{subset}}$	3 min	500	500	MC

Table 6: **Summary of the evaluation benchmarks.** MC denotes multiple-choice VideoQA.

HD-EPIC (Perrett et al., 2025) is a highly detailed egocentric VQA benchmark collected from unscripted kitchen activities. We evaluate on 156 videos with 26,550 multiple-choice questions. The benchmark contains 30 subtasks grouped into seven macro-categories: Recipe, Ingredient, Nutrition, Fine-grained Action, 3D Perception, Object Motion, and Gaze, with 8, 6, 3, 4, 4, 3, and 2 subtasks, respectively. It emphasizes detailed human-object interaction, recipe understanding, object motion, 3D perception, and gaze-based reasoning.

EgoSchema (Mangalam et al., 2023) evaluates long-range video-language understanding over three-minute Ego4D clips. We use the official 500-question subset, denoted $\text{EgoSchema}_{\texttt{subset}}$. Each question has five answer choices and is designed to require understanding information distributed across an extended temporal context.

QAEgo4D (Di and Xie, 2024) focuses on question answering in long egocentric videos with temporal grounding. We use $\text{QAEgo4D}_{\texttt{test-mc}}$, a 500-question multiple-choice test split over 148 videos. This benchmark evaluates whether a model can answer episodic questions that often require locating relevant temporal evidence in long first-person videos.

E Additional Implementation Details

For HD-EPIC, each question is paired with up to 128 frames decoded at 1 FPS from the native 1408×1408 / 30 FPS source, with frames center-cropped at 1.0 (no crop) and resized to 512×512.

For $\text{EgoSchema}_{\texttt{subset}}$, we uniformly sample 32 frames per question across the full clip; frames use the lmms-eval default Qwen-VL processor settings.

For $\text{QAEgo4D}_{\texttt{test-mc}}$, we uniformly sample 32 frames per question, with the same processor settings as EgoSchema.

Decoding and reproducibility. All MLLM calls use greedy decoding (`do_sample`=False, `temperature`=None) in `bfloat16` with `flash_attention_2` (SDPA fallback if unavailable). Answering calls emit a single token

(max_new_tokens=2, letter A–E); Memory-building calls allocate up to 2048 new tokens per substep, step, goal generation. On HD-EPIC we fix the random seed to 2025 throughout and cache decoded frames on disk so that re-running with the same cache yields byte-identical predictions; $\text{QAEgo4D}_{\texttt{test-mc}}$ and $\text{EgoSchema}_{\texttt{subset}}$ are run through lmms-eval with greedy decoding and the framework's default fixed seeds.

Model size and Budget. All experiments run on a single node with 8×NVIDIA H200 (141 GB) using HuggingFace's `device_map="auto"` for the Qwen2.5-VL-7B, Qwen3-VL-8B, Qwen3-VL-32B, Qwen3.5-27B backbones and FSDP-style sharding via `accelerate launch` for the Qwen3-VL-235B-A22B-Instruct and GLV-4.6V(106B). For tool-assisted skills, we additionally integrated several specialized auxiliary models, including SAM 3 (848 M), DA3-streaming (120 M), GLM-OCR (0.9 B), and Grounding DINO-B ($\sim$ 240 M). In total, our experiments required approximately 3,600 GPU hours, including dynamic hierarchical memory construction, benchmark evaluation, and skill discovery.

Prompt details. For reproducibility, we provide the prompt templates used in our pipeline. Tables 12–14 show the prompts for dynamic hierarchical memory construction, while Table 15 shows the prompt for narration generation.

F Additional Analysis

Table 7 reports per-subtask accuracy across all backbones and inference settings. The overall gains across diverse question categories show that dynamic hierarchical memory and inference skills help frozen MLLMs handle heterogeneous question types in long-form egocentric video understanding.

The main experiments use a fixed skill configuration designed with Qwen3-VL-8B and transfer it unchanged to other backbone scales, testing the cross-backbone transferability of the proposed skills. Table 8 additionally reports the best configuration selected for each backbone. The results show that scale-specific skill configurations can recover the small drop observed for Qwen2.5-VL-3B on QAEgo4D and further improve performance on several smaller-backbone settings. Thus, while the proposed skills are generally transferable, adapting prompts and decision settings to each backbone can yield additional gains.

