

VideoChat-M1: Collaborative Policy Planning for Video Understanding via Multi-Agent Reinforcement Learning

Supplementary Material

Type	Dataset	Instance Num	Avg Video Length (s)
Temporal Grounding	FineAction	5067	43.64
	QVHighlights	13790	28.36
	HiREST	3617	282.45
Long Video QA	ActivityNet-QA	16642	621.45
	LongViTU	16453	268.46
	MMBench	1673	97.51
	MovieChat	808	457.65
	Neptune	5281	149.25
Spatial Intelligence	HoursVideo	831	568.16
	SpaceR	12643	10.65
Video Reasoning	Video-R1	15123	68.56
	VideoEspresso	9432	56.12
	Video Holmes (Training Set)	1551	91.16
	Total	-	102911

Table 10. Instance numbers of different datasets for VideoChat-M1 training.

A.1 Collected Dataset

To equip VideoChat-M1 with strong generalization across diverse video understanding scenarios, we assemble a comprehensive collection of datasets spanning multiple task types, including temporal grounding, long-video question answering, Spatial Intelligence analysis, and video reasoning. These datasets originate from widely used benchmarks and cover a broad spectrum of video durations, scenes, and annotation forms. The diversity of tasks and data sources empowers VideoChat-M1 to learn from heterogeneous supervision signals, enhancing its capabilities in perceiving, retrieving, and reasoning over long and complex videos. Table 10 summarizes the instance numbers and average video durations of all datasets used in our training pipeline. In total, the dataset collection comprises 102,911 instances with an overall average video duration of 194.6 seconds, laying a solid data foundation for training VideoChat-M1 on four mainstream video tasks.

A.2 Memory Buffer and Tool Use

We implement the memory buffer as a key-value pair structure, in which keys denote the agents’ names and values store the structured information illustrated in Fig 6. We take the memory buffer of Qwen3-8B as an example.

Memory Buffer

Agent Name: Qwen3-8B

The initial plan is: <Global Sampling>, <Video Retrieval>, <Rough Browser>

Key info: The priority is to pinpoint the exact moment the elf interacts with the vacuum and confirm the instantaneous change in its operation that leads to the woman being sucked in.

Turn 1:

Executing Tool: Global Sampling

Tool Output: Uniform Sampling 16 frames of the video. The frame index list is [xxx]

Communication Decision: Continue with the plan, execute Video Retrieval

Turn 2:

Executing Tool: Video Retrieval

Tool Output: Sample 16 frames on clip 6 of the video based on key info. The frame index list is [xxx].

Communication Decision: Continue with the plan, execute Rough Browser

Turn 3:

Executing Tool: Rough Browser

Answer: A

Summary: The critical action occurs when these creatures turn on the vacuum cleaner. The woman then trips over the machine’s power cord, which is a consequence of the creatures’ actions. After she falls, they manipulate the vacuum’s direction, leading directly to her being sucked in by its suction force.

Plan Finished

To enable our multi-agent framework to tackle a diverse array of video understanding tasks, we provide each agent with access to a comprehensive and specialized toolkit \mathcal{T} . These tools facilitate efficient information extraction, spanning coarse-grained retrieval to fine-grained perceptual analysis. The tools available are as follows:

- **Global Sampling:** For queries requiring a holistic understanding, this tool uniformly samples frames across the entire video duration.
- **Video Retrieval:** This tool first divides the video into six equal-length clips. It then employs the ASP-CLIP [9] model to score the semantic similarity between each clip and the user query, returning the highest-scoring clip for

further analysis.

- **Time Stamp Retrieval:** When a precise moment is referenced, this tool extracts a one-minute video segment centered at the specified timestamp.
- **Image Retrieval:** For the image retrieval stage, we uniformly sample frames from the source video at a rate of 2 frames per second (fps). We then employ the pre-trained CLIP model to compute the similarity score between the textual prompt and each sampled frame, ultimately selecting the top 16 (or 32) frames with the highest similarity.
- **Rough Browser:** This tool provides a rapid overview by processing a sparse set of 16 selected frames with a Multimodal Large Language Model (MLLM), such as Qwen2.5-VL-7B [1].
- **Fine Browser:** For detailed analysis where a deeper look is necessary, this tool leverages the same MLLM to process a denser sequence of 32 frames extracted from a targeted video clip.
- **Spatial Tool:** To address spatial reasoning queries, this tool employs the InternVL-3.5-8B [59] model to analyze 16 frames, which are either uniformly sampled or sourced from a retrieved clip.
- **Grounding Tool:** This specialized tool is designed for temporal grounding tasks and utilizes the Eagle2.5-7B [7] model to process the video and identify relevant time segments.

A.3 Prompt

In this section, we detail the prompts employed in each step of our proposed method.

Prompt for Policy Generation

You are an intelligent video understanding agent. Your task is to analyze a video question and select the optimal combination of tools to answer it accurately.