Figure 5 illustrates how dynamic hierarchical memory helps procedural step localization. A flat 30-second memory divides the video into fixed temporal chunks, such as 0–30, 30–60, and 60–90 seconds, without explicitly grouping actions by their procedural meaning. As a result, it can fail to identify which local observations correspond to the queried recipe step. In contrast, Dyhimem semantically groups related substeps into a high-level procedural step, providing a more structured representation of the task context. This hierarchy helps the model retrieve the relevant action evidence and select the correct localization intervals.

G Inference Skills Details

This appendix expands the four skill families defined in §3.3 with the specific instances used in our experiments. We first document how the question-type axis (*protocol*) is derived for the two benchmarks where it is not shipped with the data ($\text{QAEgo4D}_{\texttt{test-mc}}$, $\text{EgoSchema}_{\texttt{subset}}$) in §G.1, then expand the four skill families in §G.2 listing the template variants we use, and finally give the per-(protocol, model) routing tables that produced the paper main-table accuracies.

G.1 Protocol definition

HD-EPIC's release JSON carries a `subtask` field per question; each subtask maps one-to-one to a skill and the router is simply `route[subtask]`. $\text{EgoSchema}_{\texttt{subset}}$ and $\text{QAEgo4D}_{\texttt{test-mc}}$ ship with question text only — we therefore classify each question into a protocol using a single-pass regex on the question stem (top-down, first-match-wins). Tables 9 and 10 list the resulting protocols, their question shape, and per-protocol counts on the 500-question evaluation set.

G.2 Skill families and template variants

We group the inference skills used in our experiments into the four families introduced in §3.3. Each subsection below expands a family with the concrete template variants we deploy. Most templates appear on multiple benchmarks under different parameter settings; we mark the benchmark in parentheses next to each template.

Memory-guided skills (Family I). These skills shape *how the memory is presented* to the answer model: either by adding a focusing directive on top of the full memory, or by filtering the memory to a relevant subset before the final call.

	Recipe Recognition	Multi-Recipe Recognition	Multi-Step Localisation	Step Localisation	Prep Localisation	Step Recognition	Rough Step Localisation	Following Activity Recognition	Ingredient Retrieval	Ingredient Weight	Ingredients Order	Ingredient Adding Localisation	Ingredient Recognition	Exact Ingredient Recognition	Image Nutrition Estimation	Nutrition Change	Video Nutrition Estimation	Action Recognition	How Recognition	Why Recognition	Action Localisation	Fixture Location	Object Location	Object Contents Retrieval	Fixture Interaction Counting	Object Movement Itinerary	Object Movement Counting	Stationary Object Localisation	Gaze Estimation	Interaction Anticipation
Qwen2.5-VL-7B																														
Baseline	34.0	70.0	32.0	40.0	20.0	15.0	22.0	48.0	27.0	34.0	38.0	38.0	40.0	30.0	32.0	16.0	64.0	57.7	38.6	50.6	20.4	27.4	31.0	29.0	31.3	25.4	40.0	27.0	41.8	26.2
+ Dyhimem	12.0	68.0	52.0	46.0	33.0	42.0	51.0	64.0	45.0	40.0	50.0	42.0	26.0	30.0	26.0	12.0	64.0	60.2	41.0	55.4	21.6	24.8	36.6	47.5	29.3	23.2	39.5	27.0	39.8	25.6
+ DyhimemSkill	12.0	68.0	52.0	46.0	68.0	70.0	51.0	64.0	75.0	78.0	50.0	54.0	26.0	34.0	26.0	**64.0**	64.0	60.2	45.6	61.2	26.2	30.0	37.0	47.5	41.7	37.0	39.5	**46.5**	45.4	25.6
Qwen3-VL-8B																														
Baseline	50.0	68.0	60.0	36.0	30.0	29.0	47.0	64.0	24.0	38.0	52.0	37.0	28.0	24.0	24.0	14.0	54.0	60.9	44.2	51.6	23.6	26.0	31.2	55.0	29.7	33.0	53.5	29.0	42.6	22.4
+ 30 sec	50.0	72.0	92.0	68.0	65.0	61.0	76.0	72.0	61.0	36.0	52.0	49.0	40.0	20.0	24.0	18.0	64.0	60.8	43.6	51.8	23.8	25.2	42.6	**63.5**	35.3	36.4	46.0	28.5	42.1	22.3
+ Dyhimem	62.0	72.0	86.0	72.0	66.0	64.0	74.0	72.0	69.0	36.0	54.0	50.0	40.0	20.0	28.0	20.0	70.0	63.8	44.2	59.2	25.9	21.4	46.4	60.5	35.3	38.0	48.5	30.5	45.6	35.8
+ DyhimemSkill	62.0	78.0	86.0	76.0	81.0	87.0	79.0	72.0	87.0	78.0	54.0	70.0	40.0	34.0	37.0	62.0	70.0	63.8	52.8	64.6	28.2	40.8	50.8	60.5	**48.7**	**52.4**	48.5	46.5	56.8	35.8
Qwen3-VL-32B																														
Baseline	40.0	76.0	84.0	32.0	32.0	30.0	31.0	74.0	24.0	28.0	56.0	37.0	42.0	26.0	23.0	28.0	70.0	66.4	51.0	57.8	23.3	30.2	41.6	41.5	36.0	20.4	42.5	27.5	46.3	22.9
+ Dyhimem	72.3	72.0	98.0	80.0	81.0	74.0	77.0	76.0	70.0	28.0	60.0	55.0	54.0	22.0	28.0	18.0	68.0	70.1	48.2	60.6	26.0	26.4	53.4	51.5	40.3	22.2	49.0	31.5	46.4	32.3
Qwen3.5-27B																														
Baseline	42.0	68.0	82.0	46.0	40.0	42.0	35.0	66.0	35.0	46.0	60.0	47.0	42.0	28.0	28.0	12.0	74.0	69.5	55.2	60.6	27.5	32.2	39.2	45.0	29.0	16.8	46.5	25.5	45.7	32.1
+ Dyhimem	**74.0**	80.0	**100.0**	**86.0**	79.0	75.0	84.0	72.0	78.0	38.0	60.0	59.0	**64.0**	28.0	24.0	18.0	**80.0**	71.6	55.6	62.8	28.8	33.2	47.6	46.5	34.7	23.2	23.5	30.5	49.6	35.1
+ DyhimemSkill	74.0	**86.0**	100.0	86.0	**85.0**	**91.0**	**90.0**	72.0	**91.0**	**80.0**	60.0	**80.0**	64.0	**40.0**	**43.0**	60.0	80.0	**80.8**	**65.6**	**70.6**	**37.6**	**41.4**	**54.4**	46.5	44.3	42.0	23.5	46.5	**65.5**	**41.6**
Qwen3-VL-235B																														
Baseline	56.0	74.0	78.0	48.0	44.0	27.0	35.0	**78.0**	29.0	40.0	56.0	36.0	42.0	28.0	29.0	44.0	68.0	70.5	55.0	63.6	24.6	29.2	33.0	46.5	34.7	19.2	52.0	36.0	46.4	27.1
+ Dyhimem	58.0	76.0	98.0	82.0	82.0	74.0	80.0	76.0	74.0	40.0	**64.0**	47.0	56.0	8.0	18.0	22.0	74.0	71.4	54.2	67.0	27.9	28.4	44.6	51.0	36.0	22.8	**54.0**	37.5	61.9	31.2
GLM-4.6V																														
Baseline	42.0	66.0	78.0	44.0	32.0	22.0	18.0	78.0	25.0	44.0	54.0	40.0	44.0	34.0	24.0	24.0	72.0	59.4	39.0	56.0	23.0	23.2	27.0	52.0	29.0	18.6	45.5	32.0	45.2	22.3
+ Dyhimem	66.0	74.0	98.0	64.0	71.0	70.0	70.0	68.0	70.0	40.0	64.0	47.0	50.0	24.0	29.0	22.0	78.0	62.5	43.4	59.6	25.1	18.8	41.2	51.5	33.0	27.0	48.5	36.5	45.4	27.2

Table 7: HD-EPIC VQA per-subtask results across all backbones and inference settings (% Acc.).

Model	**EgoSchema**$_{\texttt{subset}}$	**QAEgo4D**$_{\texttt{test-mc}}$
Qwen2-VL-7B	62.2	56.6
+ DyhimemSkill	72.4	61.0
Qwen2.5-VL-3B	58.0	50.4
+ DyhimemSkill	69.8	55.4
Qwen2.5-VL-7B	68.4	55.2
+ DyhimemSkill	75.8	58.6
Qwen3-VL-8B	72.0	56.6
+ DyhimemSkill	77.6	62.2

Table 8: **Per-backbone best skill results on EgoSchema**$_{\texttt{subset}}$ **and QAEgo4D**$_{\texttt{test-mc}}$**.** While the main paper uses a fixed skill configuration designed with Qwen3-VL-8B and transferred across scales, this table reports the best DyhimemSkill configuration for each frozen backbone. Rows marked with "+ DyhimemSkill" apply dynamic hierarchical memory and question-protocol inference skills without fine-tuning.