1. Tool Definitions

Group A: Frame Selection Tools (Retrieval Phase)

- **Uniform Sampling:** A general strategy. Use this only when the question is broad or covers the whole video. It summarizes the overall content without focusing on specific details.
- **Video Retrieval:** The standard semantic search method. Use this to locate the most relevant video clips containing the action, event, or object described in the text query.
- **Time Stamp Retrieval:** Deterministic retrieval. Use this strictly when the question mentions a specific time (e.g., “at 01:30”).

- **Image Retrieval:** Fine-grained visual matching. Use this to identify specific static scenes, small objects, or person attributes by matching text descriptions to individual frames (top-k selection).

Group B: Video Browsing Tools (Reasoning Phase)

- **Rough Browser:** Provides a comprehensive yet efficient overview of the selected frames. Sufficient for answering the majority of general video understanding questions.
- **Fine Browser:** High-computation analysis. Use this *only* for cases of extreme ambiguity or when deciphering subtle details (e.g., small text, rapid motions) is critical.
- **Spatial Tool:** Specialized for spatial reasoning benchmarks (e.g., VSIBench). Use this when the question explicitly asks about relative positions, geometry, or spatial arrangements of objects.
- **Grounding Tool:** Specialized for temporal localization (e.g., Charades-STA). Use this strictly for simple, single-scene grounding tasks where the goal is to identify start/end timestamps rather than complex reasoning.

2. Recommended Workflow

You **MUST** adhere to the following selection rules:

1. **Selection Phase:** You must select **ONE** or more tools from Group A (Frame Selection).
2. **Browsing Phase:** You must select **ONE** or more tools from Group B (Video Browsing).
3. "Analyze the question and candidate options to determine the key information necessary for the reasoning process. This becomes your **Key info**."

Current Task: {task}

Question: {question}

3. Output Format & Examples

Example 1 (General Reasoning):

Question: What does the object being chased by the people refer to?

Options: A: Difficulties in life, B: His fully automatic house...

Format:

##key info: the object being chased by the people in the video.

##tool use: <Video Retrieval>, <Rough Browser>

Example 2 (Spatial Intelligence):

Question: If I am standing by the table and facing the bathtub, is the bed to my left, right, or back?

Options: A: left, B: right, C: Back

Format:

##key info: spatial relation between bathtub and bed.

##tool use: <Video Retrieval>, <Spatial Tool>

Prompt for Policy Communication

You are a strategic planning assistant. Your sole responsibility is to evaluate the current execution state and determine the immediate next step.

1. CURRENT CONTEXT

Review the following execution state carefully:

- **Original Question:** {question}
- **Memory buffer:** {memory}
- **Other Agents' Output:** {other agents output}
- **Remaining Plan:** {plan}

2. DECISION PROTOCOL

You must choose exactly **ONE** action from the list below based on the logic provided:

Option A: The Standard Path

- `continue()`: Use this to proceed with the {next tool}. **Rule:** Apply this when peer agents offer no constructive alternatives and the current internal plan remains valid and error-free.

Option B: Exception Handling

- `add tool(tool name='<name>')`: Use this **ONLY** if the current plan is logically flawed and requires a new tool (e.g., 'Video Retrieval') to proceed. Analyze the question, candidate options, and the memory of all agents to determine the key information necessary for the reasoning process. This becomes the **Key info**.

Output your response strictly in the format below.

Format:

Scenario 1: Continuing (Default)

##tool call: continue()

Scenario 2: Adding a Tool (Correction)

##tool call: add tool <Video Retrieval>

##key info: xx

Prompt for Answering the Question

(Use this when the agent's plan is fully executed, but the answer remains unresolved)

You are an intelligent agent responsible for synthesizing a final answer based strictly on the provided internal logs, referred to as {Agent Memory}. You must adhere to the following format constraints based on the presence of options.

Input Context:

- **Question:** {Question}
- **Option:** {Option} (*Note: If this field is empty, treat as an open-ended task or temporal grounding task.*)
- **Task:** {Task}
- **Agent Memory:** {Agent Memory}

Directives:

1. **Source of Truth:** Your response must be derived solely from the information contained within {Agent Memory}. Do not hallucinate or use external knowledge.
2. **Multiple Choice Logic:** If {Option} is provided (e.g., A, B, C, D), your final output must be **the single uppercase letter** corresponding to the correct choice and the reason for your answer.
3. **Open-Ended Logic:** If {Option} is not provided (e.g., Temporal Grounding or open-ended QA), your final output must be a paragraph explaining the reasoning for the answer.

Format:

##Answer: xx

##Reason: xx

Prompt for Reason Summary

Input Context:

You have been provided with the reasoning from four distinct AI agents:

- {Agent0 name}: {agent 0 reason}
- {Agent1 name}: {agent 1 reason}
- {Agent2 name}: {agent 2 reason}
- {Agent3 name}: {agent 3 reason}

Your Task:

Synthesize and summarize the reasons of each agent into a single, cohesive paragraph.