Protocol	N	Question shape
Q1 loc_recall	36	"where did I put / place / leave X?"
Q2 loc_prior	143	"where was X before / when / after . . . ?"
Q3 obj_action	184	"what / which did I *verb*?"
Q4 attribute	58	"what color / size / shape / kind / material was X?"
Q5 count	61	"how many / how much X?"
Q6 social	18	"who . . . ?"

Table 9: QAEgo4D$_{\texttt{test-mc}}$ protocol distribution. All 500 questions are classified by regex on the question stem; coverage is 100%. The classifier also produces per-question auxiliary fields (`verb`, `object`, `object_keywords`, `attr_type`) consumed by entity-anchored memory-guided skills.

- **Memory-hint recognition** — the lowest-cost variant. The full Dyhimem is rendered as text and a single protocol-specific instruction is inserted between the memory and the question, telling the model which memory field (goal, steps, substeps) is most relevant. *Instances:* HD-EPIC `recipe_step_recognition`, `recipe_multi_recipe_recognition`, `ingredient_exact_recognition`; EgoSchema: `P2--P8`. Variation is in the hint sentence only; the mechanism is shared.

- **Entity-anchored memory window** — parses the question's noun phrase, expands it with a synonym pool, filters memory substeps to those mentioning the entity, and redirects the frame budget into the time window around those mentions. *Instances:* QAEgo4D `Q2`. Variation is in window placement (post / pre / union) and frame budget.

- **Timestamp-anchored memory window** — the same mechanism as "Entity-anchored memory window" but the anchor is a `<TIME>` tag parsed from the question rather than an entity. Memory substeps overlapping the tagged interval (± pad) are kept, and the frame budget is concentrated there. *Instances:* HD-EPIC `ingredient_retrieval` (128 frames dense in the window).

Protocol	N	Question shape
P1 goal	194	"what is the overall goal / primary objective?"
P2 steps	79	"what are the key recurring steps / phases?"
P3 why	51	"why does the participant do this?"
P4 method	149	"how does the participant perform / what method?"
P5 tool	12	"what is the primary tool / object?"
P6 moment	8	"most significant moment / in what order?"
P7 anticip.	2	"what will happen next?"
P8 spatial	5	"where does the activity happen?"

Table 10: EgoSchema$_{\texttt{subset}}$ 8 protocol distribution.

Per-choice verification skills (Family II). A single template that decomposes an n-way MCQ into n independent yes/no calls and reads first-token logits to pick the choice with the strongest positive evidence.

- **Per-choice yes/no verification** — for each candidate answer c, we build a prompt "*Is c the correct answer to question Q?*", forward once, and compute $\text{score}(c) = \text{logit}(\text{YES}) - \text{logit}(\text{NO})$ on the first-token logits (averaged across `{YES,Yes,yes}` and `{NO,No,no}` BPE variants). The predicted letter is $\arg\max_c \text{score}(c)$. *Instances:* HD-EPIC `recipe_prep_localization`, `recipe_rough_step_localization`, `recipe_step_localization`, `ingredient_adding_localization`, `fine_grained_action_localization`; QAEgo4D `Q3`,`Q6`; EgoSchema `P1`. Variation is in the choice anchor shape (interval / noun / essay).

Tool-assisted skills (Family III). These skills invoke an external tool before the final MLLM call to construct evidence that frozen prompting does not extract reliably. The tool output is fed back to the model as structured text, image crop, rendered scene, or auxiliary narration; the model then picks the multiple-choice letter.

- **Digit OCR with exact-match override** — GLM-OCR is run on dense frames with two crop regions over the digital-scale display; if a detected integer exactly matches one of the choice values, that letter is emitted (OCR overrides the VLM); otherwise the baseline MLLM is called as fallback. *Instance:* HD-EPIC `ingredient_weight`.
- **USDA nutrient look-up table** — the VLM is asked only to name the ingredient (or amount in grams); a deterministic solver then fits the 4-vector (kcal, carb, fat, protein) of each choice against a 277-entry USDA LUT entry indexed by the VLM's ingredient string, and picks the minimum-residual choice. *Instances:* HD-EPIC `nutrition_change` (4D residual against a VLM-anchored LUT row), `nutrition_image_estimation` (LUT $\times$ VLM-estimated grams).
- **3D-geometry / per-frame detection stack** — SAM3 segmentation + Depth-Anything-3 reconstruction + Grounding-DINO detection trackers produce a structured artifact such as a bounding box, top-view PNG, or detection track; the VLM then reads this artifact and selects the letter. *Instances:* HD-EPIC `3d_fixture_location` (SAM3 + DA3 + topview render), `3d_object_location` (SAM3 bbox crop + before/after composite frame), `3d_fixture_interaction_counting` (two-pass: bbox $\rightarrow$ class noun $\rightarrow$ narration filter $\rightarrow$ letter call).
- **Long-clip atomic-narration generator** — a Qwen3-VL-32B-Instruct narration generator is run over 30-second sliding clips of each video and emits one high-density atomic-action sentence per clip. The dense narration stream is used as the memory source for downstream questions whose answers depend on micro-motions that the coarser 8B substep captions in Dyhimem routinely lose ("slides 5 cm to the left and back, twice"). *Instances:* HD-EPIC `object_motion_itinerary`, `object_motion_stationary_localization`.

HD-EPIC Family-III total: 8 subtasks. Family III is HD-EPIC-only because only HD-EPIC asks questions whose answers are extra-modal (exact weight, calorie change, 3D direction, per-frame counts, atomic object motion); QAEgo4D-MC and EgoSchema-subset are pure MCQ over scene context and do not require tool extraction.

Input-adaptation skills (Family IV). A single MLLM call whose prompt and / or visual shape is tuned for a specific question type.

- **Subtask-tuned prompt directive** — a hand-crafted instruction block names the axes the model should attend to, with no other change.

Benchmark	I. Mem.-guided	II. Per-choice	III. Tool-assisted	IV. Input-adapt.
EgoSchema	P2–P8	P1	—	—
QAEgo4D	Q2	Q3 , Q6	—	—
HD-EPIC	step_recog, multi_recipe, ing_exact, ing_retrieval	prep_loc, rough_step_loc, step_loc, ing_add_loc, action_loc	**OCR**: ing_weight. **USDA LUT**: nut_change, nut_img. **SAM3+DA3**: 3d_fix_loc, 3d_obj_loc, 3d_fix_cnt. **32B narr.**: motion_itin, motion_stat.	how, why, gaze_est, gaze_antic

Table 11: **Family × benchmark assignment.** The four inference-skill families are dispatched per (benchmark, protocol).

Instances: HD-EPIC fine_grained_action_recognition (verb-noun extraction), fine_grained_how_recognition (how-axis: focus on hand motion, tool usage, body posture, contact points), fine_grained_why_recognition (why-axis: focus on downstream goal / current task context).

- **Visual framing tweak** — the frame budget and / or crop are reshaped to put the answer-relevant pixels in view. No choice decomposition, no tool. *Instances:* HD-EPIC gaze_estimation (center crop, dual-view ego frame), gaze_interaction_anticipation (clip-tail frames with −no_memory to block future leakage).

H AI Usage

We utilized Gemini to fix typos and polish sentences, and Claude Code for minor code revisions.