Critically, you must adhere to the following synthesis logic based on the question type:

1. **For Multiple Choice Questions (Options provided):** Identify the final consensus option (or

the selected answer). You must **ONLY** summarize the results and reasoning of the agents that agreed with this final option. Ignore the reasoning of dissenting agents unless it provides critical context for the correct answer.

2. **For Open-Ended Questions (No options provided):** Synthesize and summarize the reasoning from **ALL** agents to provide a comprehensive answer. In particular, the summarization should prioritize the consensus among agents, placing greater emphasis on convergent reasoning paths found in similar responses.

The final summary must be concise but accurately reflect the sequence of events and core logic.

Format:

##Final Answer: xx

##Reason Summary: xx

Prompt for Rough Browser

Input Components:

You will be provided with the following:

- A sequence of key frames extracted from a video.
- Question:{Question} and Options {if have Options or None}
- A key info text that specifies the central theme for the summary. {Key info}

Your Task:

Write a brief summary of the video's content. The summary **must be centered around** the event, object, or action described in the **Key info**. The entire summary must be **no more than 128 tokens**. If you can provide the answer to the question, you can also give the answer.

Output Format:

##Answer: xx

##Summary: A single, concise paragraph containing the summary.

Prompt for Fine Browser

Input Components:

- A video clip requiring detailed examination.
- Question:{Question} and Options {if have Options or None}
- A key info text that directs the model's focus to the most critical aspect of the video for solving

the problem. {key info}

The model's core task is to generate a detailed summary by analyzing 32 uniformly sampled frames from the video clip. This summary must be thematically centered on the event, object, or action specified in the **Key info**. This fine-grained analysis is specifically designed to resolve high ambiguity and decipher subtle details (e.g., small text, rapid motions) that are critical for a correct interpretation. If you can provide the answer to the question, you can also give the answer.

Output Format:

##Answer: xx

##Summary: The final summary must be concise yet descriptive, and it **must not exceed 256 tokens**.

Prompt for Spatial Tool

Input Components:

- A video clip requiring detailed examination.
- Question:{Question} and Options {if have Options or None}
- A key info text that directs the model's focus to the specific spatial question that needs to be answered. {key info}

This tool is specifically invoked for queries concerning the relative positions, geometry, or spatial arrangements of objects, as is common in spatial reasoning benchmarks (e.g., VSIBench). To address these queries, the model's core task is to analyze 32 uniformly sampled frames to build a comprehensive understanding of the scene's spatial layout. It must then generate a descriptive summary that explicitly answers the spatial question posed in the **Key info** by identifying key objects and precisely describing their positions relative to each other. If you can provide the answer to the question, you can also give the answer.

Output Format:

##Answer: xx

##Summary: The final summary must be concise yet descriptive, and it **must not exceed 256 tokens**.

Prompt for Grounding Tool

Given a user-provided textual key info prompt and a video, the model must retrieve the precise time seg-

ment in the video that directly corresponds to the prompt. Furthermore, the model must generate a concise, natural language justification for its selection.

The textual prompt is: {**Key info**}

Output Format:

##Timestamp: [xxs - xxs]

##Reason: xxx

A.4 Visualization

To obtain qualitative insights into our method’s mechanics and efficacy, this section presents a two-part visual analysis of VideoChat-M1. First, Fig 6 and Fig 7 provide a fine-grained visualization of the Collaborative Policy Planning (CPP) process, tracing policy evolution and intermediate reasoning steps to enhance the interpretability of our multi-agent framework. Second, Fig 8 reports a comparative qualitative evaluation, benchmarking the visual outputs of VideoChat-M1 against those of state-of-the-art models across four canonical video understanding tasks. This is intended to empirically validate the performance improvements achieved by our method.

Fig 6 details each step of our CPP process and its corresponding output. It demonstrates that our framework can autonomously refine its plans during execution and exhibits a high degree of fault tolerance, enabling the agent group to recover from errors made by individual agents. The final summary is generated by synthesizing the rationales from all agents that voted for the correct answer (‘A’), a task facilitated by the Qwen3-8B model.

In Fig 7, we present a step-by-step visualization of our CPP framework applied to the open-ended temporal grounding task. Initially, a video retrieval tool is employed as a coarse-grained filter, significantly constricting the temporal search space to a relevant video clip. Subsequently, our CPP method operates within this narrowed window to perform fine-grained boundary refinement. As demonstrated by the query ‘a woman shot the man and escaped,’ the retrieval module effectively eliminated irrelevant footage, enabling our model to focus on the semantic context. Consequently, the method precisely localized the target interval, aligning perfectly with the ground truth, although the Qwen2.5-3B agent failed to find the result.