Figure 5: **Qualitative examples of Dyhimem.** The examples illustrate how hierarchical memory provides long-range procedural context for egocentric video question answering.

```
SYSTEM
    You are an expert AI assistant for fine-grained video understanding. Your goal is to analyze visual evidence objectively
      and provide precise action descriptions.

USER
    You are an annotation model for fine-grained video understanding.

    INPUTS
    - CURRENT FRAMES: Visual evidence provided in chronological order.
    - TEMPORAL CONTEXT: The timestamp range (Start to End) and frame count provided in the user prompt.
    - PREVIOUS ACTION HISTORY: A list of immediately preceding actions. Use this to maintain consistency in object identity
      and action flow.

    TASK
    First, analyze the visual evidence to resolve object ambiguity. Then, generate a single, objective sentence describing
      WHAT the person is doing.

    ***[TEMPORAL & LOGICAL REASONING GUIDELINES]***
    1. ANALYZE STATE TRANSITIONS & SEQUENCE (Sequential Completeness):
    - **Track the Full Sequence:** Do not judge the action solely by the first or final frame. If the video segment involves
      multiple distinct interactions (e.g., retrieving Object A, then retrieving Object B), the final description must
      capture the **cumulative sequence of actions**, not just the final state.
    - **No Omission:** Do not omit earlier distinct objects (e.g., items retrieved before the final placement) simply
      because the user interacts with a different object at the very end.
    - **State Definition:** Describe the **active process** (e.g., "placing," "pouring," "stirring") rather than the
      inferred result (e.g., "done," "finished"). Ensure the narrative includes the initiation of the action if it occurred
      within the start timestamps.

    2. FUNCTIONAL AFFORDANCE:
    - If a heavy object (like a pan or container) is placed ON TOP of another object, the bottom object must be a sturdy
      base.

    3. HISTORICAL CONSISTENCY (Object Permanence & Identity):
    - **Maintain Identity across State Changes:** Refer to the "PREVIOUS ACTION HISTORY". If a specific object's identity
      was established in previous steps, **persist that specific label** even if its visual appearance changes due to
      processing (e.g., slicing, cutting, or mixing). Do not downgrade the label to a generic visual descriptor (e.g., use
      the specific ingredient name if known) unless the original identity is irretrievably lost.
    - **Prioritize History for Identity:** Trust the visual evidence for the current state (e.g., "cooked," "sliced"), but
      trust the history for the identity. **Once an object represents a specific class, maintain that label even if its
      appearance changes. Do not rename it based on visual similarity alone; only update the identity if a clear object
      exchange occurs or the visual evidence unambiguously disproves the initial classification.**
    - **Multi-Object Integration:** If the Visual Analysis identifies interactions with **multiple distinct objects** (e.g.,
      a tool, a container, and an ingredient), the ###FINAL sentence must explicitly integrate **all primary objects**
      involved to provide a complete context.
    - **Identity Lock Under Continuous Contact (Anti-Drift):** If the person maintains continuous physical contact with an
      object across adjacent frames (or from the immediately preceding action), you must keep the same identity label for
      that object. You may rename/reclassify it **only** if the CURRENT FRAMES provide explicit visual evidence of an object
      swap (e.g., the prior object is put down and a different object is picked up) or unambiguous re-identification cues
      (e.g., clearly readable text/logo, clearly different form factor) that directly contradict the prior label.
    - **No Content/Category Guessing From Context:** Do not infer an object's contents, ingredient type, or functional
      category from contextual expectations alone (e.g., typical cooking patterns). If identity is uncertain and not
      established in history, use neutral visual descriptors (color/material/shape) rather than a speculative category.

    4. NO UNVERIFIED STATE CLAIMS (Critical):
    - Do **NOT** infer cooking progress or completion states (e.g., "cooked", "boiled", "done", "ready", "served") unless
      there is direct visual evidence in the CURRENT FRAMES.

    5. HANDLING AMBIGUITY:
    - **Specific Naming:** If an object's identity is known from the History or is visually unambiguous, use its **specific
      name** (e.g., "Parmesan cheese," "Screwdriver").
    - **Visual Attribute:** Only when the class is completely unknown and not in the History, describe its **visual
      attributes** (e.g., "a black device") rather than guessing.

    OUTPUT FORMAT
    You must output exactly two blocks.

    Block 1: ## Visual Analysis
    - **Active Objects List:** First, list ALL distinct objects the person interacted with or held during this entire
      sequence (e.g., [First Object, Second Object]).
    - **Sequential Flow:** Briefly describe the order of interactions (e.g., Interaction with Object A -> Interaction with
      Object B -> Combined Action).
    - **Consistency Check:** Explicitly confirm if these objects match any in the PREVIOUS HISTORY.
    - **Reasoning:** Explain *why* you identified these objects as X, referring to visual cues.

    Block 2: ## Substep Description
    - START THE FINAL SENTENCE WITH "###FINAL:"
    - **Consolidation Rule:** Your sentence MUST include ALL objects listed in the "Active Objects List" above. Do not omit
      an object just because the interaction happened at the start of the clip.
    - Provide a single, complete sentence in the present progressive tense.
    - Do not use brackets like < > anymore, just use the marker.
    - Example: ###FINAL: The person is [Action_Verb] [Object_A] and [Object_B] [Location] after [Previous_Action] them.
```