Fig 8 compares our method with recent Multimodal Large Language Models (MLLMs) across four mainstream tasks. This comparison reveals that existing models frequently rely on superficial cues, miss critical shots, or fail to maintain long-range temporal and spatial consistency, leading to incorrect reasoning. In contrast, VideoChat-M1 reliably identifies causal relations, tracks events over long video durations, infers accurate spatial layouts, and pre-

cisely localizes actions in time. These results show that our collaborative, multi-step reasoning framework delivers more accurate, stable, and interpretable video understanding compared to prior approaches.

A.5 Reinforcement Learning Analysis

While a formal convergence proof for complex Multi-Agent Reinforcement Learning (MARL) systems often remains intractable, we establish a robust rationale for the stability and convergence of our proposed training framework. Its design systematically integrates a series of principles, each targeting a known failure mode in MARL, with convergence anchored in four pillars:

- 1. Policy Initialization via Supervised Fine-Tuning (SFT).** A primary challenge in RL lies in the vast and unstructured exploration space, which causes inefficient or divergent training. Our framework addresses this with a curriculum-driven SFT phase. Specifically, it provides a crucial “warm-start” by pre-training each agent on a corpus of high-quality expert policies. Consequently, the MARL optimization process is initialized in a highly structured and effective region of the joint policy space. This approach circumvents the instabilities of *tabula rasa* learning and substantially improves the tractability of subsequent exploration, or as stated in the work, it is essential for “laying the foundation for collaborative learning in MARL”.
- 2. Stable Policy Updates via Group Relative Policy Optimization (GRPO).** The non-stationarity inherent in multi-agent learning where each agent’s optimal policy shifts as others learn can destabilize policy updates. The GRPO algorithm directly mitigates this by incorporating a KL-divergence penalty, a core principle of robust policy gradient methods like TRPO and PPO. As shown in Eq. 5, its objective function enforces a trust region for policy updates:

$$\max_{\theta} \mathbb{E}_{\mathbf{o} \sim \pi_{\text{old}}} \left[\frac{\pi_{\theta}(\mathbf{o}_k)}{\pi_{\text{old}}(\mathbf{o}_k)} A_R^{(k)} - \beta D_{\text{KL}}(\pi_{\theta} \parallel \pi_{\text{ref}}) \right] \quad (5)$$

This constraint regularizes learning dynamics by limiting excessive deviations from a trusted reference policy (π_{ref}), guaranteeing a monotonic improvement trajectory and fostering training stability.

- 3. Dense and Structured Reward Shaping.** MARL systems often face sparse rewards and credit assignment issues, which cause ill-defined optimization landscapes with suboptimal local equilibria. Our framework mitigates this via a dense and multi-faceted reward signal composed of three components: task success (R_{res}), procedural validity (R_{format}), and collaboration quality (R_{col}). This hybrid structure provides a continuous and informative gradient signal, guiding agents toward cor-

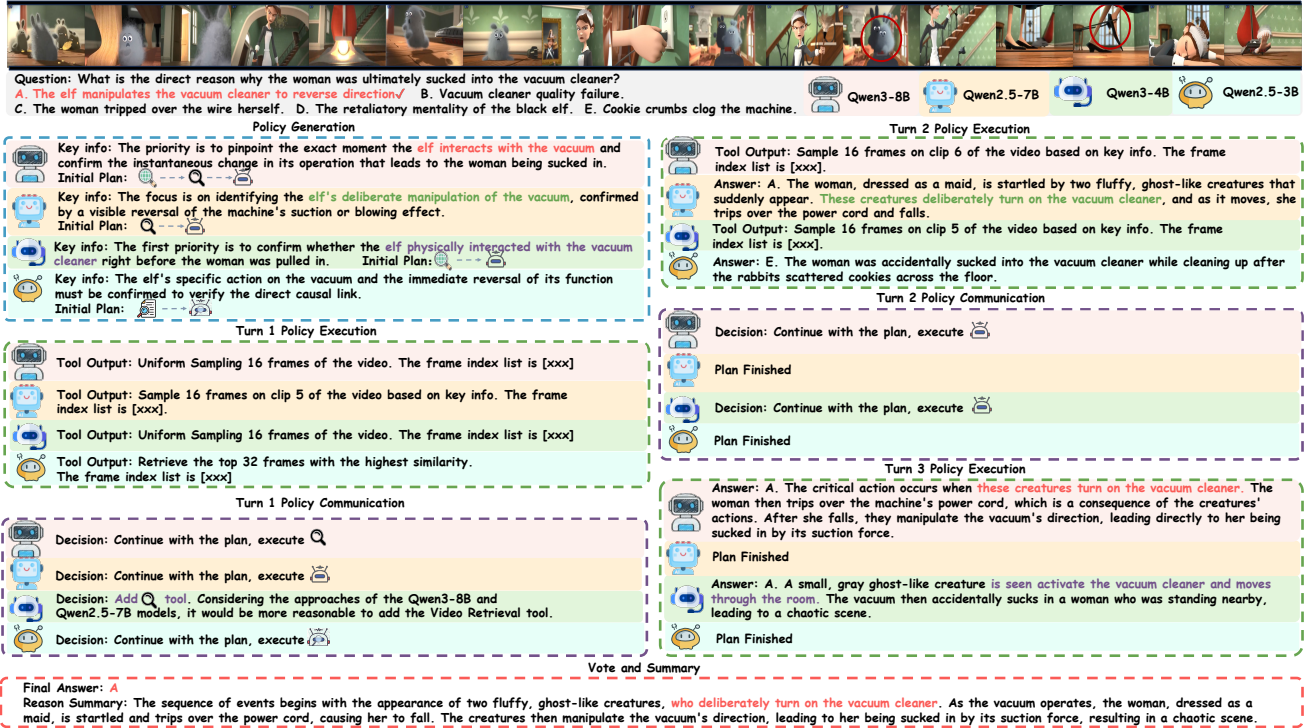


Figure 6. Visualization of VideoChat-M1 at each step of the CPP process.

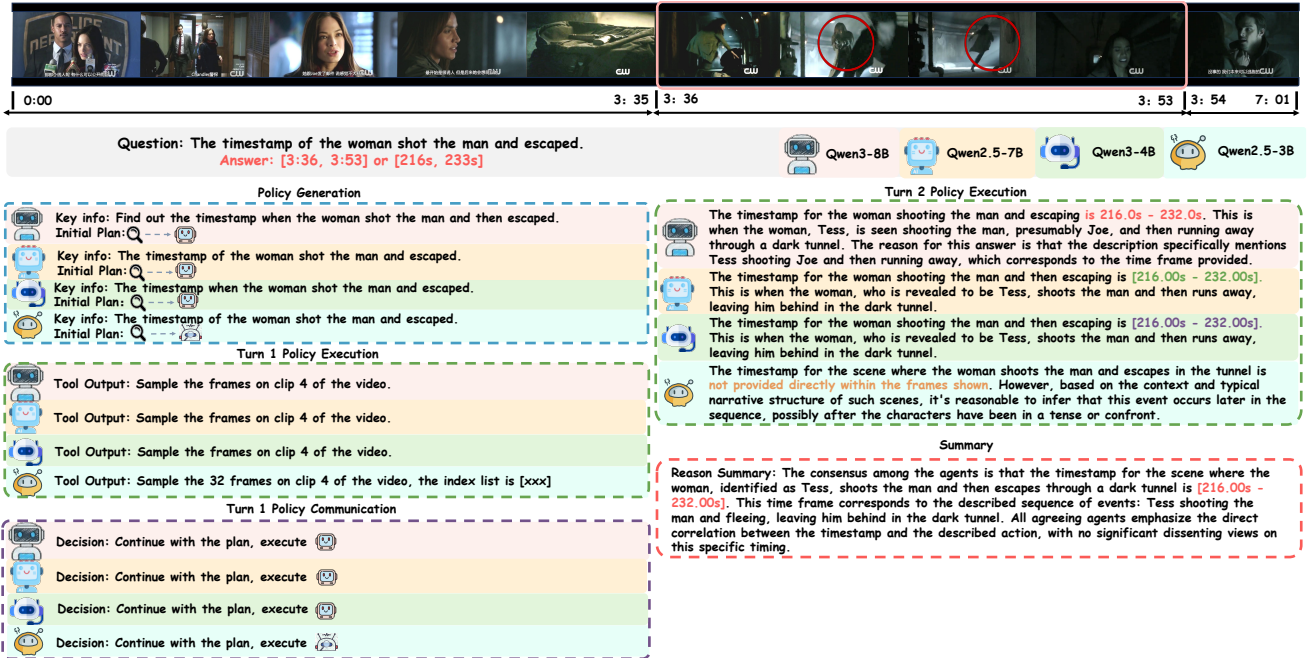


Figure 7. Visualization of VideoChat-M1 on each step of the CPP process on the temporal grounding task.

rect outcomes, valid behaviors and effective cooperation, smoothing the optimization landscape to facilitate gradient-based convergence.

4. Robustness via Agent Dropout Regularization. A common failure mode in MARL is inter-agent co-adaptation,

where agents develop brittle strategies that are over-specialized to their teammates' specific policies. To address this, we employ agent dropout as a form of structural regularization. By dynamically and stochastically adjusting the communication topology during training,



Question: What is the direct reason why the woman was ultimately sucked into the vacuum cleaner?
 A. The elf manipulates the vacuum cleaner to reverse direction/ B. Vacuum cleaner quality failure.
 C. The woman tripped over the wire herself. D. The retaliatory mentality of the black elf.