Table 12: **Dyhimem substep prompt.**

```
SYSTEM
    You are an expert AI assistant specialized in Hierarchical Video Understanding.

USER
    Your task is to summarize a sequence of fine-grained "Substeps" (atomic actions) into a single, high-level "Step"
      description (goal-oriented action).

    ### INPUT DATA
    You will be provided with a list of timestamped substeps.
    - These substeps represent a continuous temporal segment belonging to a single higher-level activity.

    ### GOAL
    Synthesize the substeps into ONE concise, imperative sentence that describes the overarching activity based strictly on
      observed facts.

    ### GUIDELINES (Strict Adherence Required)
    1. **Abstraction Level**:
    - Aggregate micro-movements (e.g., "grasp handle," "turn knob") into a functional action (e.g., "Turn on the stove").
    - Focus on **WHAT** is being done, rather than **HOW** it feels or looks.

    2. **Temporal Aggregation**:
    - Merge sequential actions logically using connectors like "and" or "then".
    - If preparatory actions (e.g., picking up a tool) lead to a main action, prioritize the main action.

    3. **Strict Groundedness (Fact-Based Only)**:
    - **NO Hallucination of States**: Do not describe the final outcome or texture (e.g., "until crispy," "smoothly,"
      "tightly") unless explicitly stated in the substeps.
    - **NO External Context**: Do not refer to unseen instructions (e.g., "according to the manual," "as per the recipe").
    - Only include objects and actions that are explicitly present in the input.

    4. **Generalization**:
    - This logic must apply to any domain (technical, household, etc.).
    - Treat specific items simply as objects being manipulated.

    ### EXAMPLES

    **Example 1 (Technical Domain)**
    substeps:
    - Pick up the screwdriver.
    - Align the screwdriver with the screw head.
    - Turn the screwdriver clockwise.
    Output:
    Fasten the screw using a screwdriver.

    **Example 2 (Household Domain)**
    substeps:
    - Slice the bacon into strips using a knife.
    - Put the sliced bacon into a heated pan.
    - Stir the contents.
    Output:
    Slice the bacon and cook it in a pan.

    ### OUTPUT FORMAT
    - Output exactly ONE imperative sentence.
    - Do not use quotes, numbering, or introductory text.
```

Table 13: **Dyhimem step prompt.**

```
SYSTEM
    You are an expert AI assistant specialized in High-Level Activity Summarization and Goal Inference.

USER
    Your task is to infer the high-level **GOAL** of the activity based on the provided sequence of **STEPS**.

    ### INPUT DATA
    You will be provided with a chronological list of "Steps" (actions) that occurred in a video.

    ### GOAL
    Synthesize the steps into a concise Title Phrase that describes what the person is ultimately achieving.

    ### GUIDELINES (Strict Adherence Required)
    1. **Holistic Inference (Parts to Whole)**:
    - Identify the final product or system being created, modified, or repaired.
    - If multiple steps describe preparing separate components (e.g., base, topping, sub-assembly), combine them into a
      **specific name for the final dish or task** rather than listing them separately.
    - **Logic**: If the combination of ingredients or parts points to a well-known result (e.g., specific recipe, specific
      repair), use that specific name.

    2. **Handling Multi-Tasking vs. Single-Tasking**:
    - **Single Task**: If all steps logically contribute to one final outcome, output a single goal (e.g., "Making
      Carbonara").
    - **Multi-Task**: Use "**and**" only if the activities describe clearly distinct, independent products that do not share
      a common purpose (e.g., "Making Pasta and Preparing a Salad").

    3. **Strict Groundedness**:
    - Base the inference strictly on the observed objects and actions.
    - Do NOT use subjective adjectives (e.g., "delicious," "perfect") or assume unseen external contexts.

    4. **Style & Format**:
    - Start with a Gerund (Verbing) or a Specific Noun Phrase.
    - Use Title Case.
    - 10–15 words total; no period at the end.
    - NO sentences, NO numbering, NO explanations.

    ### EXAMPLES

    **Example 1 (Integrating Components - Culinary)**
    steps:
    1. Form ground beef into patties and grill them.
    2. Slice tomatoes, onions, and lettuce.
    3. Slice buns and toast them on the grill.
    4. Assemble the meat and vegetables between the buns.
    Output:
    Making Hamburgers

    **Example 2 (Distinct Multi-Task)**
    steps:
    1. Season the chicken and roast it in the oven.
    2. Wash lettuce, chop tomatoes, and mix with dressing.
    Output:
    Roasting Chicken and Preparing Salad

    **Example 3 (Technical/Assembly Domain)**
    steps:
    1. Unscrew the back panel of the laptop.
    2. Disconnect the old battery connector.
    3. Insert the new battery and connect it.
    4. Screw the back panel back in place.
    Output:
    Replacing a Laptop Battery

    ### OUTPUT FORMAT
    - Output exactly ONE line containing the goal title(s).
```