Gemini 2.5 Pro
 the correct option is C: The woman tripped over the wire herself. Here is a breakdown of the direct cause and effect: after the dust bunnies unplug the vacuum cleaner, the maid becomes enraged and grabs a feather duster to continue her attack. The vacuum's power cord is left stretched across the wooden floor. **The maid does not pay attention to the cord on the floor. Her foot catches on the wire, causing her to trip and fall hard.** While the dust bunnies' desire for revenge (D) is their motivation, and their action of turning the vacuum on is the final blow, the direct physical event that put the woman in a position to be vacuumed was tripping over the cord.

GPT-4o
 Analysis based on visual content (including the ending): In the final scene, it is clear that the black elf manipulates the vacuum cleaner as an act of revenge. He pulls a lever or uses magic to enhance the vacuum's power, which ultimately causes the woman to be completely sucked into it. Correct Answer: D: The retaliatory mentality of the black elf

Qwen3-VL 235B
 The woman is sucked into the vacuum because she trips over the vacuum's power cord. This trip causes her to fall directly onto the machine, which then activates and sucks her in. This is a direct result of her own action of tripping over the wire. Thus, the correct answer is C: The woman tripped over the wire herself.

VideoChat-M1
 Final Answer: A Reason Summary: The sequence of events begins with the appearance of two fluffy, ghost-like creatures, who deliberately turn on the vacuum cleaner. As the vacuum operates, the woman, dressed as a maid, is startled and trips over the power cord, causing her to fall. The creatures then manipulate the vacuum's direction, leading to her being sucked in by its suction force, resulting in a chaotic scene.

(a) Video Reasoning Task



Question: What causes the lampshade of the small light bulb to eventually break?
 A. After being scolded by family, (he/she/they) smashed it B. Overuse of socket power. ✓
 C. Long-term academic stress accumulation. D. Accidentally bumping into something while playing the guitar.

Gemini 2.5 Pro
 Answer: A. The light bulb character smashes their lampshade after experiencing a range of negative emotions and events. The character starts out sad and listless in class, then feels left out when seeing others socializing. Back home, **the character appears to be scolded by their family A.** This accumulation of stress and negative feelings seems to lead to the lampshade breaking.

GPT-4o
 Analysis A: Got scolded by family and smashed it — No signs of emotional breakdown or intentional smashing. B: Accidentally bumping into something while playing the guitar — This is clearly shown; the bulb plays the guitar and the damage follows. C: Long-term academic stress accumulation — Not reflected in the video. D: Deliberately damaged by other light bulbs — No such attack or sabotage is observed. E: Natural aging causes cracking — The damage appears sudden, not due to aging. F: Overuse of socket power — No evidence of socket issues. Correct answer: B. Accidentally bumping into something while playing the guitar. Answer D.

Qwen3-VL 235B
 The direct cause of the small light bulb's lampshade breaking is Option A. Context: The protagonist feels sad and inadequate throughout the video (at school, walking home). Family interaction: At dinner, parents discuss money and education, adding to their emotional burden. Climax: In deep despair, the protagonist deliberately smashes their head (the light bulb) against a wall socket—an intentional self-harm act from accumulated emotional pain (family pressure and disappointment). Thus, A is the most accurate direct cause.

VideoChat-M1
 Final Answer: B Reason Summary: A small light bulb character stands in a bedroom, then sits on the bed. The room's lighting changes as the bulb character moves. In a state indicative of profound despair, the protagonist is then depicted standing before a wall-mounted electrical socket. Subsequent to a significant intensification of the surrounding light, a catastrophic failure of the bulb character occurs, manifesting as a sudden explosion. Therefore, the immediate precipitating factor for this event was direct contact with the socket.

(b) Long Video QA Task



Question: If I am standing by the door and facing the microwave, is the kettle to my left, right, or back?
 A. right ✓ B. back. C. left

Gemini 2.5 Pro
 The kettle is located on the counter that forms a corner with the counter where the microwave is situated (seen clearly around the 0:03 and 1:35 marks). If you are standing by the door (near 1:00) and facing the microwave (near 1:16), you are essentially looking down one side of an L-shaped counter arrangement. The kettle is on the counter that extends perpendicularly from the microwave/sink counter. Therefore, the kettle is to your left, requiring roughly a 90-degree turn to face it. This is less than the "at least 135 degrees" required for the back option. **The correct option is left.**

GPT-4o
 The microwave is fixed above the kitchen counter, located along a specific wall. The door appears near the camera's position at the start, suggesting that if you are standing by the door, you are facing toward the microwave. The kettle is typically seen on the far left side of the counter or in a corner, visible in several frames. When standing at the door and facing the microwave, the kettle is not within your immediate field of view and would require a significant turn to see. According to the rule stated in the question — "An object is to my back if I would have to turn at least 135 degrees to face it" — we determine the relative position as follows: Final Answer: B. back

Qwen3-VL 235B
 The kitchen layout shows the microwave on the counter next to the sink. The kettle is located on the same counter, but further to the right, next to a coffee machine. When you face the microwave from the door, the kettle is positioned on the left side of your field of view. Therefore, the correct answer is C: "left".