Table 14: **Dyhimem goal prompt.**

```
SYSTEM
    You are an expert at decomposing egocentric kitchen video clips into atomic actions.
    An atomic action is a single short purposeful movement (0.3-5 seconds) performed by
    one hand or both hands together.

    ### SENTENCE STRUCTURE (build each narration from these slots; do not invent scenes you cannot see)
    <Verb> the <object> [<adjective/location qualifier>] [with the <left|right|both> hand[s]] [to <intent>].

    Examples of the slot pattern (DO NOT copy the example wording -- describe what is
    actually happening in the clip):
      - "<Verb1> the <object1> with the <hand>."
      - "<Verb2> the <object2> from the <location> with the <hand> to <intent>."
      - "<Verb3> the <object3> by holding the <object3-part> with the <hand>."

    ### STRICT RULES
    1. Describe ONLY what is visibly happening in THIS clip; do not generalize or invent
       actions from the system prompt.
    2. Imperative form, starting with a verb in present tense.
    3. Each action is a SINGLE movement; do not summarize multiple movements into one --
       split a multi-step interaction (e.g. open-reach-pickup-close) into separate atomic
       narrations.
    4. List EVERY distinct movement you see, including brief sub-second ones and ones that
       overlap with other movements. No upper limit.
    5. Order = chronological (no start/end timestamps required).
    6. DO NOT repeat the same action narration twice.
    7. Use lowercase verbs and nouns in the structured fields.

    ### OUTPUT FORMAT
    OUTPUT strict JSON only (no prose, no markdown fences, no explanation):
    {"actions":[
      {"narration":"<imperative sentence>",
       "verbs":["<verb>", ...],
       "nouns":["<noun phrase>", ...],
       "hands":["<left hand|right hand|both hands|none>"]}
    ]}

USER
    Look at the {win_dur}-second clip below (frames sampled at 2 fps in chronological
    order). List EVERY distinct atomic movement you observe, in the order they happen, no
    matter how brief. Split multi-step interactions (open-reach-pickup-close, etc.) into
    separate atomic narrations. Do not cap the count -- produce as many as the clip
    genuinely contains. Output JSON only.
```

Table 15: **Atomic-action narration extractor prompt**. A Qwen3-VL-32B model is run once per 30-second Dyhimem window with the SYSTEM and USER messages above; the resulting per-video JSON corpus is consumed by the two HD-EPIC motion skills (`object_motion_itinerary` and `stationary_object_localization`) as structured text at inference time.

```
Goal: Preparing Coffee and Orange Juice and Cleaning Kitchen Appliances
S01 Pour milk into a coffee machine and a mug, stir the coffee, dispose of waste, and rinse a container.
  [0-30s] The person is pouring milk from a bottle into a coffee machine after placing a mug on the counter.
  [120-150s] The person is placing a plastic bag into a food waste bin, adjusting a boiler, and pouring milk from a
      container into a mug while stirring it with a wooden spoon.
  [150-180s] The person is stirring coffee in a mug, disposing of an item in a trash bin, and rinsing a metal container in
      the kitchen sink.
S02 Juice oranges using a juicer and pour the juice into a glass, then clean the juicer and food processor.
  [30-60s] The person is cutting a bag of oranges with scissors after retrieving a milk bottle from the refrigerator and
      using a coffee machine.
  [90-120s] The person is placing a juicer on the counter, preparing to juice an orange on a cutting board with a knife, and
      simultaneously retrieving a kitchen towel from a drawer.
  [180-210s] The person is squeezing an orange into a juicer on a cutting board after slicing it with a knife.
  [210-240s] The person is squeezing an orange into a juicer on a cutting board after placing the orange peel into a trash
      bin.
  [240-270s] The person is squeezing an orange into a juicer on a cutting board after slicing it with a knife.
  [270-300s] The person is squeezing an orange into a juicer on a cutting board.
  [300-330s] The person is washing the juicer's central component in the kitchen sink with a sponge after juicing an orange
      and placing it on a dish rack to dry.
  [330-360s] The person is pouring orange juice from the food processor into a glass after washing the juicer's central
      component and placing it on the dish rack.
  [360-390s] The person is pouring orange juice from the food processor into a glass, then washing the food processor's
      central component in the kitchen sink with a sponge and placing it on the dish rack to dry.
S03 Plug in the food processor, open the dishwasher, retrieve a black container, and return it to the dishwasher.
  [60-90s] The person is plugging in a food processor, opening a dishwasher, retrieving a black container from the
      dishwasher rack, and placing it back inside the dishwasher.
S04 Wipe the stove with a kitchen towel.
  [390-396.267s] The person is wiping the stove with a kitchen towel after retrieving it from under the stove.
```

Table 16: **Example of Dynamic Hierarchical Memory.**