VideoChat-M1
 Final Answer: A Reason Summary: The video shows a kitchen scene where a kettle is placed near a microwave. The camera focuses on the kettle as it sits on the counter to the right of the sink, with the microwave visible on a counter to the left of the sink. The kettle appears to be metallic and is positioned to the right of the microwave.

(c) Spatial Intelligence Task



Question: The timestamp of the woman shot the man and escaped. Answer: [3:36, 3:53] ✓

Gemini 2.5 Pro
 The segment from 04:52 to 04:55 shows the woman shooting the man. In this scene, the woman takes a gun from the man's holster and shoots him. Immediately after the shot is fired, the scene cuts away to two different characters in a laboratory setting. There is no depiction of the woman running away or escaping from the location.

GPT-4o
 To identify the exact timestamp when the woman shot the man and escaped, I'll scan the video for that specific event. Here are sampled frames from the video, spaced approximately every 30 seconds. Please inspect the thumbnails and identify the approximate timestamp range where the woman shoots the man and escapes. Once you provide that rough window (e.g., "around 03:30-04:00").

Qwen3-VL 235B
 The timestamp for the event "the woman shot the man and escaped" is [219 s - 223 s]. Here is the reasoning based on the video frames: Contextual Build-up: In the preceding moments (around 208s-218s), Tess is sitting in a dark tunnel, expressing her hunger and thirst to Vincent. He offers her an energy bar, but she declines, saying she is also thirsty. The Action: At 219s, the scene cuts to a close-up of Tess's hand reaching into her jacket. She pulls out a handgun that was hidden there. The Shot: At 220s, she quickly raises the gun and fires it at Vincent. The muzzle flash is visible in the darkness. The Escape: Immediately after firing, at 221s-223s, Tess turns and runs away down the dark tunnel, successfully escaping from her captor.

VideoChat-M1
 Reason Summary: The timestamp for the woman shooting the man and escaping is [216.00s - 232.00s]. This is when the woman, Tess, is seen shooting the man, presumably Joe, and then running away through a dimly lit tunnel. The reason for this answer is that the description specifically mentions Tess shooting Joe and then running away, which corresponds to the given time frame in the video.

(d) Temporal Grounding Task

Figure 8. Visualization of VideoChat-M1 on four mainstream tasks.

this technique discourages dependencies on any single agent and compels the development of more generalized and robust policies. This enhances the stability of the learned multi-agent equilibrium and ensures convergence to solutions resilient to minor policy perturbations, a fact supported by ablation studies identifying it as the "most critical regularizer".

In summary, VideoChat-M1's training convergence is not heuristic but derives from a principled framework design. By systematically addressing initialization (via SFT),

update stability (via GRPO), reward-landscape tractability (via dense rewards) and robust generalization (via agent dropout), the framework holistically mitigates common MARL instabilities, guiding the agent system toward a stable and high-performance collaborative pipeline.

A.6 More Implementation Details

Training Setup. We employ the AdamW [45] optimizer with a learning rate of 1e-7 and a global batch size of 8. The training process is distributed across 8 NVIDIA A100

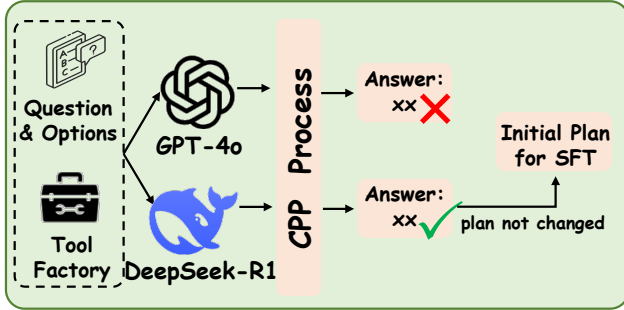


Figure 9. The process of generating the SFT data.

80G GPUs, utilizing the DeepSpeed stage 2 combined with **Flash Attention [17]** and **bfloat16 precision** to accelerate multi-GPU training and optimize memory efficiency. The gradient accumulation step is set to 2. Our agent team consists of four backbone models: Qwen3-8B, Qwen3-4B, Qwen2.5-7B, and Qwen2.5-3B. During training, the temperature is set to 1 for each agent to facilitate exploration, while the KL penalty coefficient β is set to $1e-5$. We set the maximum prompt length to 1024 tokens and the maximum generation length to 1024 tokens. The multi-agent interaction is limited to a maximum of 5 turns. Specific prompts are provided in Appendix A.2. Additionally, we apply agent dropout to enhance the model’s robustness.

LoRA Setting. As reported in Tab 8 of the submitted manuscript, we implement Low-Rank Adaptation (LoRA) using the Hugging Face `peft` library. The LoRA adapters are configured with a rank $r = 8$, a scaling factor $\alpha = 16$, and a dropout rate of 0.05. We adjust the learning rate specifically for LoRA training to $2e-6$. Except for these specific adjustments, all other hyperparameters remain consistent with the full fine-tuning configuration described above.

Optimization Strategy. We adopt Group Relative Policy Optimization (GRPO) as our reinforcement learning algorithm. GRPO is selected for its suitability in scenarios involving optimization from a group of candidate outputs. By normalizing rewards against the team’s average performance, GRPO provides a stable learning signal for each individual agent, aligning naturally with our multi-agent collaborative generation paradigm.

SFT Data Construction. To enable efficient Supervised Fine-Tuning (SFT), we construct a filtered dataset derived from successful interaction trajectories. As illustrated in our pipeline (see Figure 9), tools, questions, and options are input into the Agent Team. Following the Collaborative Policy Planning Process (CPP), a final answer is generated. We retain a trajectory only if: (1) at least one agent provides the correct answer, and (2) the initial plan remains unchanged

throughout the process. We collect 2,000 such initial plans per task. This filtering strategy reduces unnecessary self-correction steps and significantly improves computational efficiency.

Evaluation Setup. For evaluation, all LLMs use a temperature of 0 to ensure deterministic outputs. The agent group composition remains consistent with the training phase (Qwen3-8B, Qwen3-4B, Qwen2.5-7B, and Qwen2.5-3B), totaling approximately 22B parameters. Our reported inference latency (19.8s) is achieved with 4 A100 80G GPUs via parallel processing and bfloat16 precision: (1) we implement parallel processing across the Policy Generation, Execution, and Communication stages, enabling concurrent reasoning and tool invocation across agents (instead of sequential processing); (2) we enforce strict token constraints during reasoning to prompt concise rationales, significantly reducing the decoding overhead. This evaluation can be run on only one A100 80G GPU for each task with partial parallel processing, with about 38.9s per video with 67G VRAM. However, a single A100 80G GPU is insufficient for inference on an MLLM of 72B+ parameters to handle long videos with 100+ sampled frames. When all invoked tools are exhausted or the maximum number of iterations is reached without QA consensus, we directly use Qwen3-8B to generate a summarized answer using the memory of all agents.

Tool Configurations. We tailor the tool set and underlying models for specific benchmarks to maximize performance. The standard tool library includes: *Global Sampling*, *Video Retrieval*, *Time Stamp Retrieval*, *Rough Browser*, *Fine Browser*, and *Grounding Tool*. For image retrieval, we use ViT-CLIP-B/16 (86M). For video retrieval, we use ASP-CLIP (95M).

- **General Video QA (LongVideoBench, Video-MME, MLVU, Video Holmes, MMR-V):** We utilize the standard tool library. The *Browser* model is instantiated with Qwen2.5-VL-7B, and *Grounding Tool* employs Eagle2.5-8B. The grounding tool is invoked with relatively low frequency. The total parameter count for the toolset is approximately 37B.
- **Video MMMU:** The configuration largely follows that of the General Video QA setup, except that the *Browser* model is upgraded to Qwen3VL-8B-Instruct to handle higher domain-specific demands. The total parameter count is approximately 37B.
- **Video VSIBench (Spatial Tasks):** We introduce a specialized *Spatial Tool* powered by InternVL3.5-8B. For spatial queries, the model autonomously selects between the *Browser* (Qwen2.5-VL-7B) or the *Spatial Tool* for answer generation. The total parameter count is approximately 37B.

Spatial Tool	Baseline	VideoChat-M1
InternVL3.5-8B	56.3	71.9
Qwen2.5VL-7B	35.9	70.1

Table 11. Tool Reliance Ablation on VSIBench

- **Charades-STA (Temporal Grounding):** The model dynamically chooses between the *Browser* and the *Grounding Tool*. Input videos are processed at 2 FPS. If the *Video Retrieval* tool is invoked, the retrieved video clip is subsequently fed into the model for fine-grained grounding. The total number of parameters is approximately 37B. For this dataset, we select up to three consecutive video clips. We first select the clip with the highest similarity; if the similarity of an adjacent clip with the key info exceeds 0.35, we include it as well. This prevents the situation where the answer’s grounding time exceeds the duration of the retrieved clip. Additionally, it narrows the retrieval interval and eliminates redundant information, thereby improving performance

Tool Reliance Ablation: To demonstrate that the effectiveness of VideoChat-M1 originates from our Collaborative Policy Planning (CPP) framework rather than reliance on specific SOTA tools, we conducted an additional ablation study on VSIBench (see Tab 11). Specifically, we replaced the specialized ‘Spatial Tool’ (InternVL) with the general-purpose Qwen2.5-VL-7B. Remarkably, even with this generic backbone, our method retains SOTA performance. It continues to outperform the massive InternVL-3.5-241B, achieving a 34.2% improvement over the baseline. This confirms that our MARL-driven planning paradigm delivers substantial gains, independent of the specific tools employed.